Tutorial on Interpretable Machine Learning

Wojciech Samek
(Fraunhofer HHI)

Alexander Binder
(SUTD)

15:00 - 15:15	Introduction & Motivation WS
15:15 - 16:30	Techniques for Interpretability WS
16:30 - 17:00	Coffee Break ALL
17:00 - 18:00	Applications of Interpretability WS
18:00 - 18:50	Case Study: Interpretable ML in Histopathology AB
18:50 - 19:00	Wrap-up WS

Before we start

Joint work with many people

Klaus-Robert Müller (TU Berlin)
Grégoire Montavon (TU Berlin)
Sebastian Lapuschkin (Fraunhofer HHI)
Leila Arras (Fraunhofer HHI)
Frederick Klauschen (Charite)
…

`http://interpretable-ml.org/miccai2018tutorial/`

Please ask questions at any time !

Tutorial on Interpretable Machine Learning

Part 1: Introduction & Motivation

Record Performances with ML

Game GO

Traffic Sign Recognition

Skin cancer detection

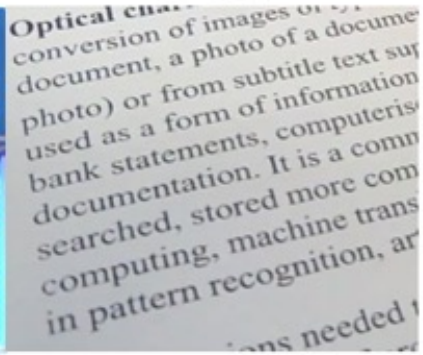
Lung cancer detection

Poker

Computer games

Jeopardy

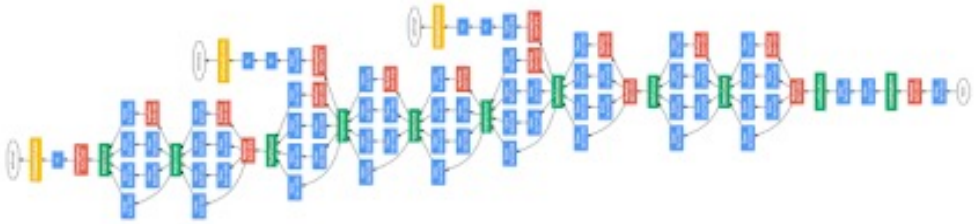
OCR



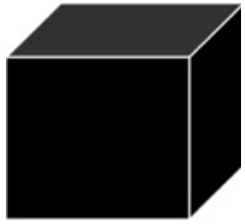
Black Box Models

Huge volumes of data

Computing power



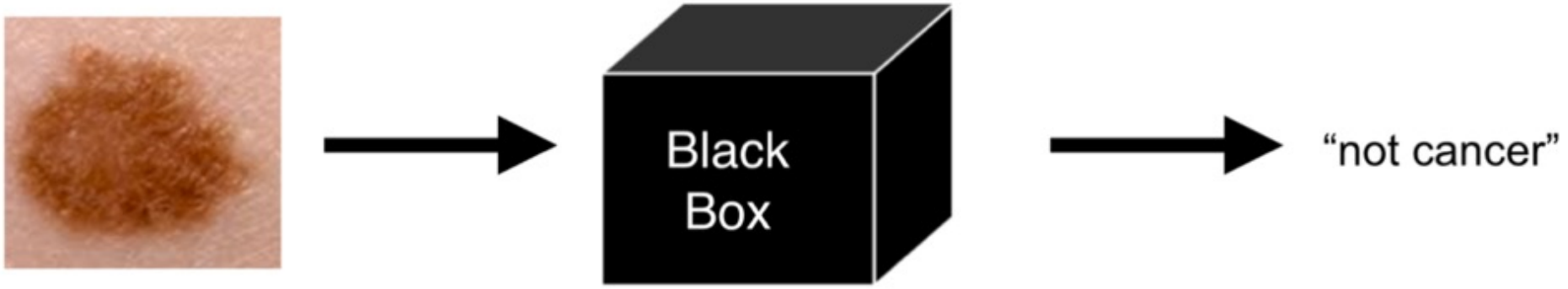
Deep Neural Network



Information (implicit)

Solve task

Black Box Models



Is minimizing the error a guarantee for the model to work well in practice?

Why interpretability ?

Why Interpretability ?

We need interpretability in order to:

verify system

legal aspects

understand weaknesses

learn new things from data

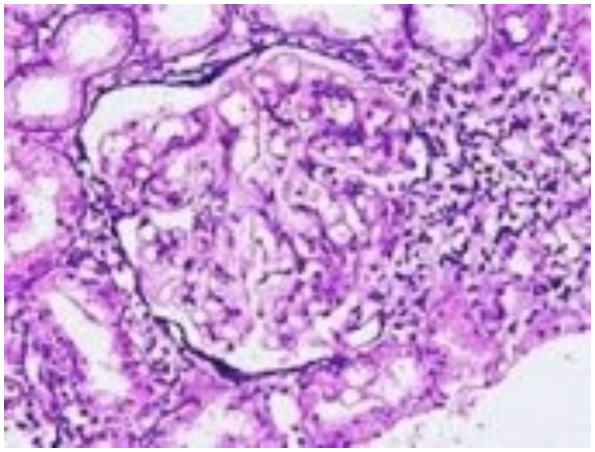
Why Interpretability ?

1) Verify that classifier works as expected

Wrong decisions can be costly and dangerous

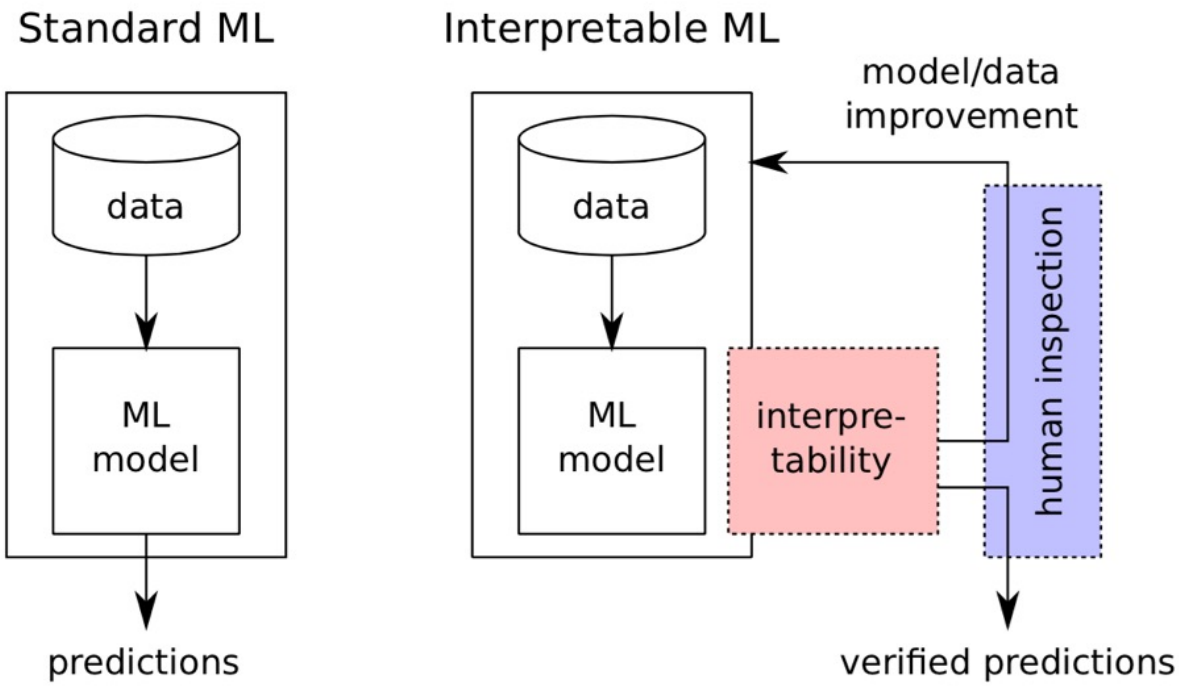
"Autonomous car crashes, because it wrongly recognizes …"

"AI medical diagnosis system misclassifies patient's disease …"



Why Interpretability ?

2) Understand weaknesses & improve classifier



Generalization error

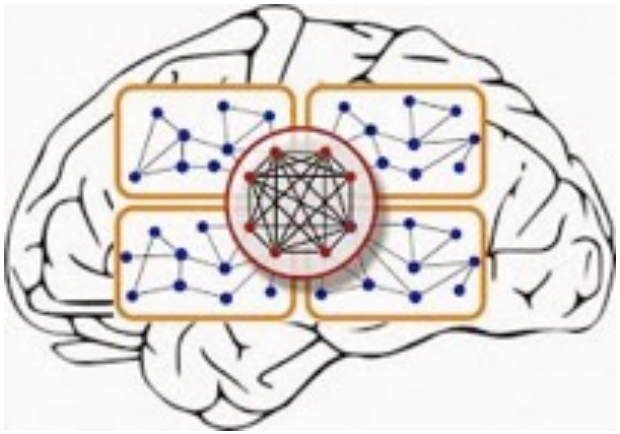
Generalization error + human experience

Why Interpretability ?

3) Learn new things from the learning machine

“It's not a human move. I've never seen a human play this move.” (Fan Hui)

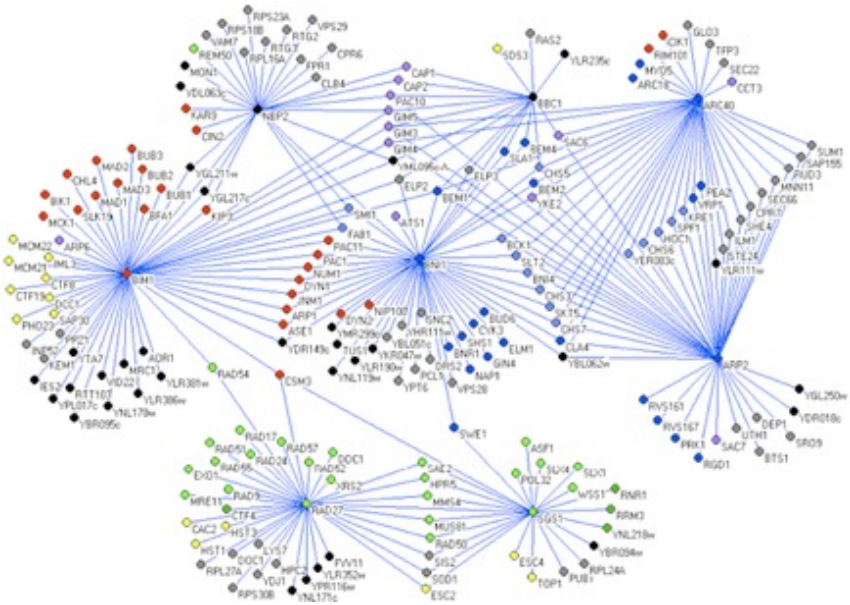
Old promise:
“Learn about the human brain.”

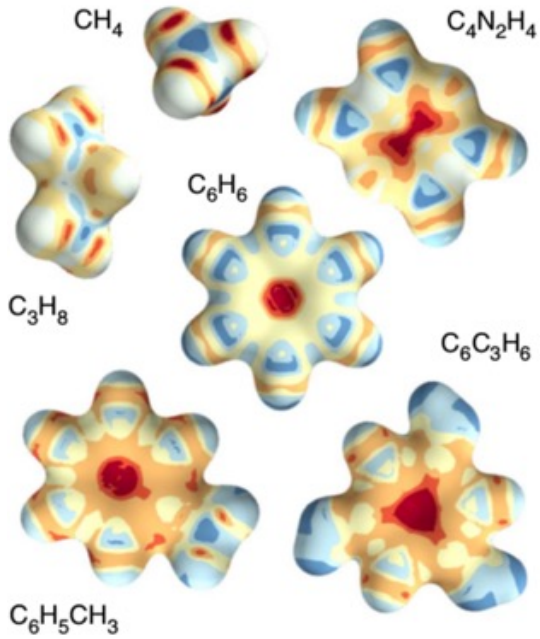


Why Interpretability ?

4) Interpretability in the sciences

Learn about the physical / biological / chemical mechanisms.
(e.g. find genes linked to cancer, identify binding sites …)





Why Interpretability ?

5) Compliance to legislation

European Union's new General Data Protection Regulation → "right to explanation"

Retain human decision in order to assign responsibility.

"With interpretability we can ensure that ML models work in compliance to proposed legislation."

ITU/WHO Focus Group on AI4Health

Focus Group on “Artificial Intelligence for Health” established by

ITU Workshop on Artificial Intelligence for Health
Geneva, Switzerland, 25 September 2018

More information about the group:
`https://www.itu.int/en/ITU-T/focusgroups/ai4h`

Dimensions of Interpretability

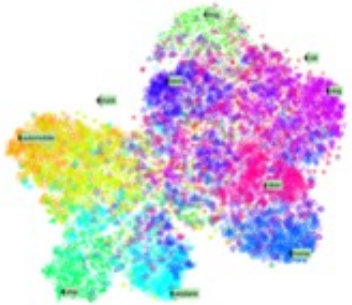
Different dimensions of "interpretability"

prediction

"Explain why a certain pattern x has been classified in a certain way f(x)."

model

"What would a pattern belonging to a certain category typically look like according to the model."



data

"Which dimensions of the data are most relevant for the task."

Dimensions of Interpretability

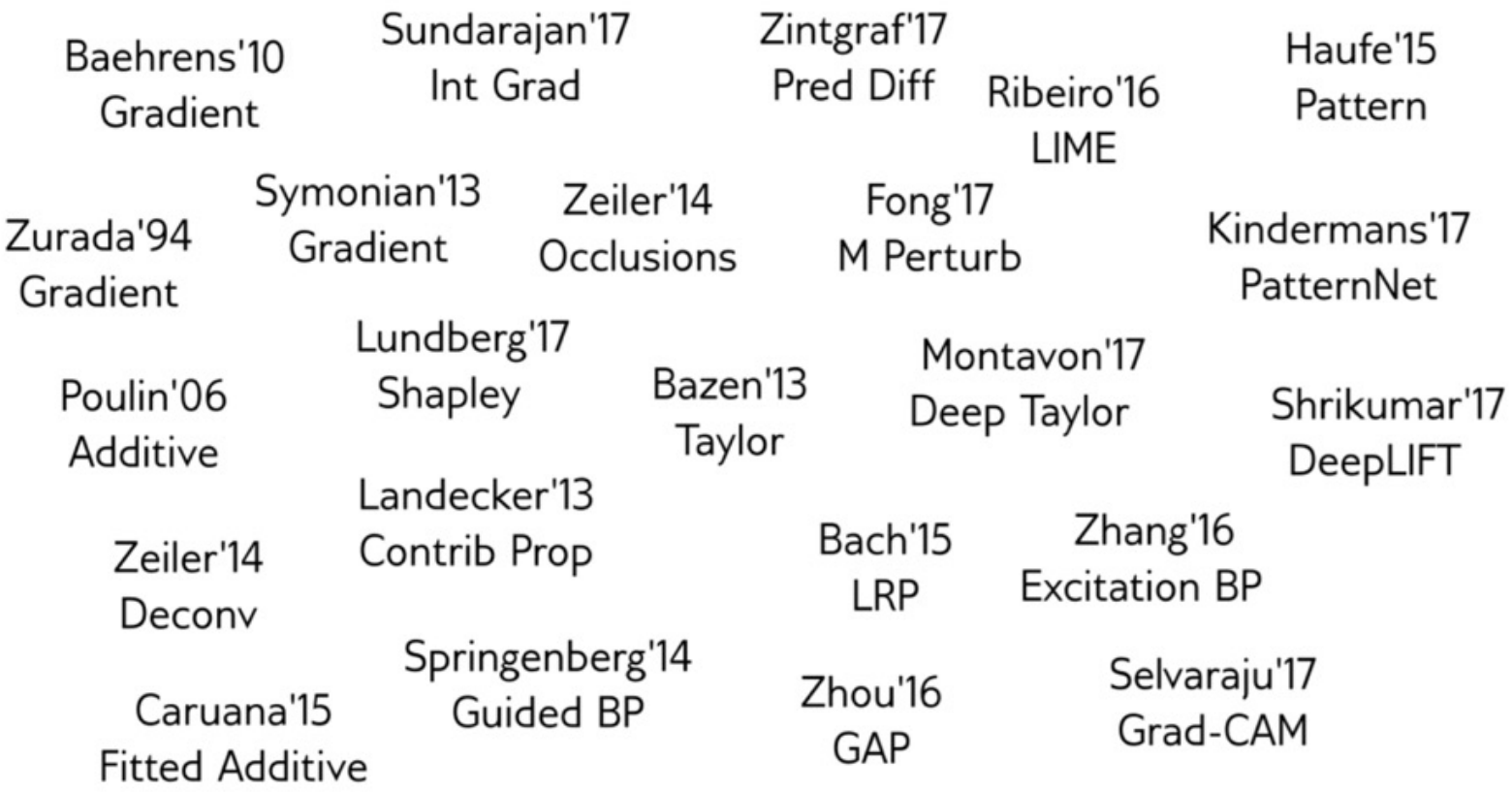
train interpretable model

suboptimal or biased due to assumptions (linearity, sparsity …)

vs.

train best model → interpret it

Dimensions of Interpretability



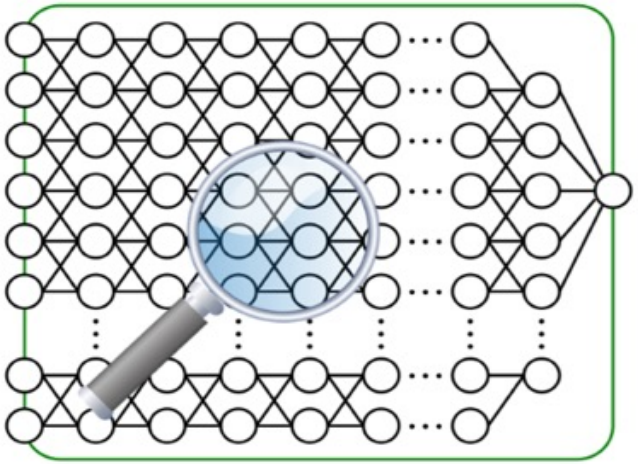
Question: Which one to choose ?

Tutorial on Interpretable Machine Learning

Part 2: Techniques of Interpretability

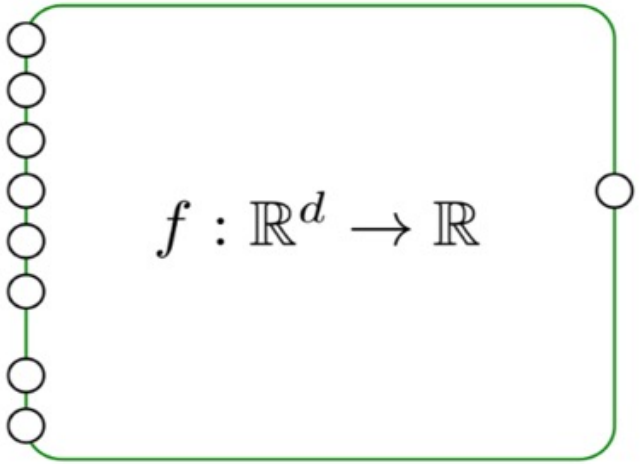
Techniques of Interpretation

mechanistic understanding



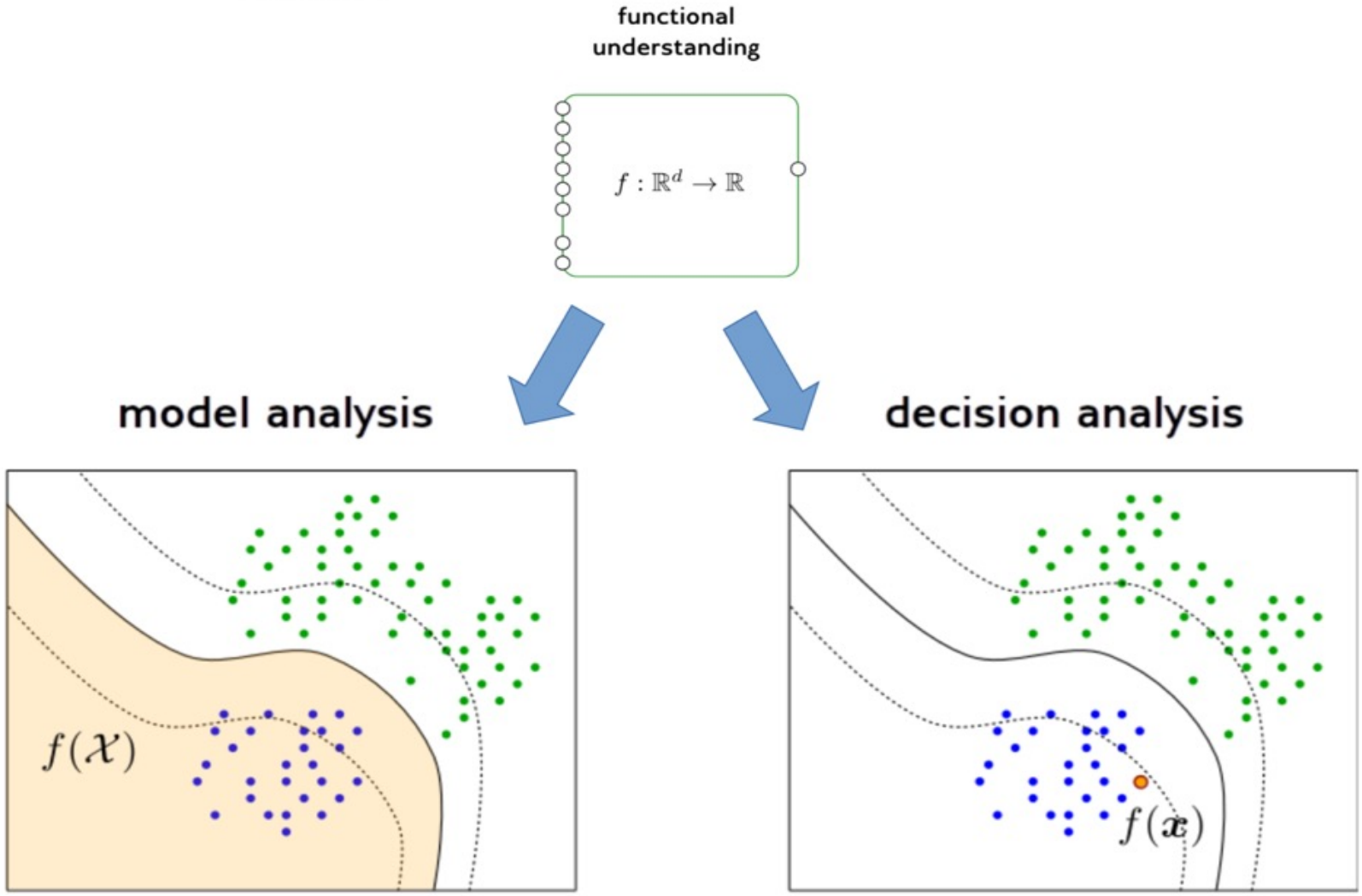
Understanding what mechanism the network uses to solve a problem or implement a function.

functional understanding



Understanding how the network relates the input to the output variables.

Techniques of Interpretation

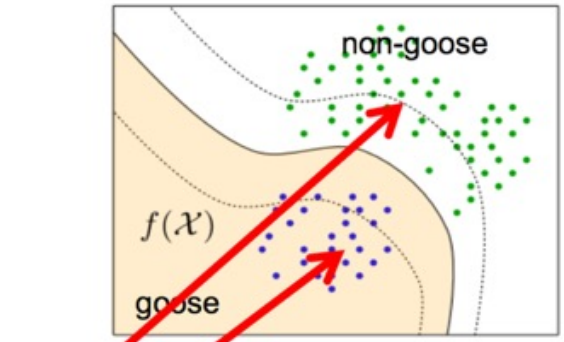


Model analysis

Interpreting the Model

Approach 1: Class Prototypes

"How does a goose typically look like according to the neural network?"



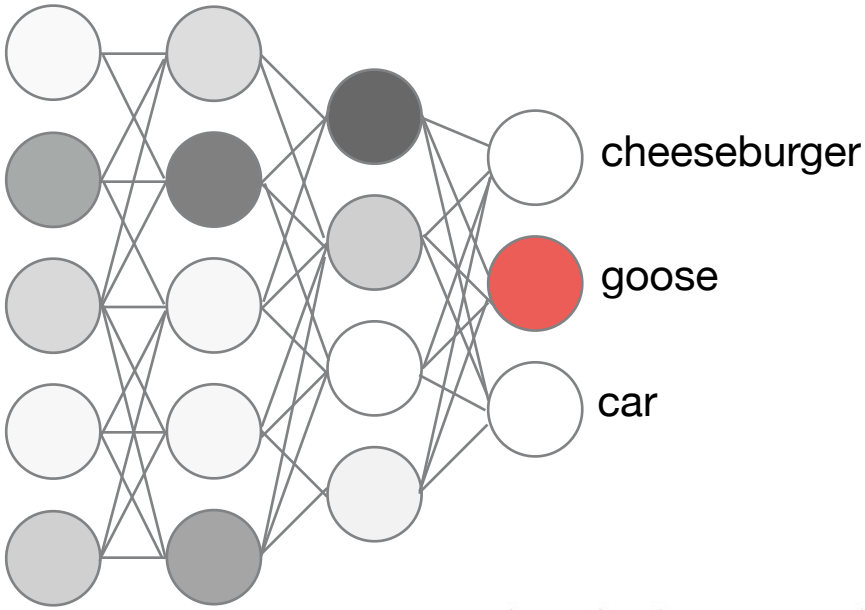
Class prototypes

$$\arg\max_x f(x) + \text{reg.}$$

Interpreting the Model

Activation Maximization

- *find prototypical example of a category*
- *find pattern maximizing activity of a neuron*



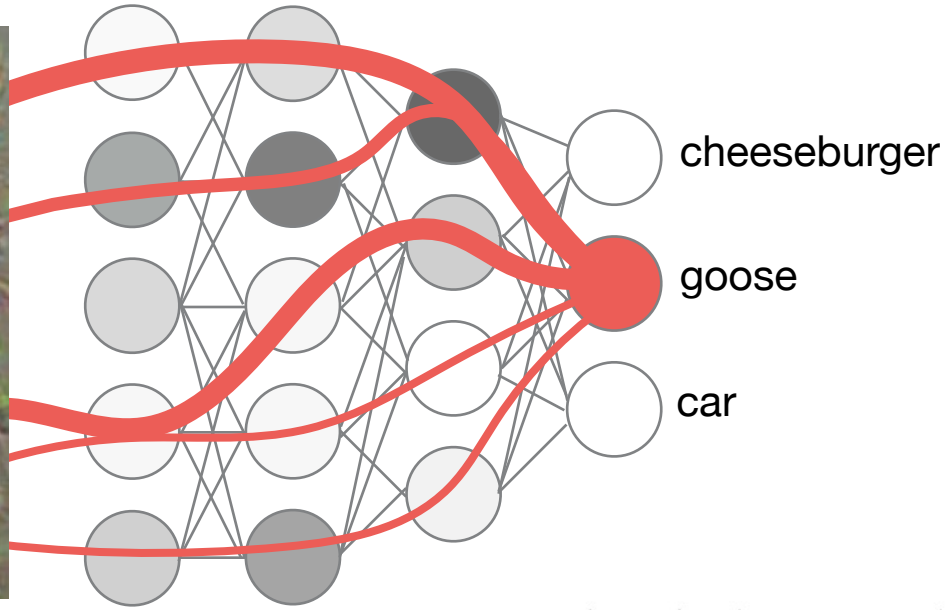
$$\max_{x \in \mathcal{X}} p_\theta(\omega_c \mid x) + \lambda \Omega(x)$$

Interpreting the Model

Activation Maximization

- *find prototypical example of a category*
- *find pattern maximizing activity of a neuron*

simple regularizer
(Simonyan et al. 2013)



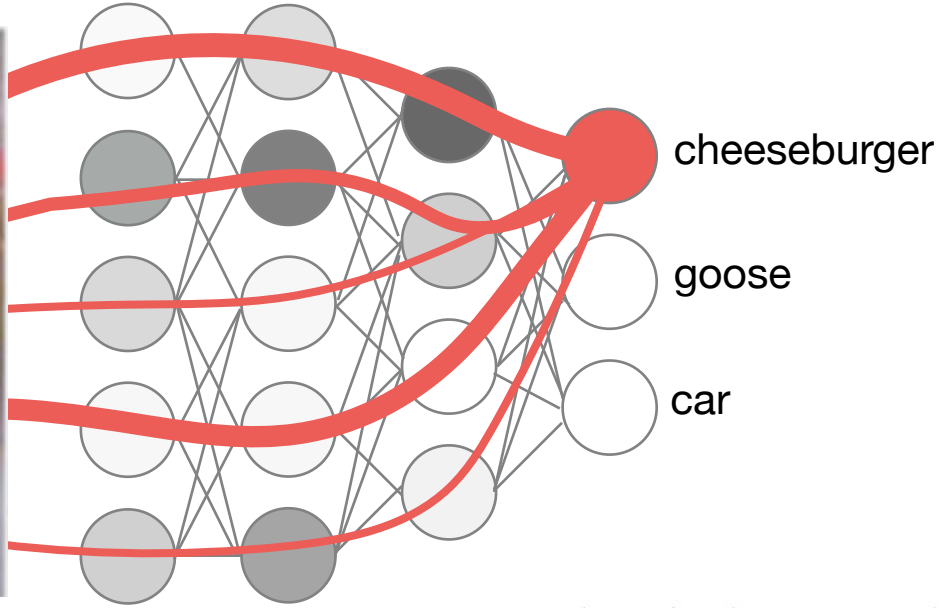
$$\max_{x \in \mathcal{X}} p_\theta(\omega_c \mid x) + \lambda \Omega(x)$$

Interpreting the Model

Activation Maximization

- *find prototypical example of a category*
- *find pattern maximizing activity of a neuron*

complex regularizer (Nguyen et al. 2016)

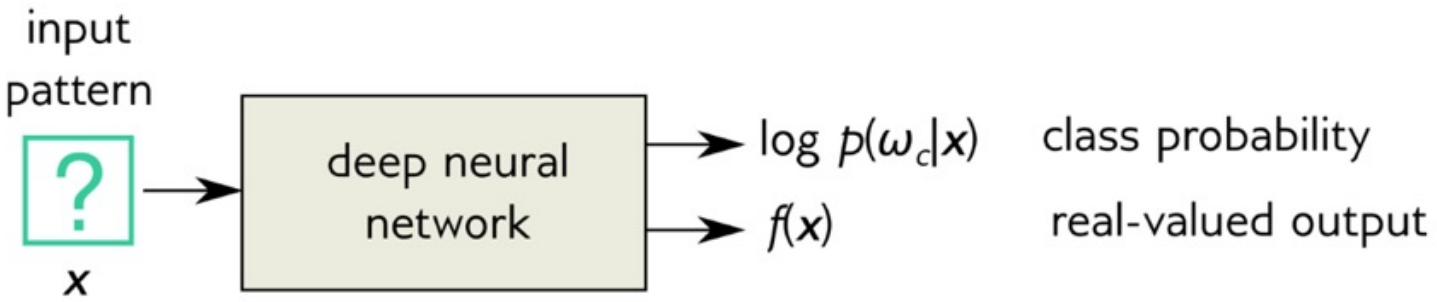


$$\max_{x \in \mathcal{X}} p_\theta(\omega_c \mid x) + \lambda \Omega(x)$$

Interpreting the Model

Activation Maximization

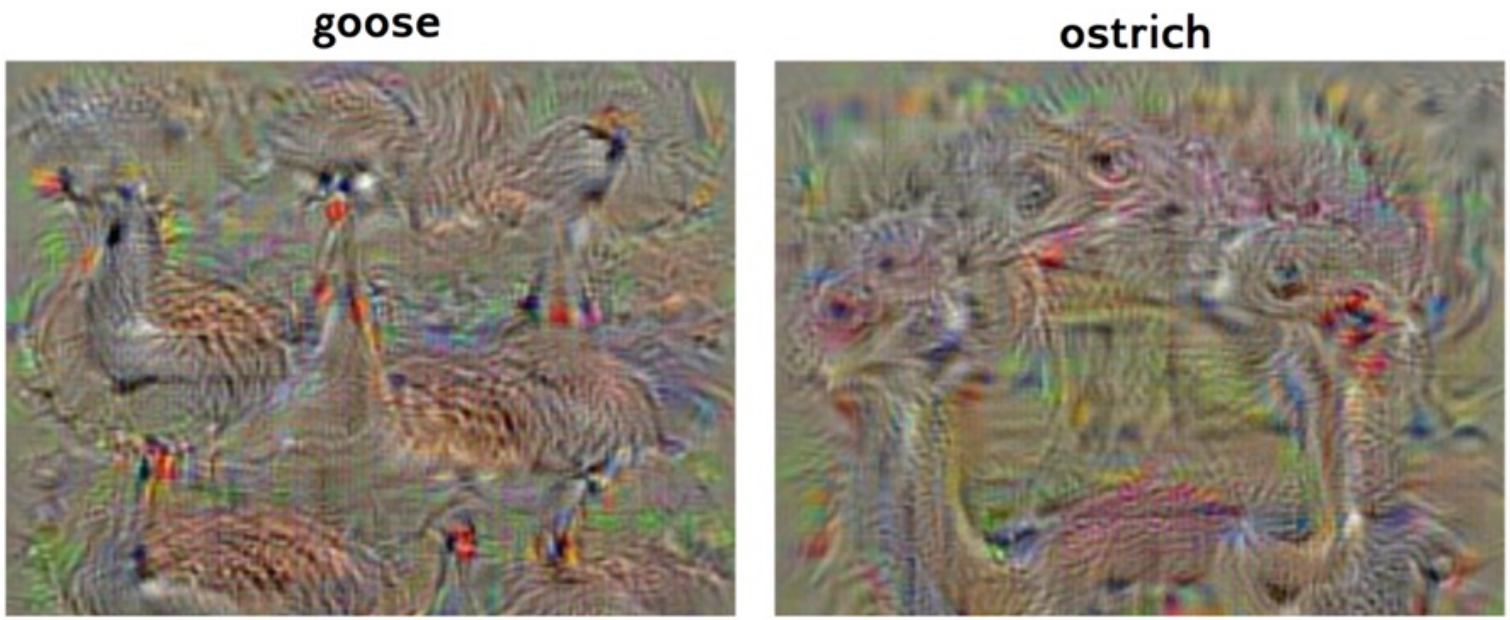
Let us interpret a concept predicted by a deep neural net (e.g. a class, or a real-valued quantity):



Examples:

- Creating a class prototype: $\max_{x \in \mathcal{X}} \log p(\omega_c|x)$.
- Synthesizing an extreme case: $\max_{x \in \mathcal{X}} f(x)$.

Interpreting the Model



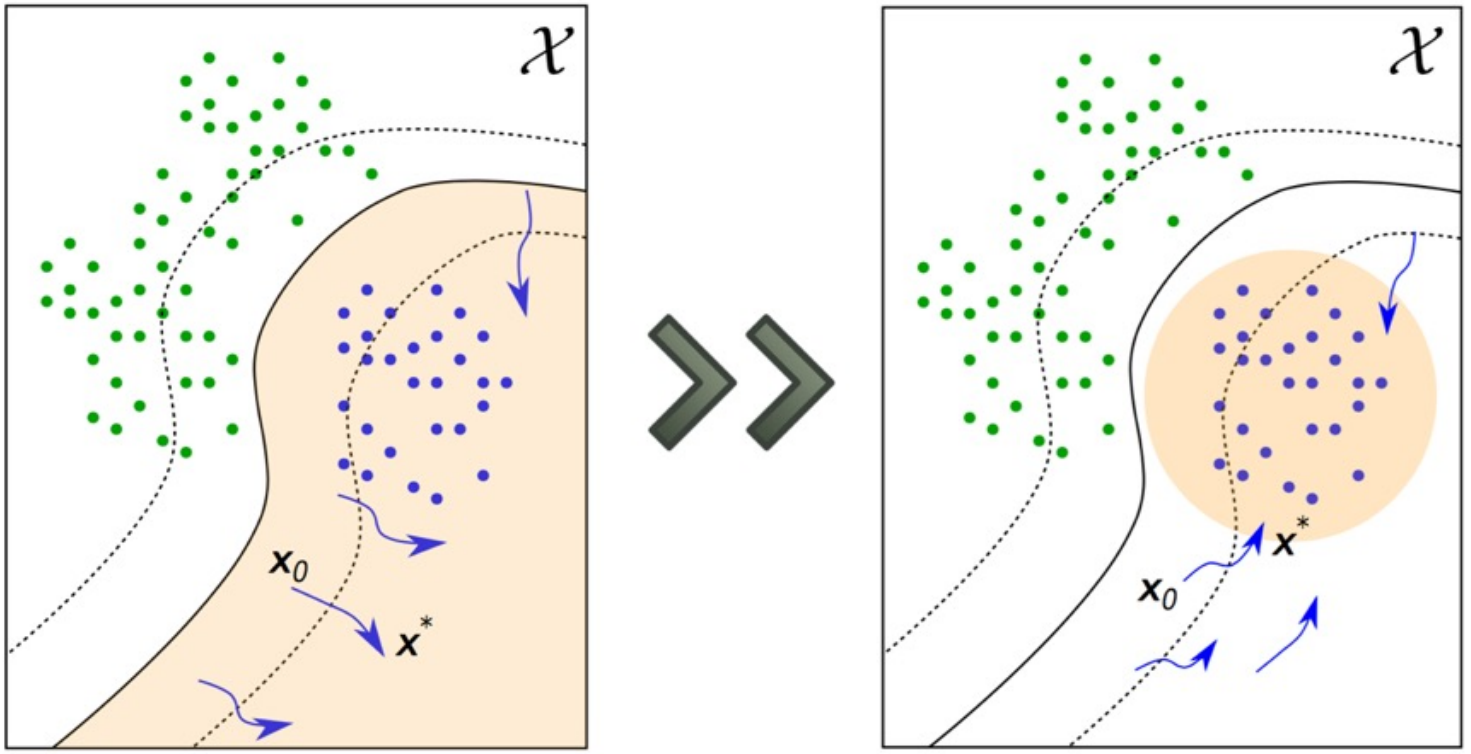
Images from **Simonyan et al. 2013** "Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps"

Observations:

- AM builds typical patterns for these classes (e.g. beaks, legs).
- Unrelated background objects are not present in the image.

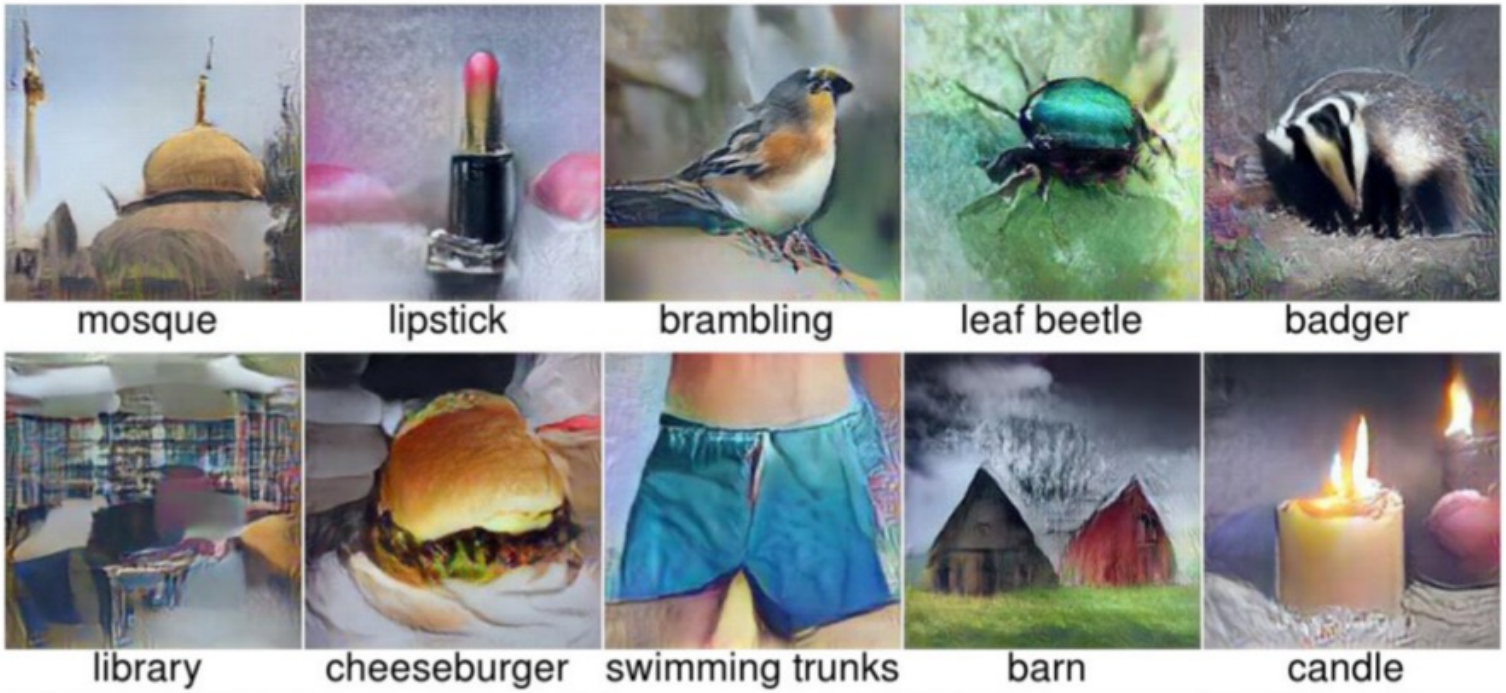
Enhancing Activation Maximization

Find the input pattern that maximizes class probability. → Find the most likely input pattern for a given class.



Enhancing Activation Maximization

Images from **Nguyen et al. 2016.** "Synthesizing the preferred inputs for neurons in neural networks via deep generator networks"

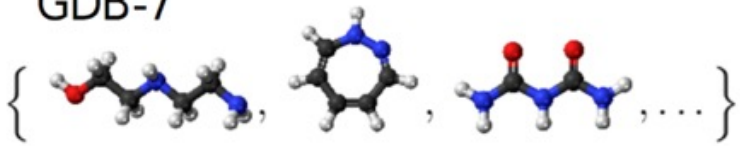


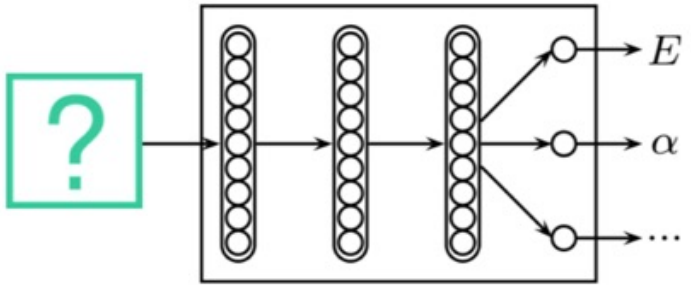
Observation: Connecting AM to the data distribution leads to more realistic and more interpretable images.

Application beyond Image Domain

Finding a prototype:

GDB-7





Question: How does a molecule with properties XYZ look like ?

Limitations of Global Interpretations

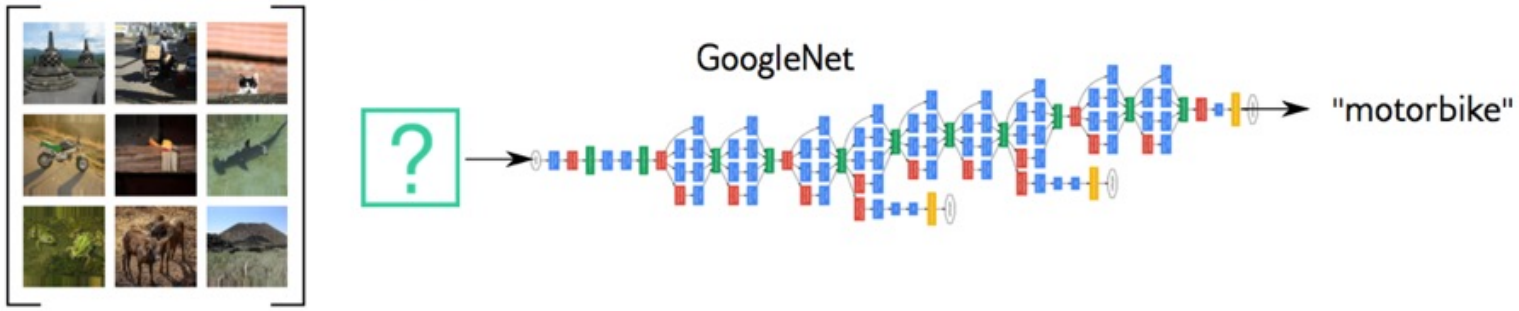
Question: Below are some images of motorbikes. What would be the best prototype to interpret the class "motorbike"?

Observations:

- Summarizing a concept or category like "motorbike" into a single image can be difficult (e.g. different views or colors).
- A good interpretation would grow as large as the diversity of the concept to interpret.

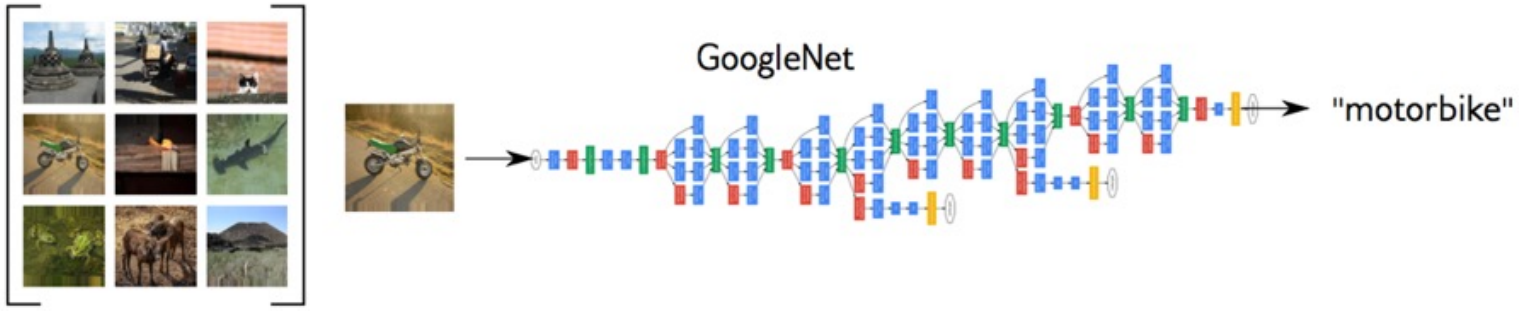
Need for Individual Explanations

Finding a prototype:



Question: How does a "motorbike" typically look like?

Individual explanation:



Question: Why is *this* example classified as a motorbike?

Need for Individual Explanations

Personalized medicine: Extracting the relevant information about a medical condition for a *given* patient at a *given* time.

Each case is unique and needs its own explanation.

Need for Individual Explanations

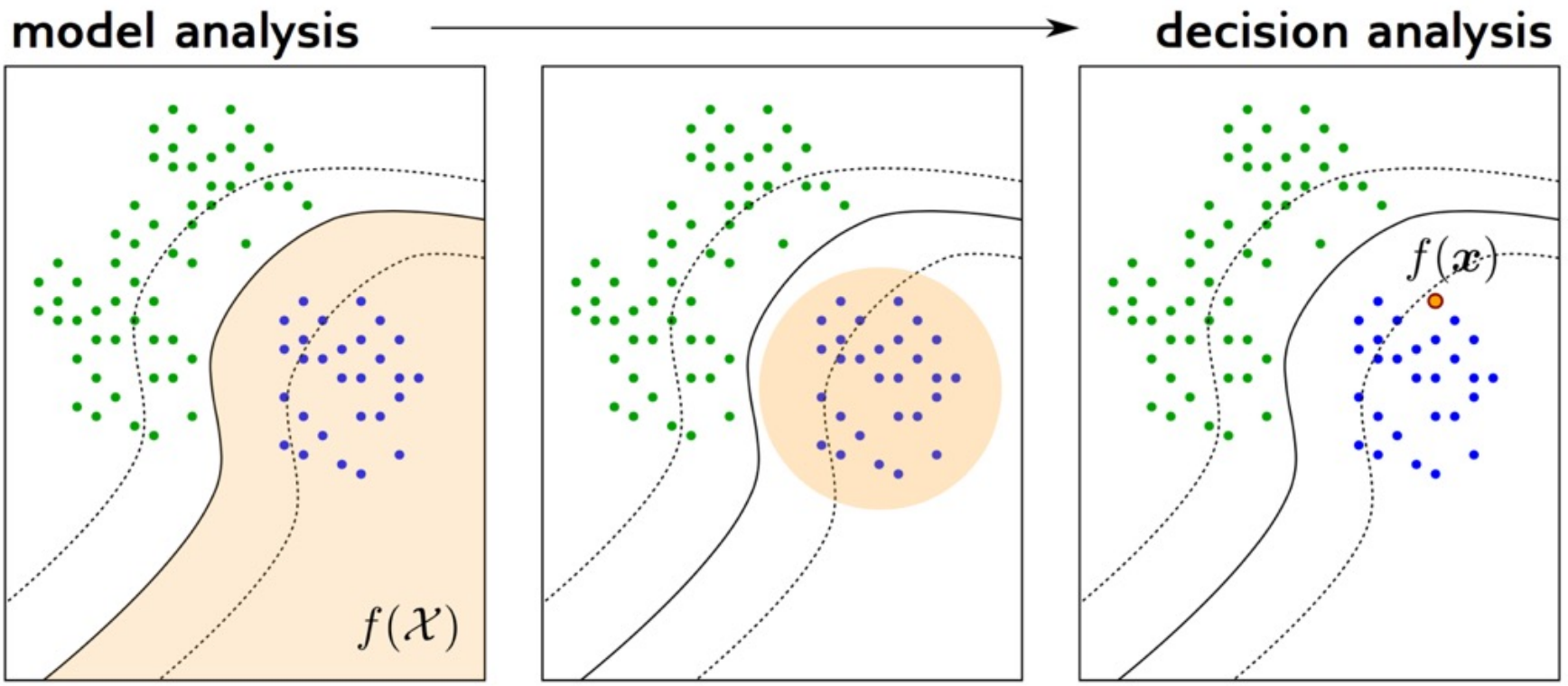
Personalized medicine: Extracting the relevant information about a medical condition for a *given* patient at a *given* time.

Each case is unique and needs its own explanation.

Population view: Which symptoms are most common for the disease

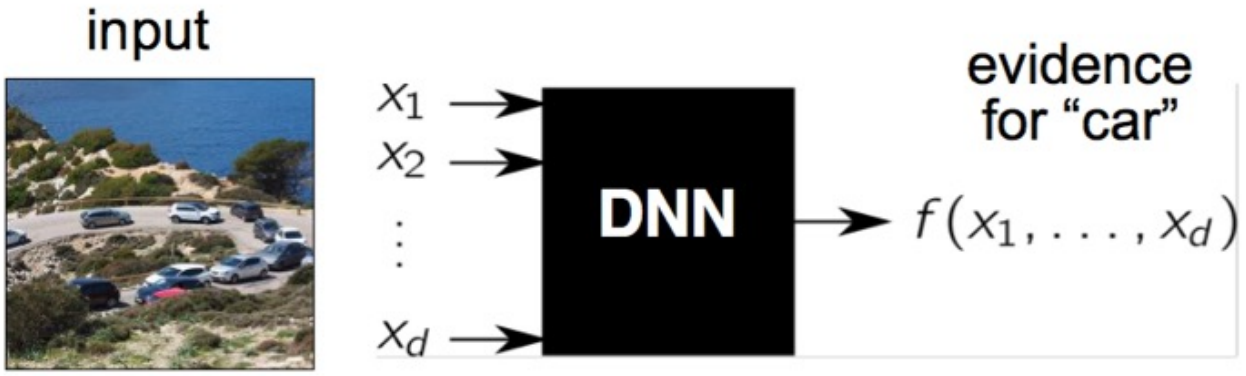
Both aspects can be important depending on who you are (FDA, doctor, patient).

Making Deep Neural Nets Transparent



Decision analysis

Decision Analysis: Sensitivity Analysis



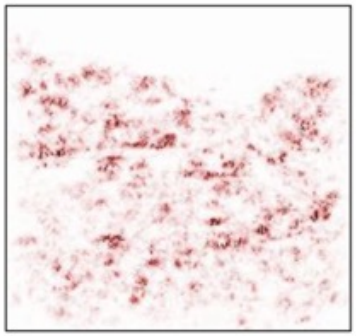
Sensitivity analysis: The relevance of input feature *i* is given by the squared partial derivative:

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

Decision Analysis: Sensitivity Analysis

Sensitivity analysis:

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



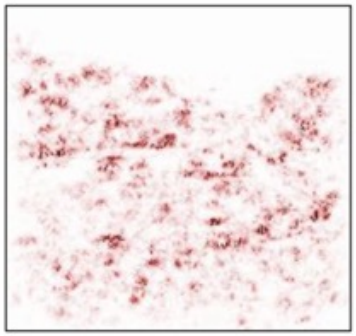
Problem: sensitivity analysis does not highlight cars

highlights parts, which (when changed) increase or decrease the prediction for “car”.

Decision Analysis: Sensitivity Analysis

Sensitivity analysis:

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



Problem: sensitivity analysis does not highlight cars

highlights parts, which (when changed) increase or decrease the prediction for “car”.

Observation:

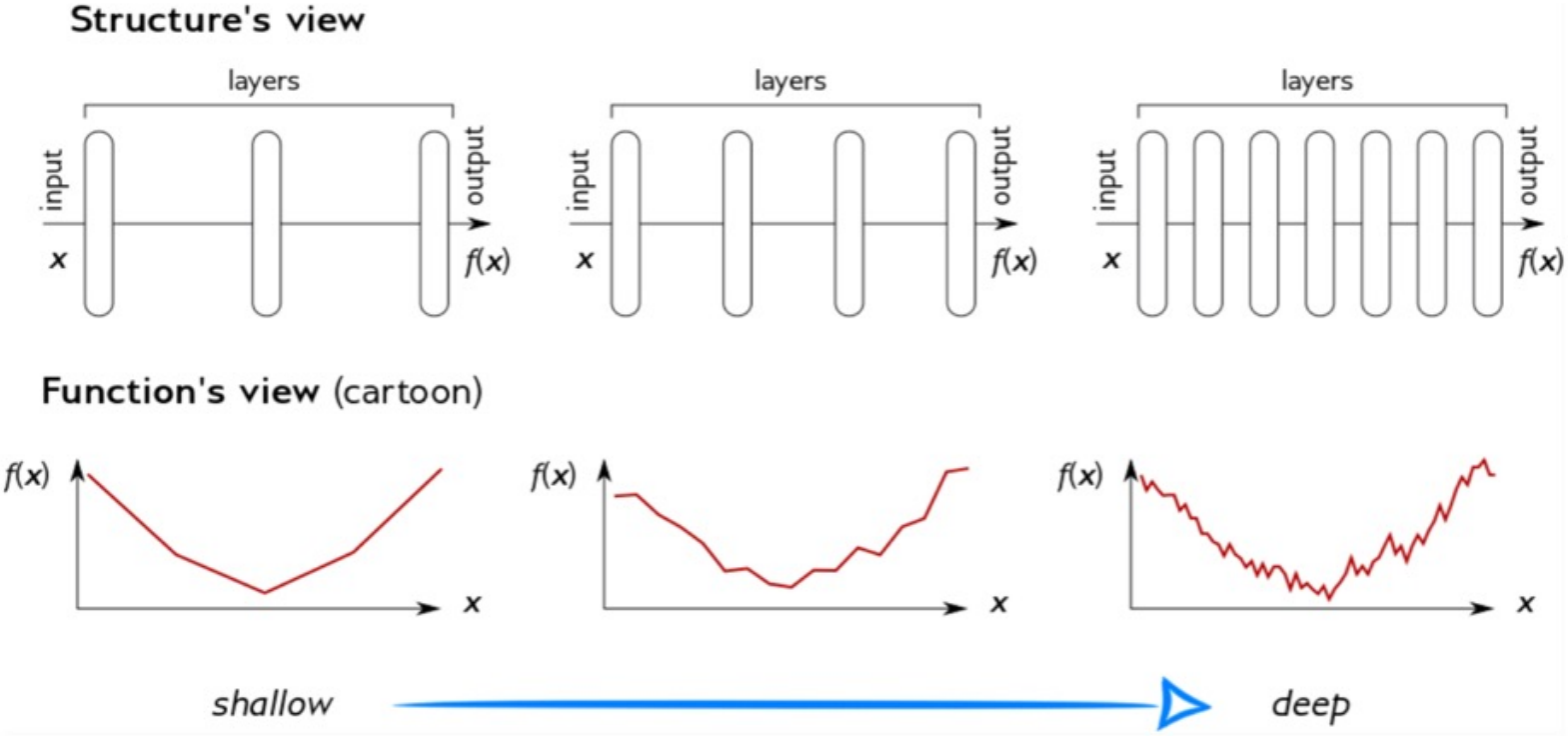
$$\sum_{i=1}^{d} \left(\frac{\partial f}{\partial x_i}\right)^2 = \|\nabla_x f\|^2$$

Sensitivity analysis explains a *variation* of the function, not the function value itself.

Decision Analysis: Sensitivity Analysis

Shattered Gradient Problem

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



Layer-wise Relevance Propagation (LRP)

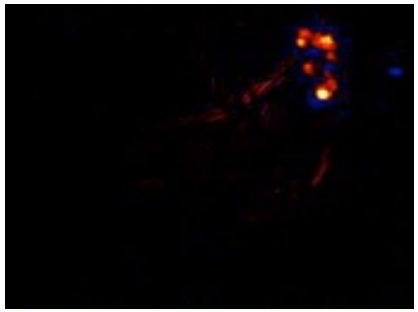
Decision Analysis: LRP

input x

Black Box

Rooster

prediction $f(x)$



heatmap

Idea: Decompose function

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

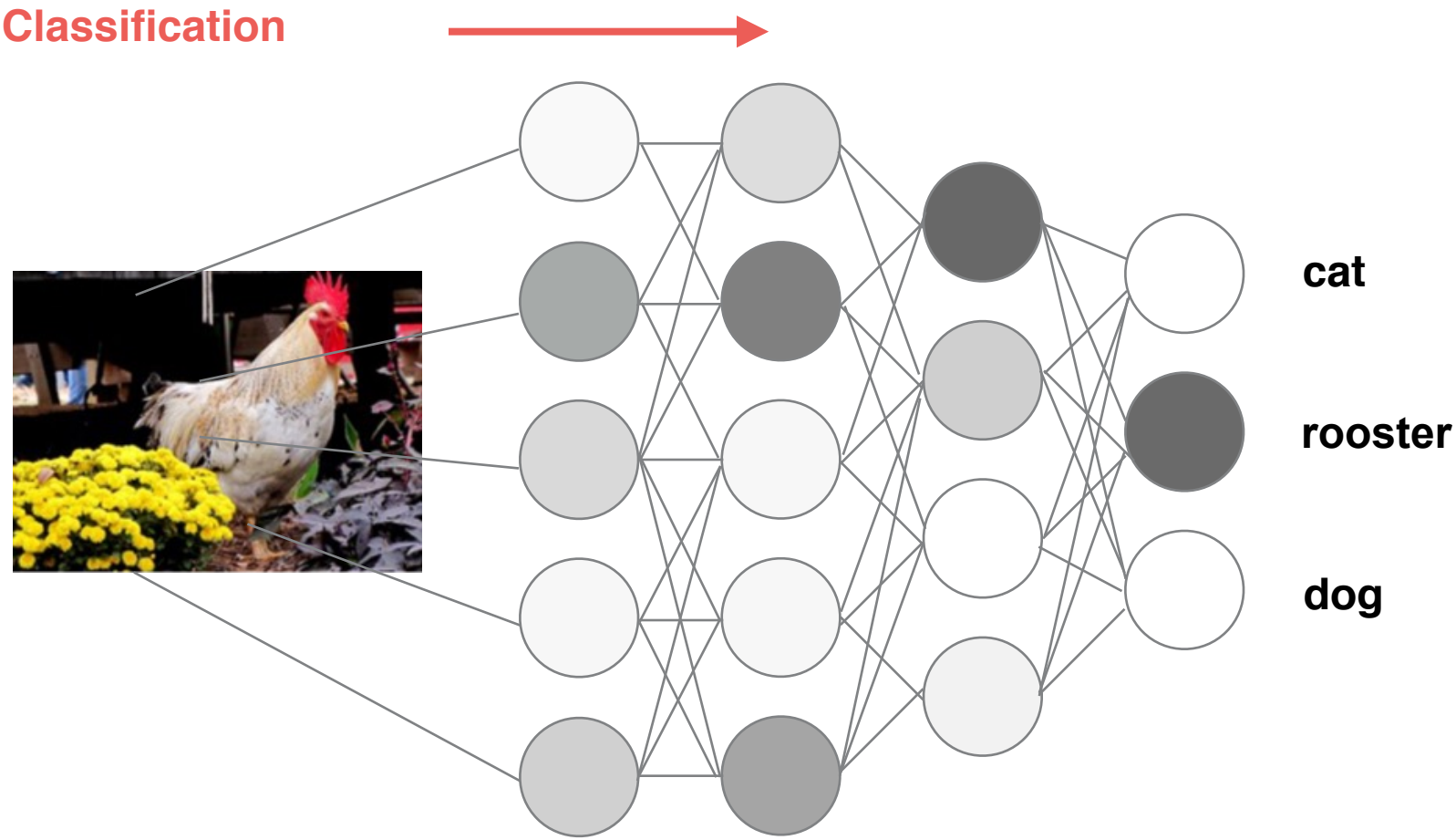
Explain prediction

(how much each pixel contributes to prediction)

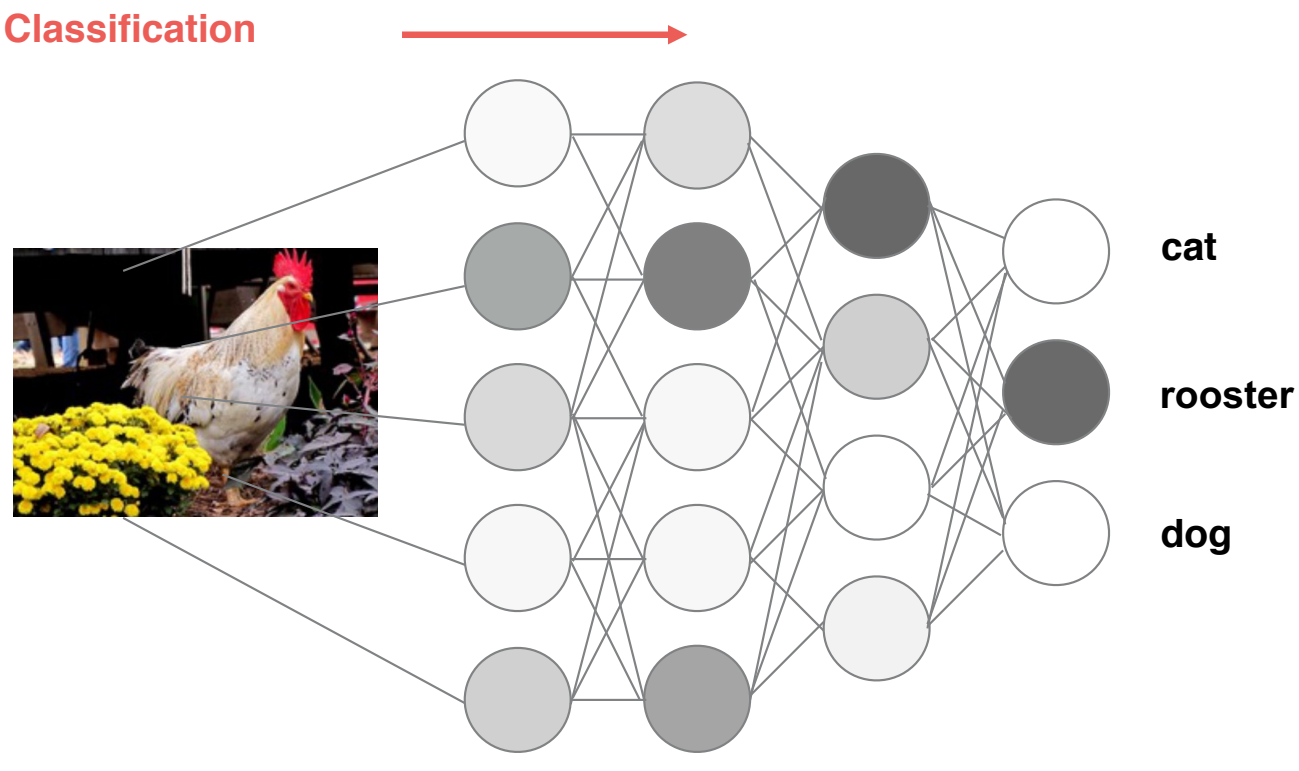
Layer-wise Relevance Propagation (LRP)
(Bach et al., PLOS ONE, 2015)

Explain prediction itself
(not the change)

Decision Analysis: LRP



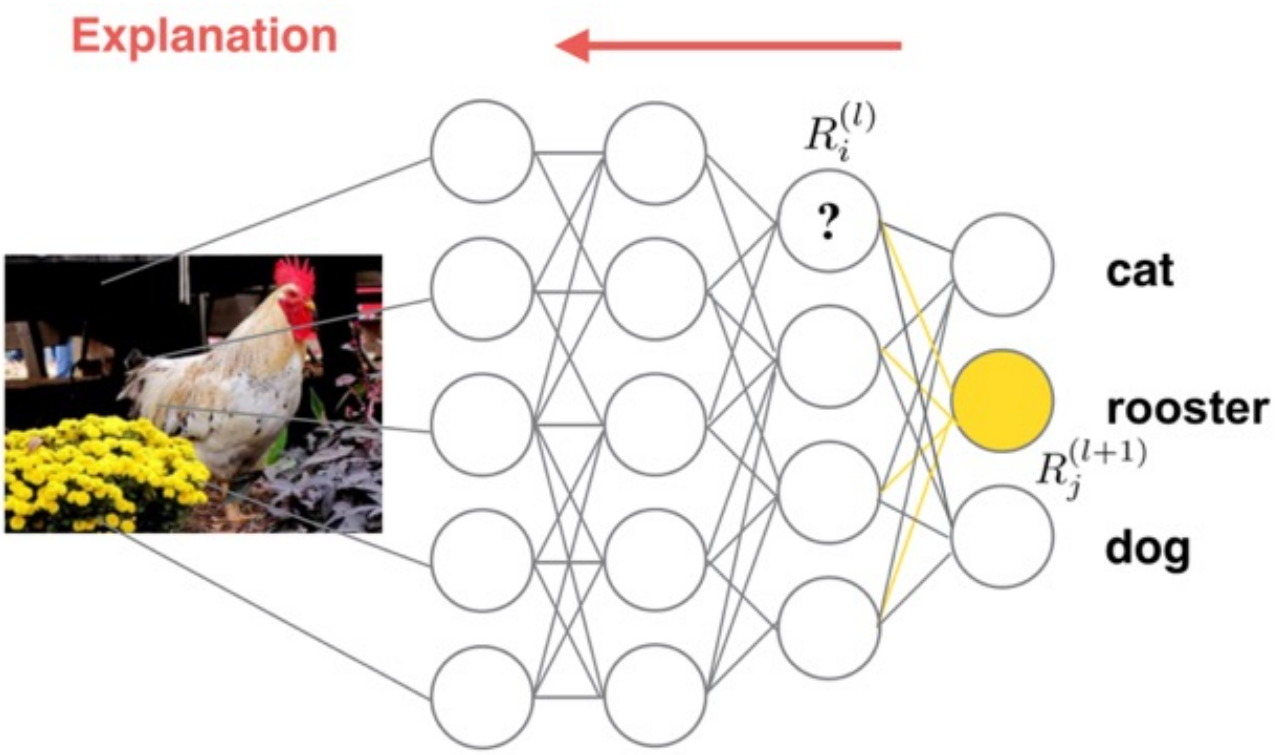
Decision Analysis: LRP



What makes this image a “rooster image” ?

Idea: Redistribute the evidence for class rooster back to image space.

Decision Analysis: LRP

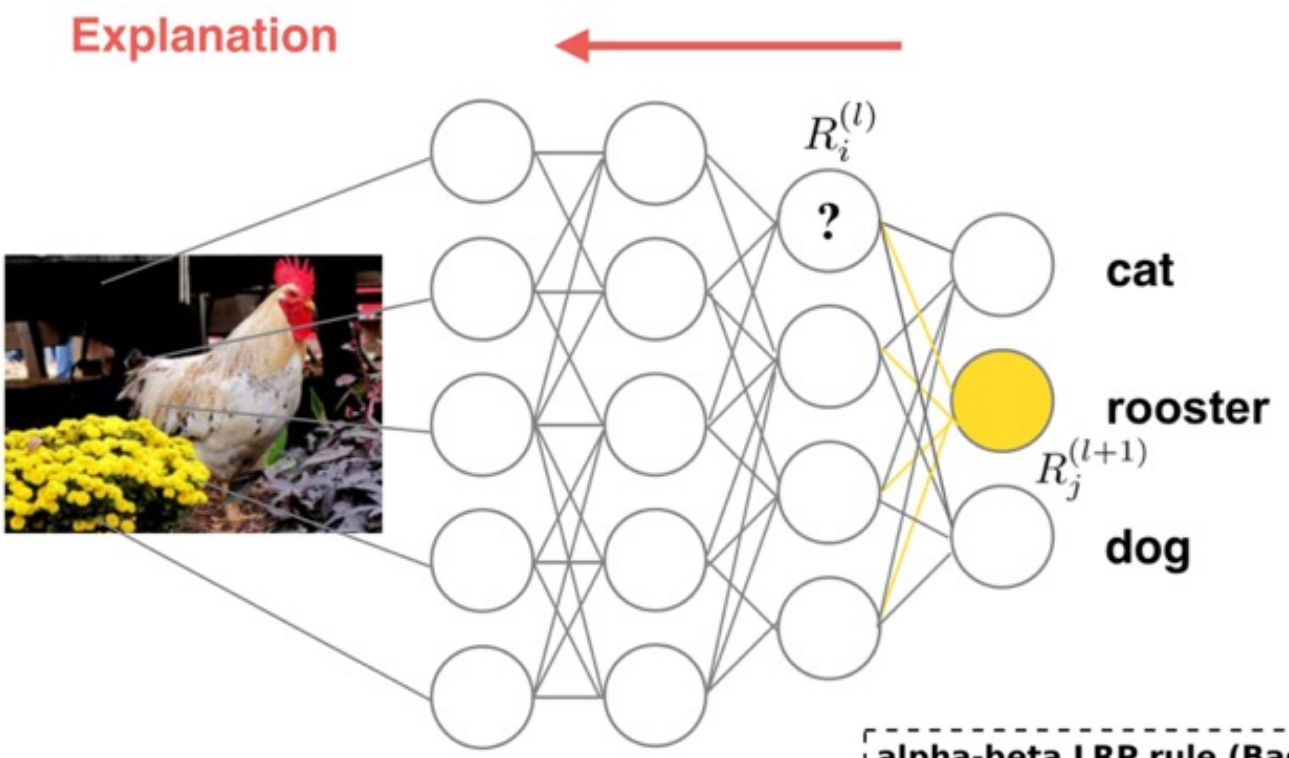


Simple LRP rule (Bach et al. 2015)

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

Every neuron gets its "share" of the redistributed relevance

Decision Analysis: LRP



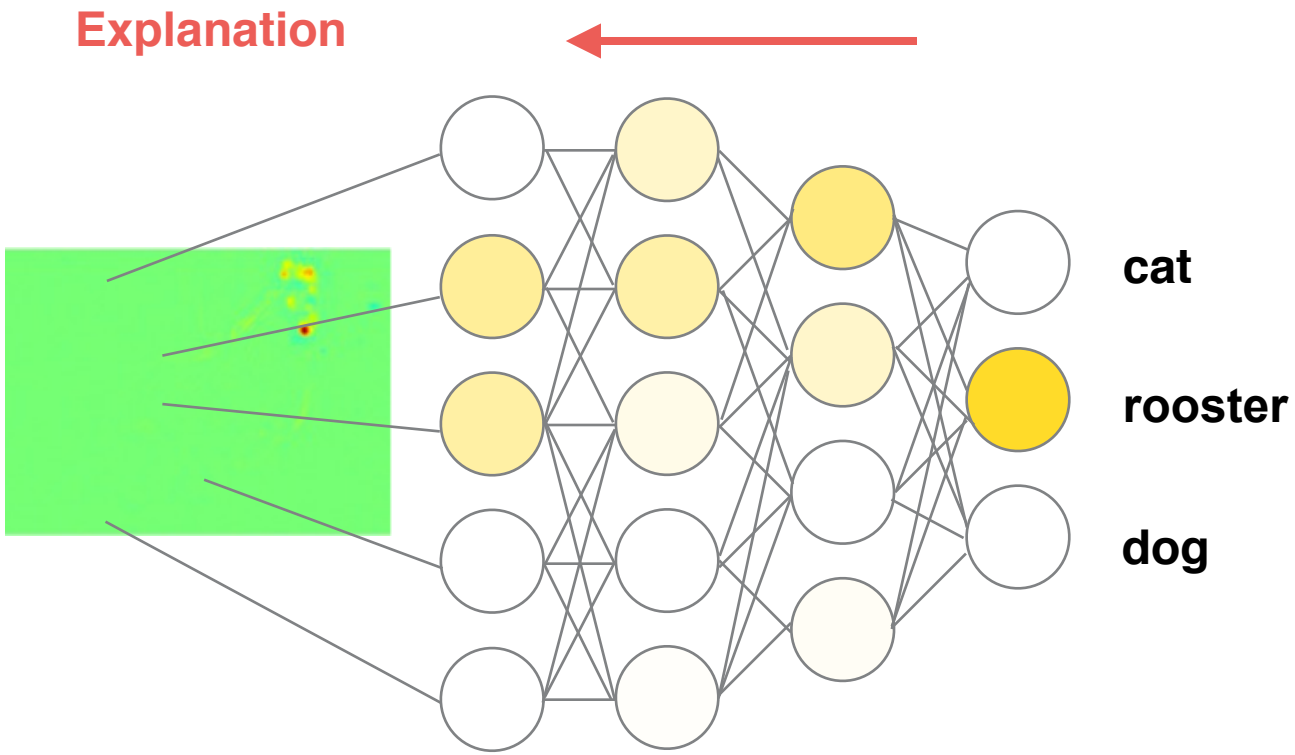
Theoretical interpretation
Deep Taylor Decomposition
(Montavon et al., 2017)
(no gradient shattering)

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

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

where $\alpha + \beta = 1$

Decision Analysis: LRP

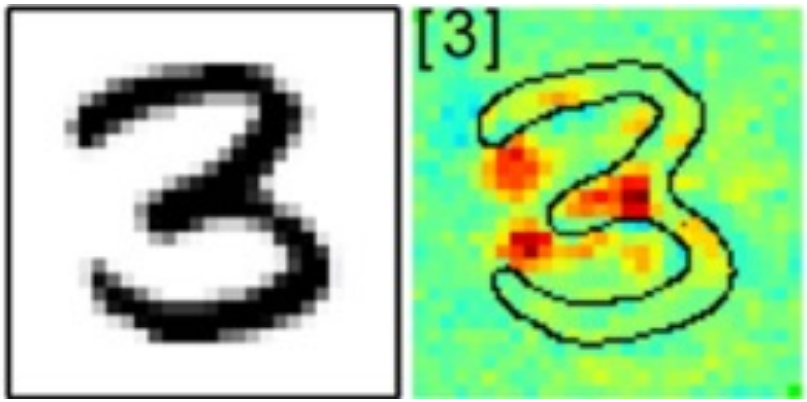


Layer-wise relevance conservation

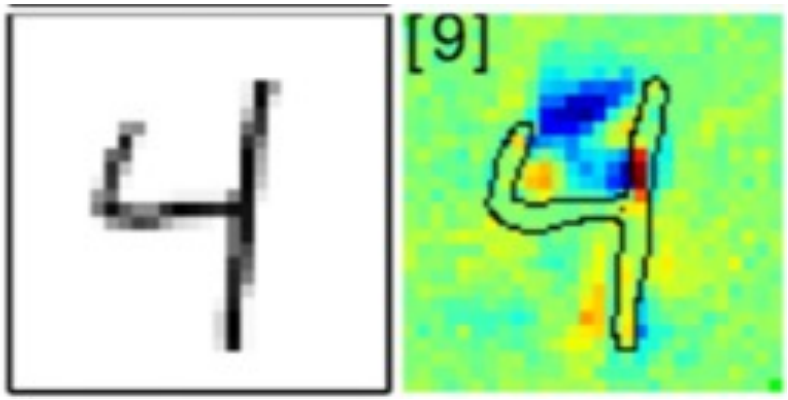
$$\sum_i R_i = \ldots = \sum_i R_i^{(l)} = \sum_j R_j^{(l+1)} = \ldots = f(x)$$

Decision Analysis: LRP

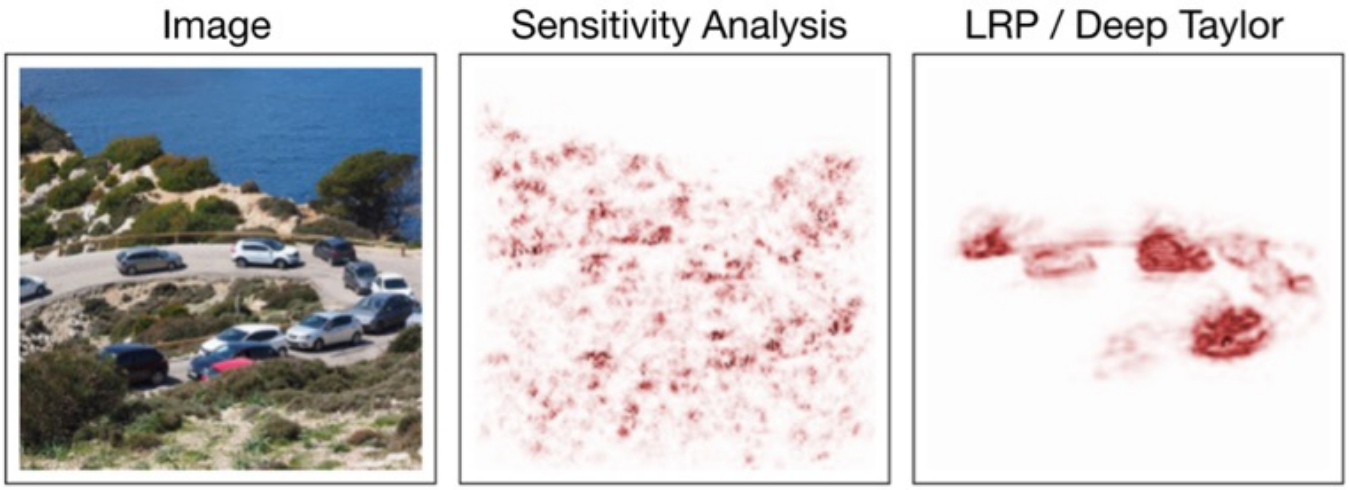
Heatmap of prediction “3”



Heatmap of prediction “9”



Decision Analysis: LRP



Explains what influences prediction “cars”.

Explains prediction “cars” as is.

Slope decomposition

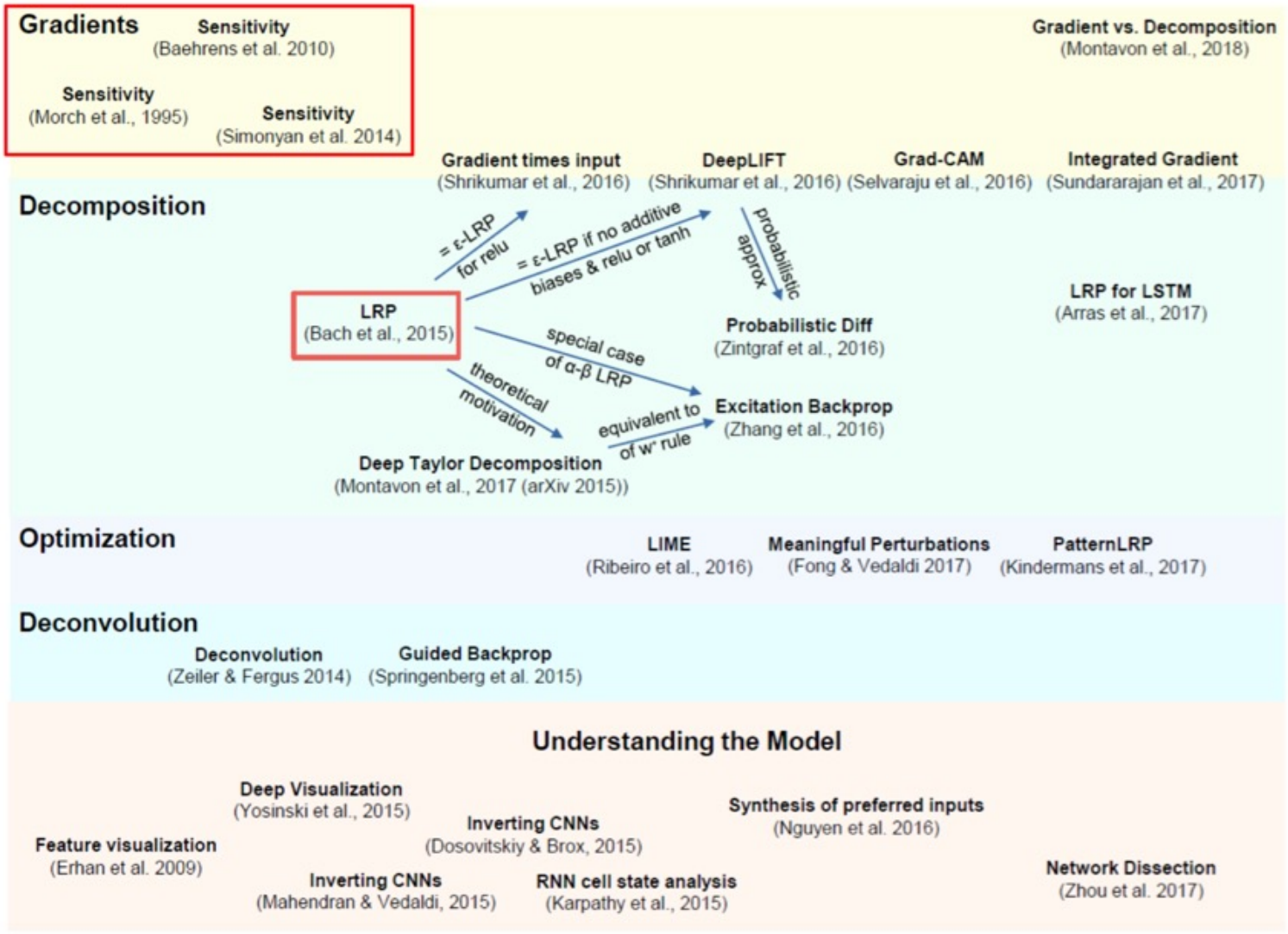
$$\sum_i R_i = \|\nabla_x f\|^2$$

Value decomposition

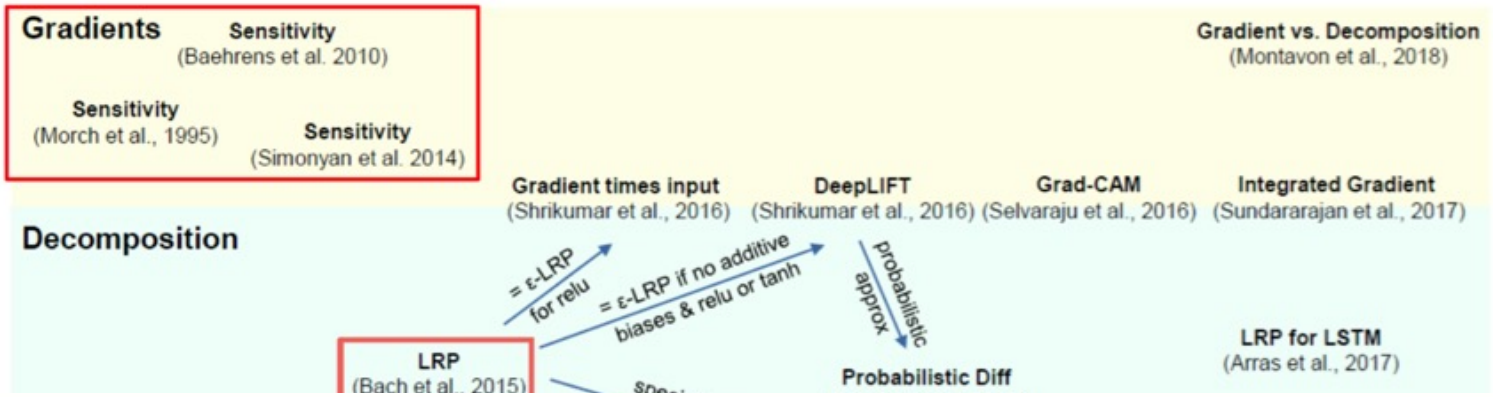
$$\sum_i R_i = f(\boldsymbol{x})$$

More information
(Montavon et al., 2017 & 2018)

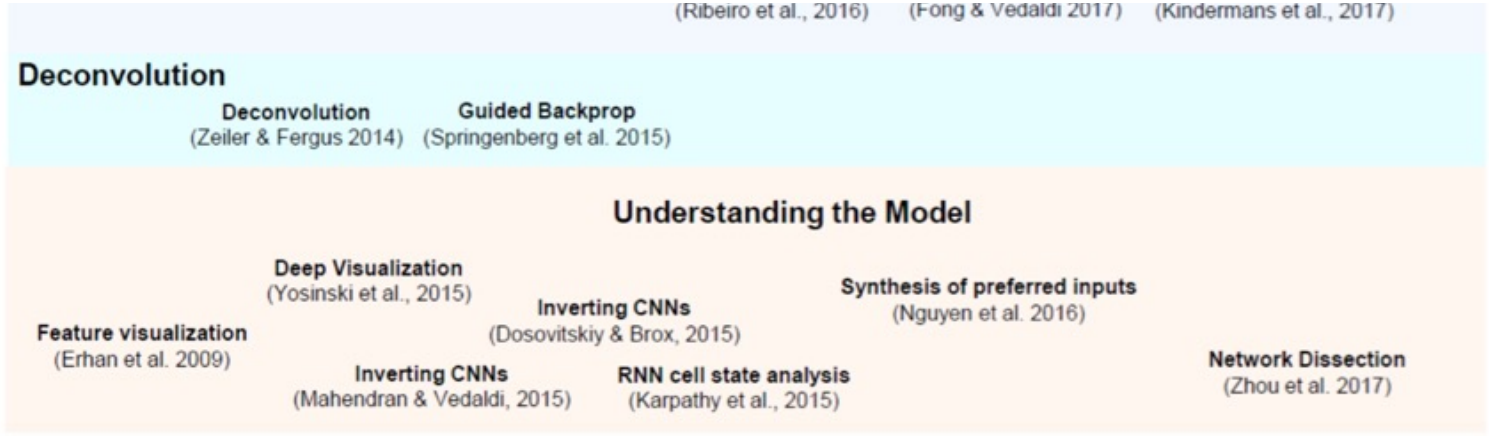
Other Explanation Methods



Other Explanation Methods

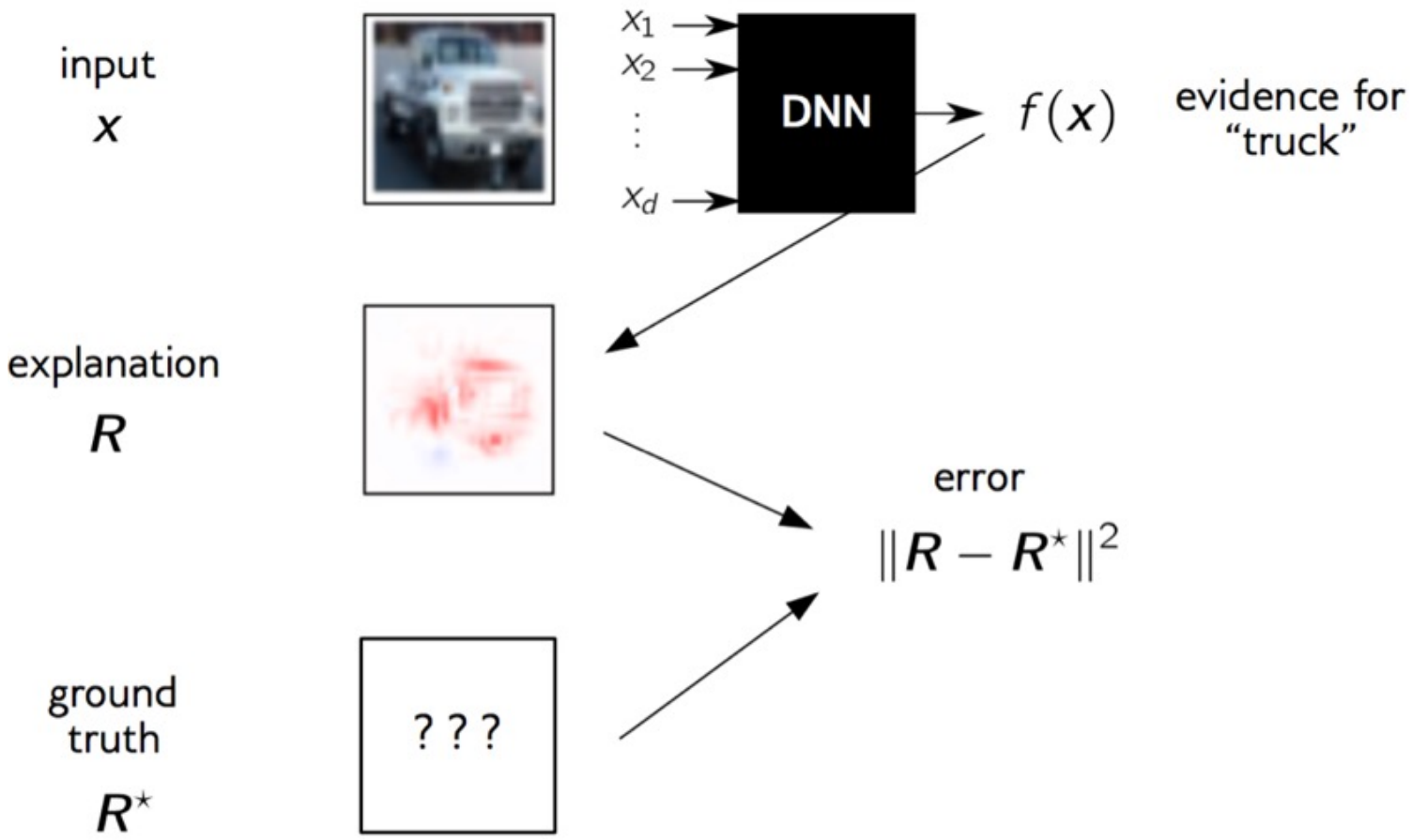


Question: Which one to choose ?



Axiomatic approach to interpretability

First Attempt: Distance to Ground Truth

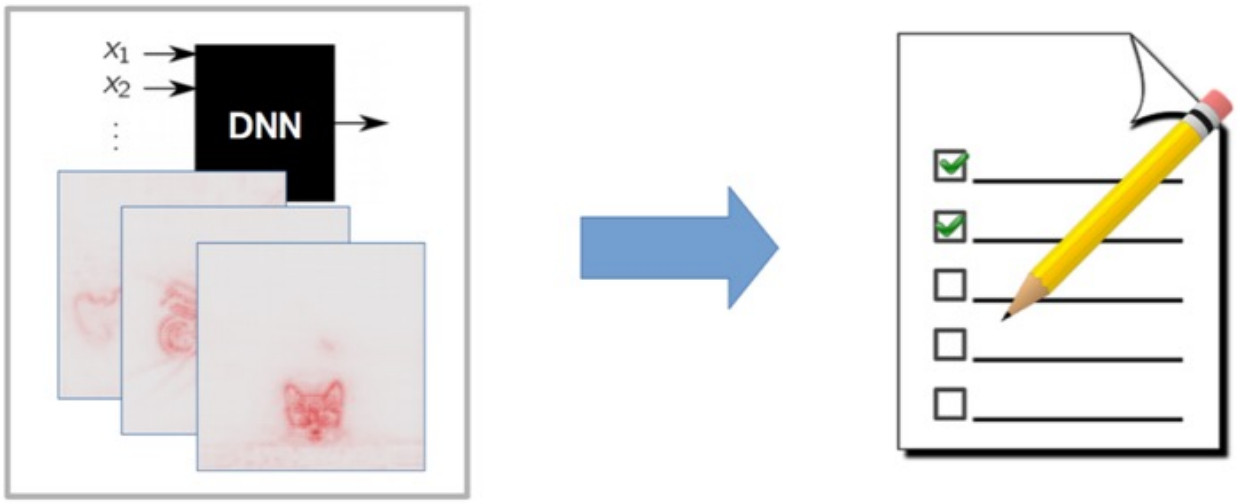


Axiomatic Approach to Interpretability

Idea: Evaluate the explanation technique axiomatically, i.e. it must pass a number of predefined "unit tests".

[Sun'11, Bach'15, Montavon'17, Samek'17, Sundarajan'17, Kindermans'17, Montavon'18].

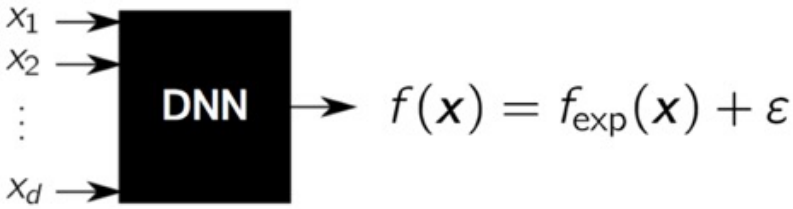
explanation technique



Axiomatic Approach to Interpretability

Properties 1-2: Conservation and Positivity

[Montavon'17, see also Sun'11, Landecker'13, Bach'15]



explanation

$R_1, \ldots, R_d$

Conservation: Total attribution on the input features should be proportional to the amount of (explainable) evidence at the output.

$$\sum_{p=1}^{d} R_p = f_{\text{exp}}(x)$$

Positivity: If the neural network is certain about its prediction, input features are either relevant (positive) or irrelevant (zero).

$$\forall_{p=1}^{d} : \; R_p \geq 0$$

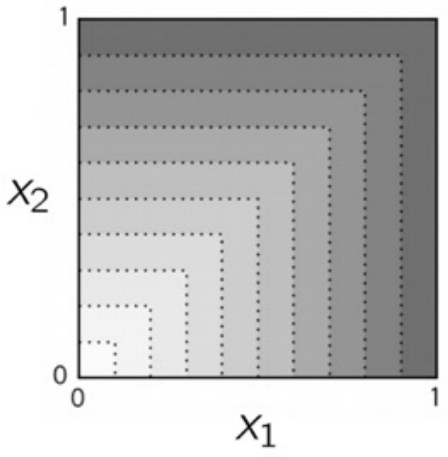
Axiomatic Approach to Interpretability

Property 3: Continuity [Montavon'18]

If two inputs are the almost the same, and the prediction is also almost the same, then the explanation should also be almost the same.

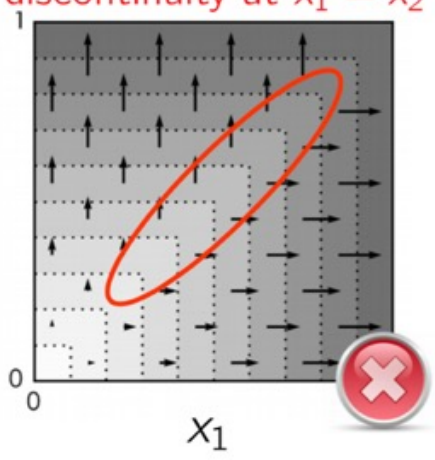
Example:

$f(\boldsymbol{x}) = \max(x_1, x_2)$

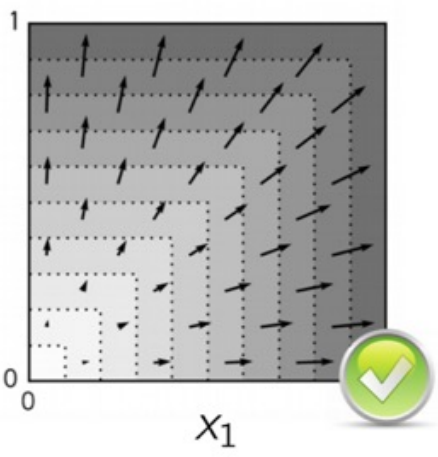


Method 1

discontinuity at $x_1 = x_2$

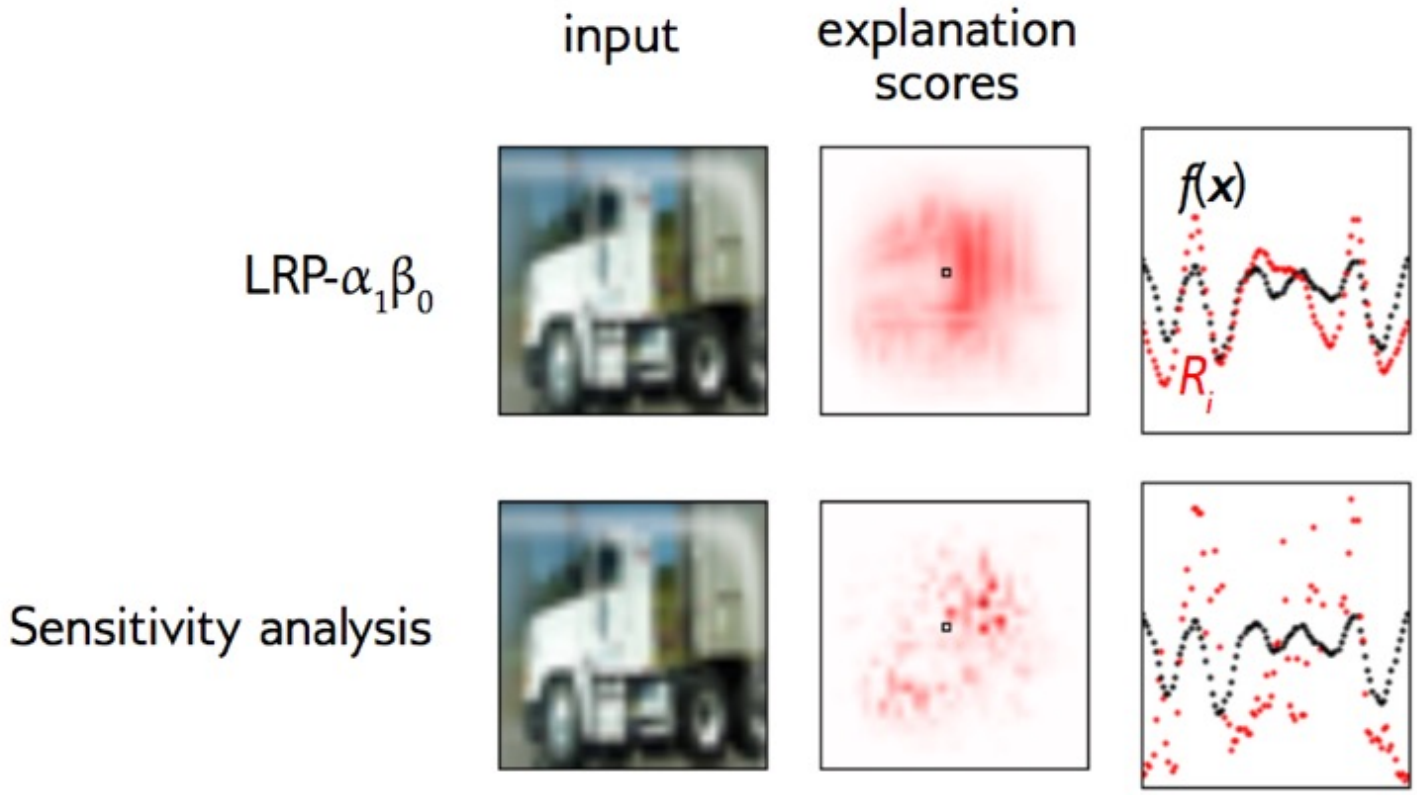


Method 2



Axiomatic Approach to Interpretability

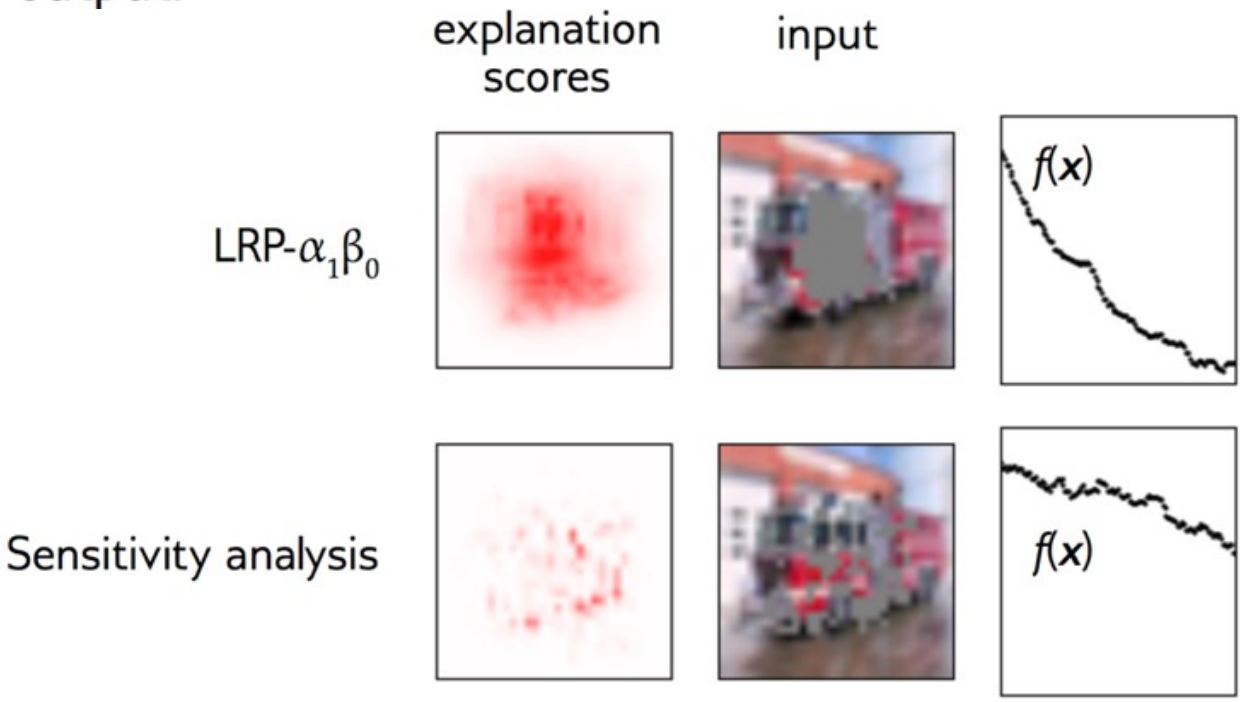
Testing Continuity



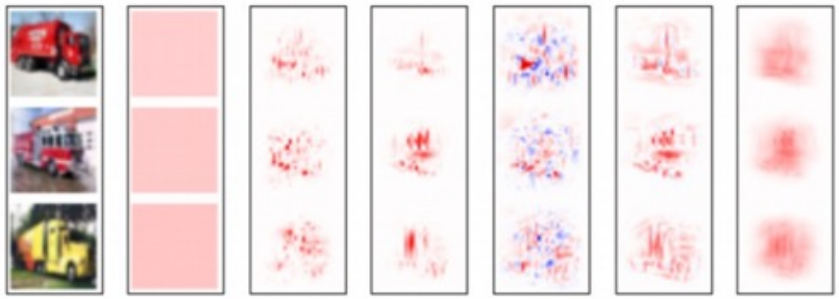
Axiomatic Approach to Interpretability

Property 4: Selectivity [Bach'15, Samek'17]

Model must agree with the explanation: If input features are attributed relevance, removing them should reduce evidence at the output.

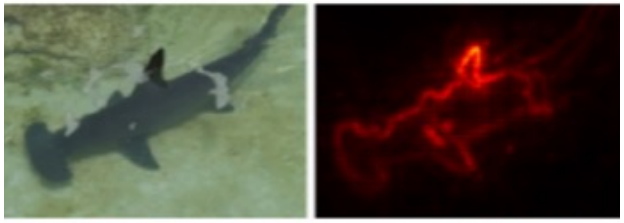


Axiomatic Approach to Interpretability

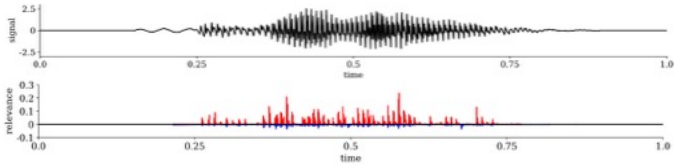
Explanation techniques		Uniform	(Gradient)^2	(Guided BP)^2	Gradient x Input	Guided BP x Input	LRP-$\alpha_1\beta_0$	...
Properties								
1. Conservation		✔			✔	✔	✔	
2. Positivity		✔	✔	✔		✔	✔	
3. Continuity		✔		✔		✔	✔	
4. Selectivity			✔	✔	✔	✔	✔	
...								

Summary LRP

General Images (Bach' 15, Lapuschkin'16)



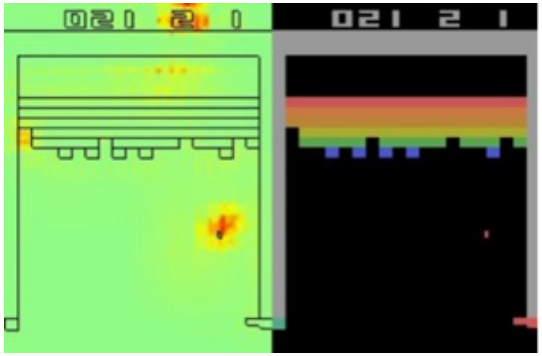
Speech (Becker'18)



Text Analysis (Arras'16 &17)

Morphing (Seibold'18)

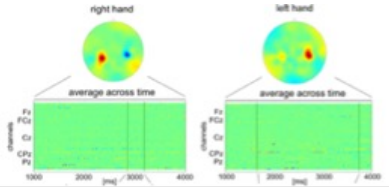
Games (Lapuschkin'18)



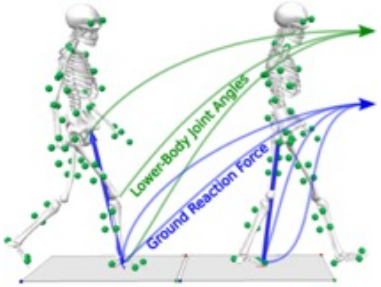
VQA (Arras'18)

Video (Anders'18)

EEG (Sturm'16)

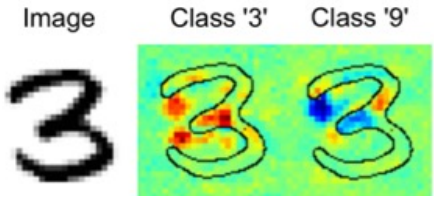


Gait Patterns (Horst'18)

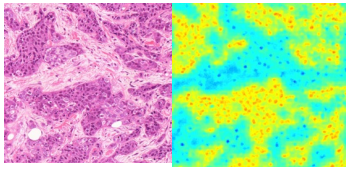


Faces (Lapuschkin'17)

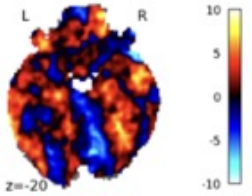
Digits (Bach' 15)



Histopathology (Binder'18)

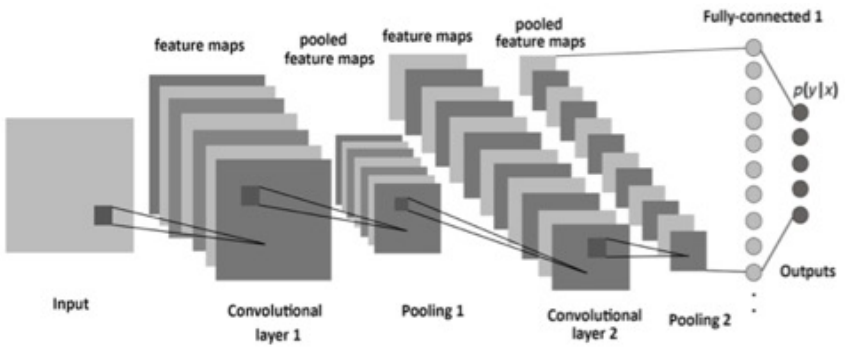


fMRI (Thomas'18)

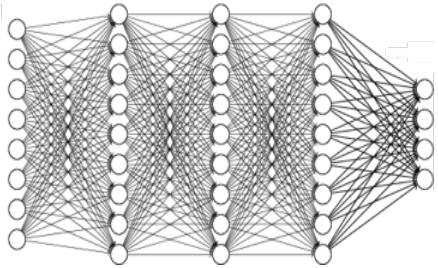


Summary LRP

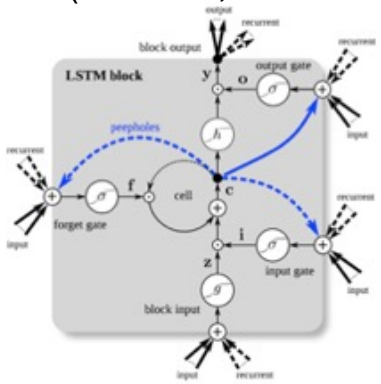
Convolutional NNs (Bach'15, Arras'17 …)



Local Renormalization Layers (Binder'16)

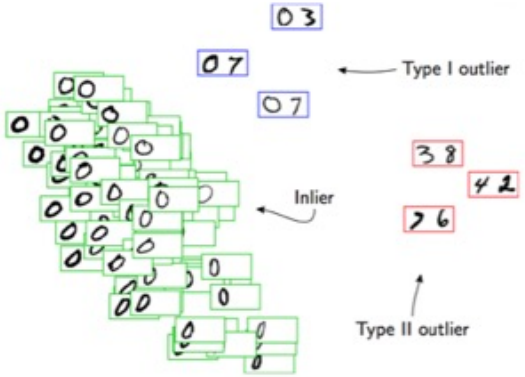


LSTM (Arras'17, Thomas'18)



Bag-of-words / Fisher Vector models (Bach'15, Arras'16, Lapuschkin'17, Binder'18)

One-class SVM (Kauffmann'18)



Summary LRP

1. LRP solves the “correct” explanation problem
2. It has a theoretical interpretation (Deep Taylor Decomposition)
3. It can be applied to various data and models (not only deep nets)
4. It fulfills various criteria (axiomatic approach)
5. It is flexible (many explanation methods are special cases of LRP)
6. In general: LRP $\neq$ Gradient $\times$ Input

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`

From LRP to Deep Taylor Decomposition

Decomposing the Correct Quantity

slope decomposition

$$\sum_i R_i = \|\nabla_x f\|^2$$

$\rightarrow$

value decomposition

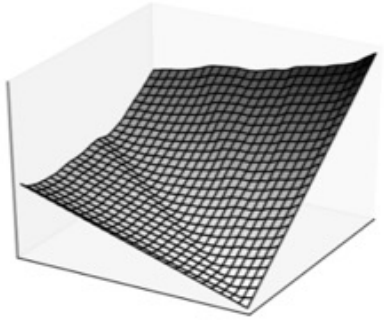
$$\sum_i R_i = f(\boldsymbol{x})$$

Candidate: Taylor decomposition

$$f(\boldsymbol{x}) = \underbrace{f(\tilde{\boldsymbol{x}})}_{0} + \sum_{i=1}^{d} \underbrace{\frac{\partial f}{\partial x_i}\Big|_{x=\tilde{x}} (x_i - \tilde{x}_i)}_{R_i} + \underbrace{O(\boldsymbol{x}\boldsymbol{x}^\top)}_{0}$$

- Achievable for linear models and deep ReLU networks without biases, by choosing:

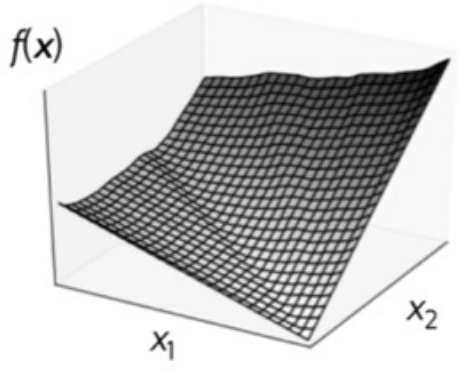
$$\tilde{\boldsymbol{x}} = \lim_{\varepsilon \to 0} \varepsilon \cdot \boldsymbol{x} \approx \mathbf{0}.$$



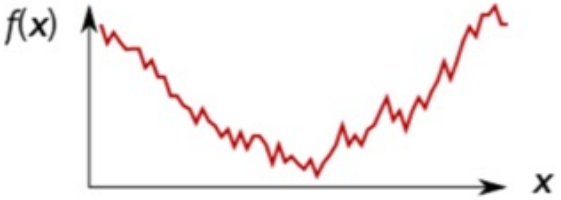
Why Simple Taylor doesn't work?

Two Reasons:

1. Root point is hard to find or too far → includes too much information (incl. negative evidence)

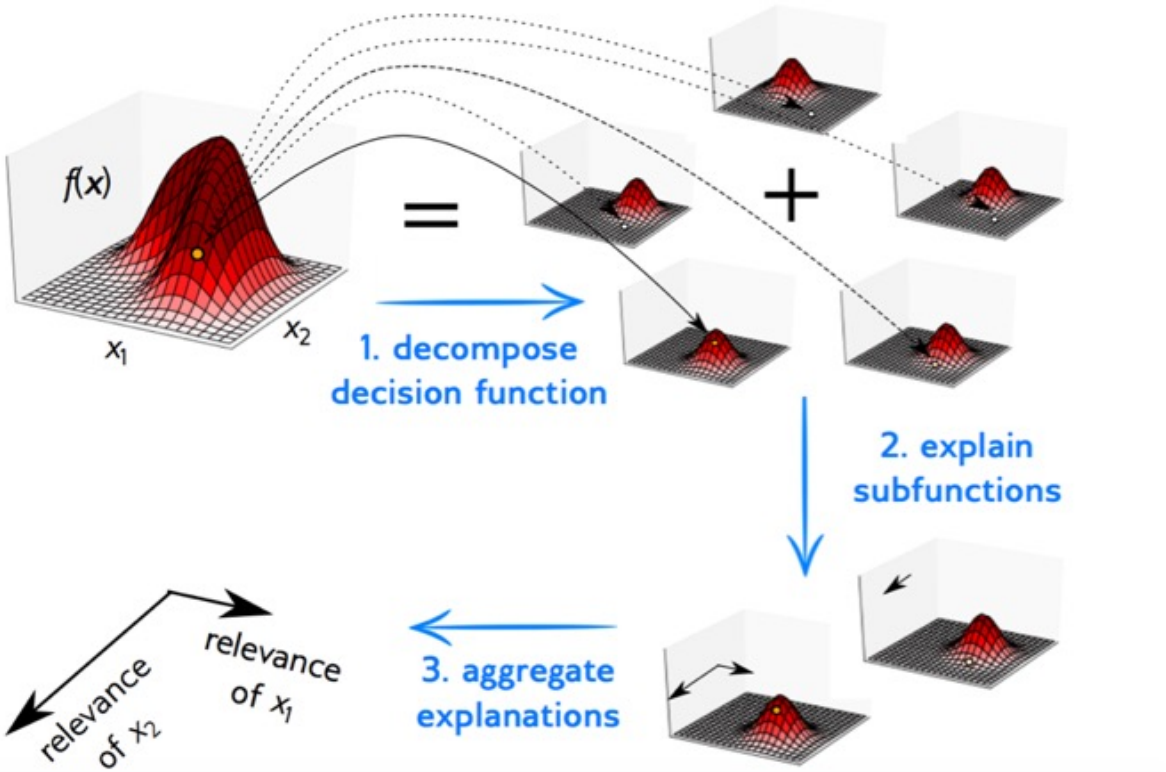


2. Gradient shattering problem → gradient of deep nets has low informative value



Deep Taylor Decomposition

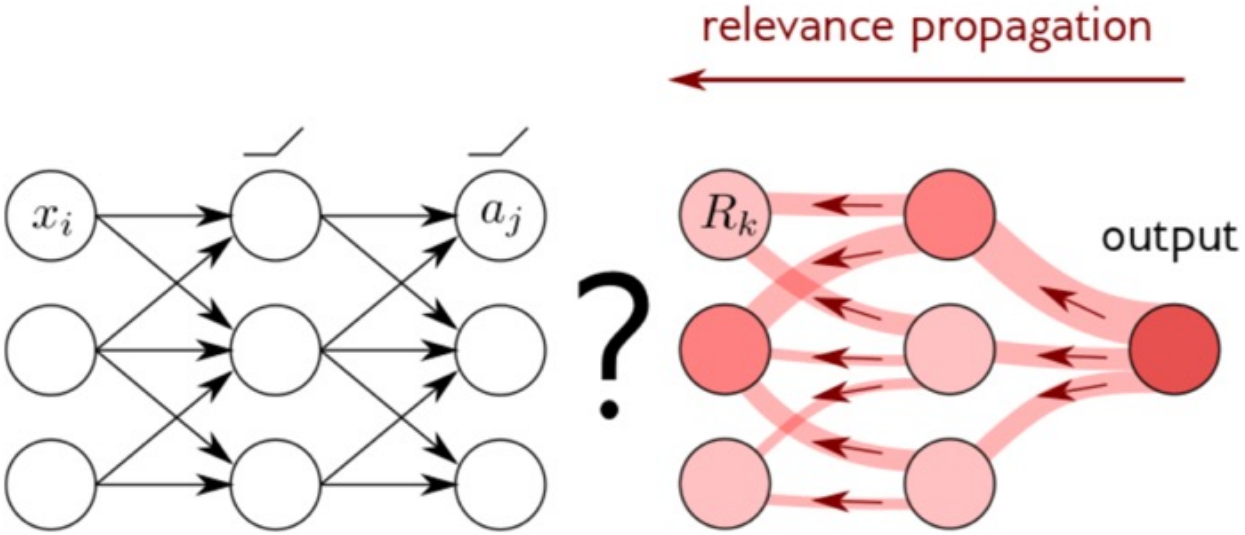
Idea: Since neural network is composed of simple functions, we propose a *deep* Taylor decomposition.



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

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

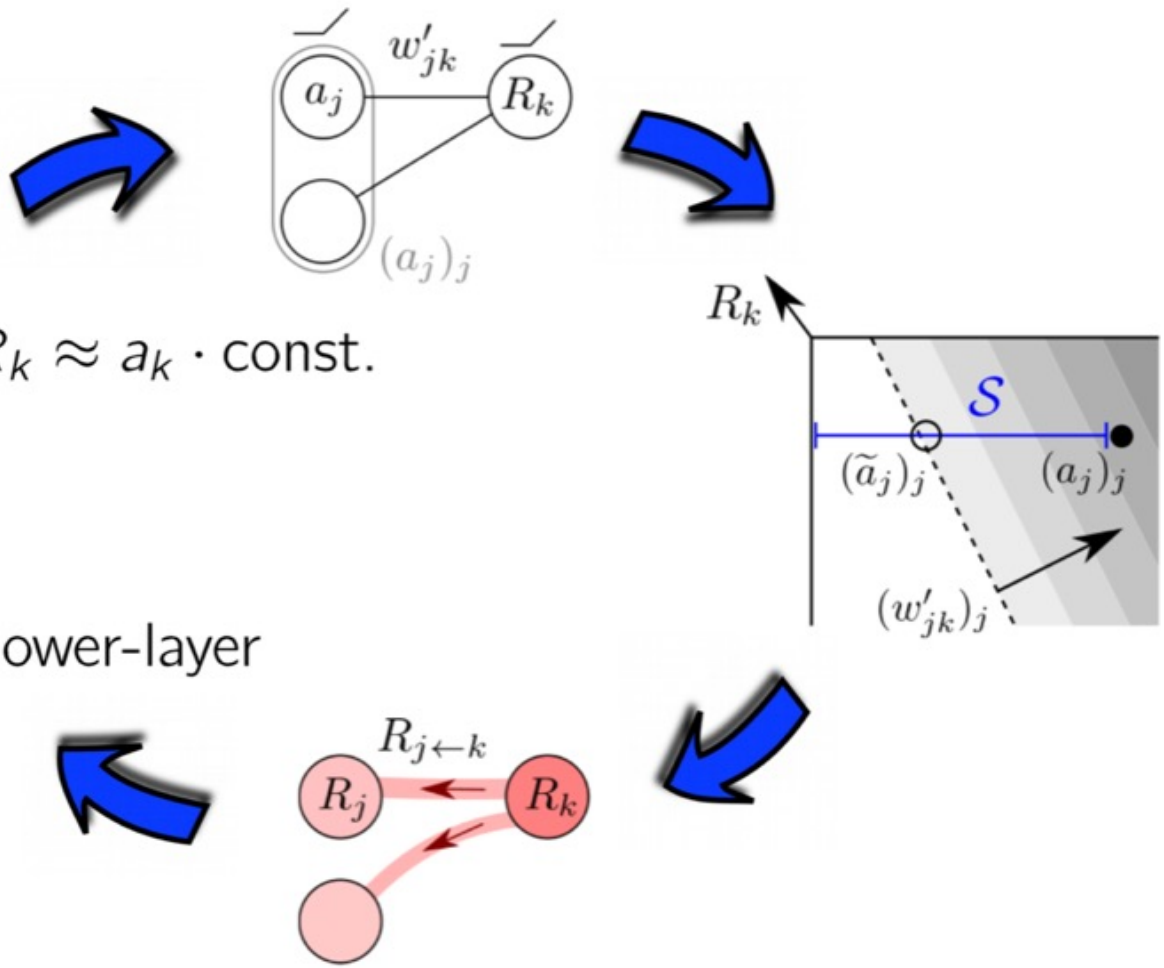
Deep Taylor Decomposition



Can we express R_k as a simple function of $(a_j)_j$?

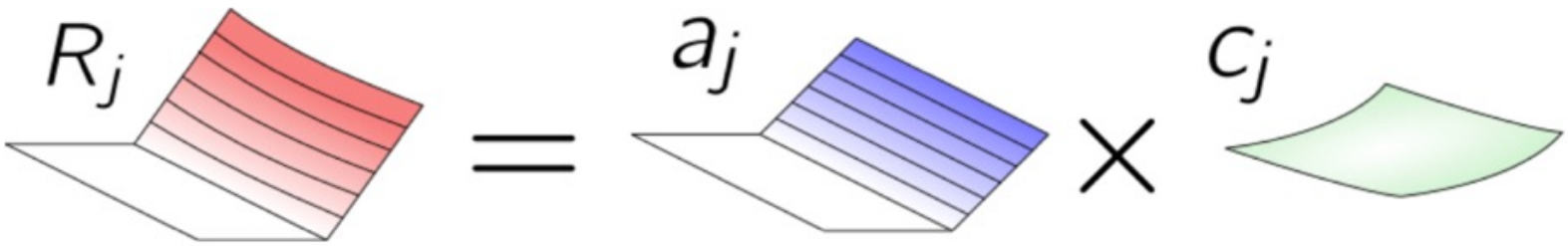
Can we do a Taylor decomposition of $R_k((a_j)_j)$?

Deep Taylor Decomposition



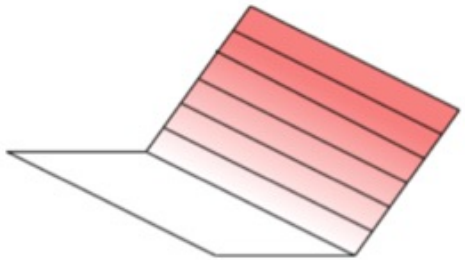
Deep Taylor Decomposition

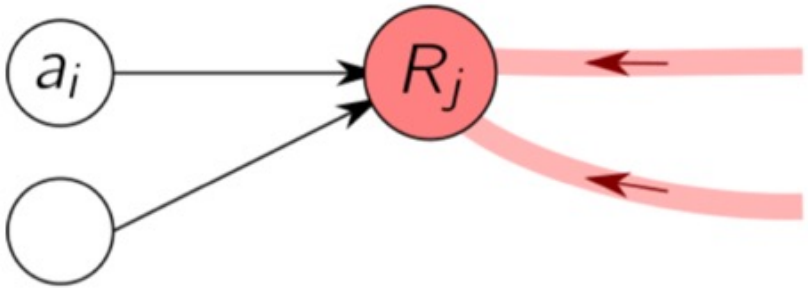
Proposition: Relevance at each layer is a product of the activation and an approximately constant term.



Deep Taylor Decomposition

1 Build the Relevance Neuron



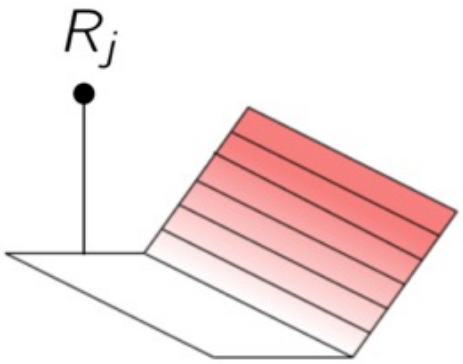


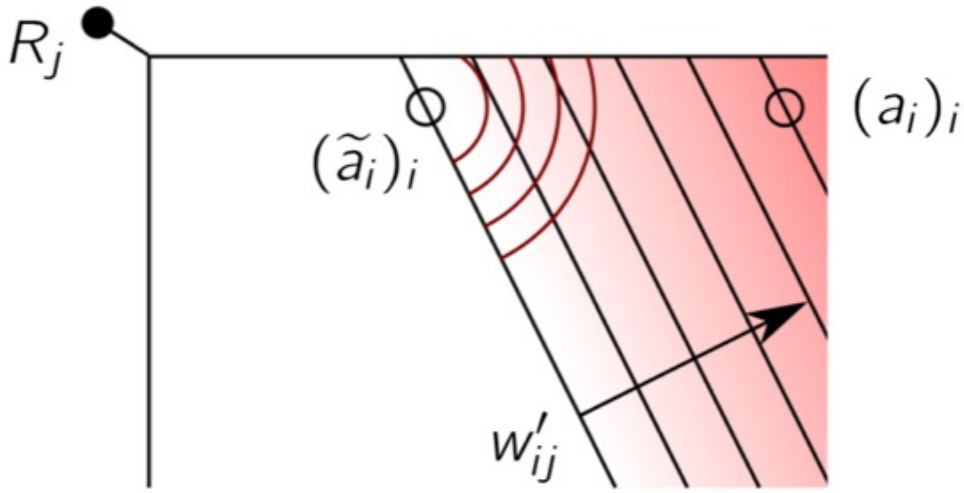
$$R_j = a_j c_j$$
$$= \max(0, \sum_i a_i w_{ij}) \cdot c_j$$
$$= \max(0, \sum_i a_i w'_{ij}) \qquad w'_{ij} = w_{ij} c_j$$

Deep Taylor Decomposition

Expand the Relevance Neuron

$$R_j((a_i)_i) = R_j((\tilde{a}_i)_i) + \sum_i \underbrace{\frac{\partial R_j}{\partial a_i}\Big|_{(\tilde{a}_i)_i} \cdot (a_i - \tilde{a}_i)}_{R_{i \leftarrow j}} + \varepsilon$$





Deep Taylor Decomposition

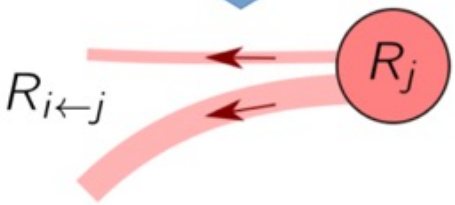
Decompose Relevance

Taylor expansion at root point:

$$R_j(\boldsymbol{a}) = R_j(\tilde{\boldsymbol{a}}^{(j)}) + \sum_i \frac{\partial R_j}{\partial a_i}\Big|_{\tilde{\boldsymbol{a}}^{(j)}} \cdot (a_i - \tilde{a}_i^{(j)}) + \varepsilon$$

$$0 \qquad \frac{(a_i - \tilde{a}_i^{(j)})w_{ij}}{\sum_i (a_i - \tilde{a}_i^{(j)})w_{ij}} R_j \qquad 0$$

Relevance can now be backward propagated



Deep Taylor Decomposition

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

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

$$R_{i \leftarrow j} = \frac{a_i w_{ij}^+}{\sum_i a_i w_{ij}^+} R_j \qquad \text{(LRP-}\alpha_1\beta_0\text{)}$$

Deep Taylor Decomposition

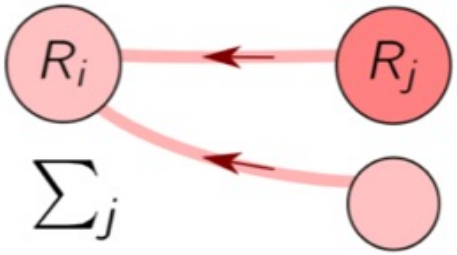
Input domain	Rule
ReLU activations $(a_j \geq 0)$	$R_j = \sum_k \frac{a_j w_{jk}^+}{\sum_j a_j w_{jk}^+} R_k$
Pixel intensities $(x_i \in [l_i, h_i],$ $l_i \leq 0 \leq h_i)$	$R_i = \sum_j \frac{x_i w_{ij} - l_i w_{ij}^+ - h_i w_{ij}^-}{\sum_i x_i w_{ij} - l_i w_{ij}^+ - h_i w_{ij}^-} R_j$
Real values $(x_i \in \mathbb{R})$	$R_i = \sum_j \frac{w_{ij}^2}{\sum_i w_{ij}^2} R_j$

Deep Taylor LRP rules [Montavon'17]

More refined rules can also be constructed to match the input data distribution [Kindermans'17]

Deep Taylor Decomposition

Pooling relevance over all outgoing neurons



Deep Taylor Decomposition

The LRP-$\alpha_1\beta_0$ rule

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

can be seen as

a deep Taylor decomposition (DTD) [Montavon'17]

which then yields

domain- and layer-specific rules


Coffee Break

Tutorial on Interpretable Machine Learning

Part 3: Applications of Interpretability

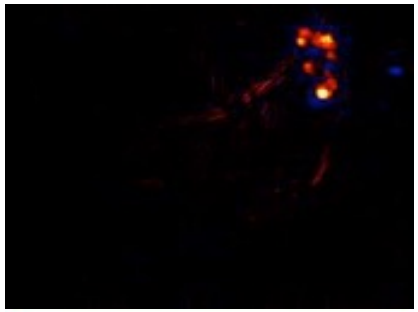
LRP revisited

input $\boldsymbol{x}$

Black Box

Rooster

prediction $f(\boldsymbol{x})$



heatmap

Explain prediction

(how much each pixel contributes to prediction)

Idea: Decompose function

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

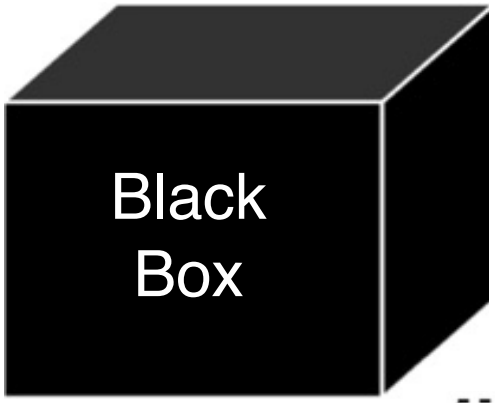
Simple LRP rule (Bach et al. 2015)

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

Every neuron gets its "share" of the redistributed relevance

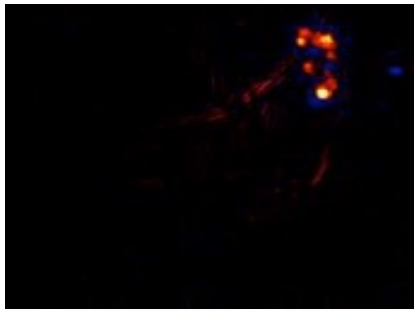
LRP revisited

input $\boldsymbol{x}$



Rooster

prediction $f(\boldsymbol{x})$



heatmap

Explain prediction

(how much each pixel contributes to prediction)

Idea: Decompose function

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

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

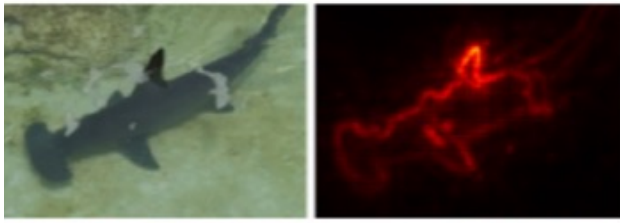
$$R_i^{(l)} = \sum_j (\alpha \cdot \frac{(x_i \cdot w_{ij})^+}{\sum_{i'} (x_{i'} \cdot w_{i'j})^+} + \beta \cdot \frac{(x_i \cdot w_{ij})^-}{\sum_{i'} (x_{i'} \cdot w_{i'j})^-}) R_j^{(l+1)}$$

where $\alpha + \beta = 1$

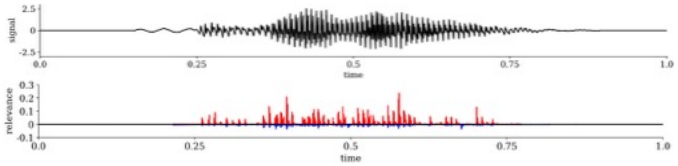
Theoretical Interpretation
(Deep) Taylor decomposition

LRP revisited

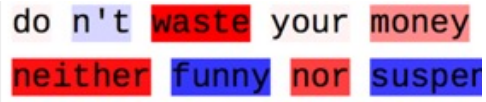
General Images (Bach' 15, Lapuschkin'16)



Speech (Becker'18)

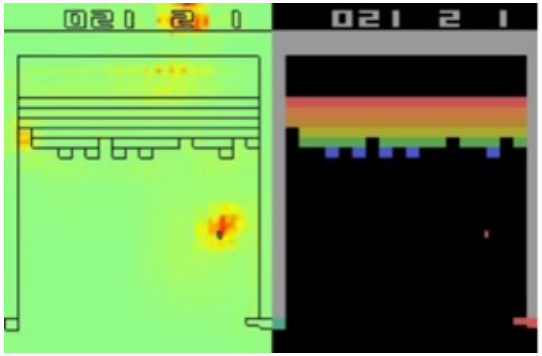


Text Analysis (Arras'16 &17)



Morphing (Seibold'18)

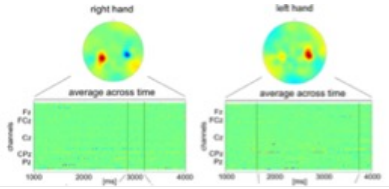
Games (Lapuschkin'18)



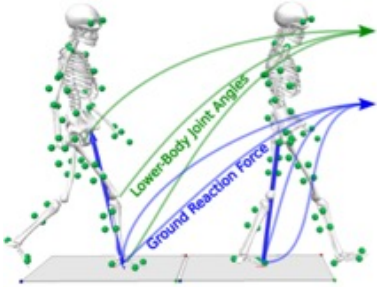
VQA (Arras'18)

Video (Anders'18)

EEG (Sturm'16)

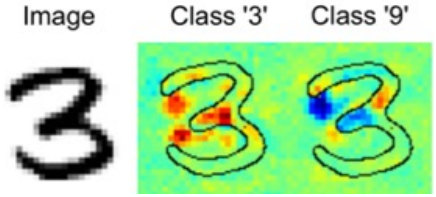


Gait Patterns (Horst'18)

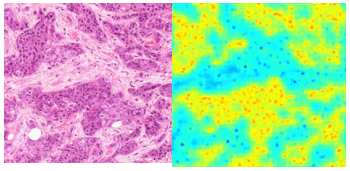


Faces (Lapuschkin'17)

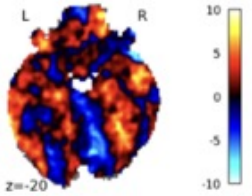
Digits (Bach' 15)



Histopathology (Binder'18)

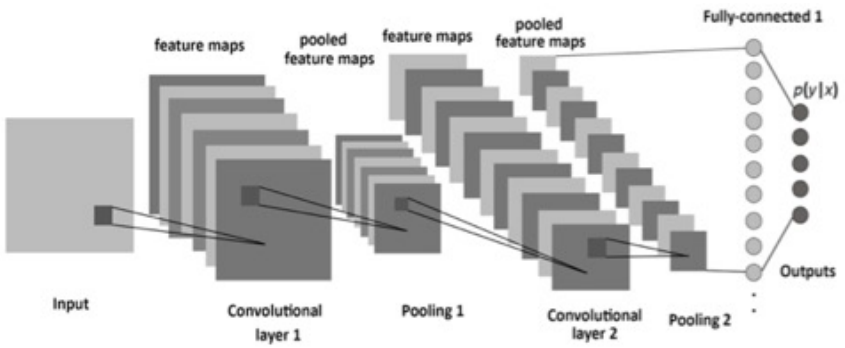


fMRI (Thomas'18)

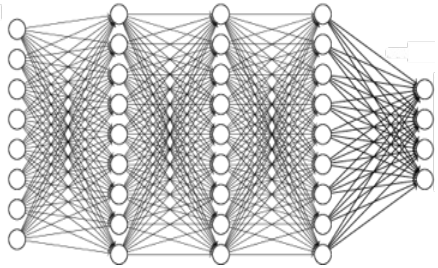


LRP revisited

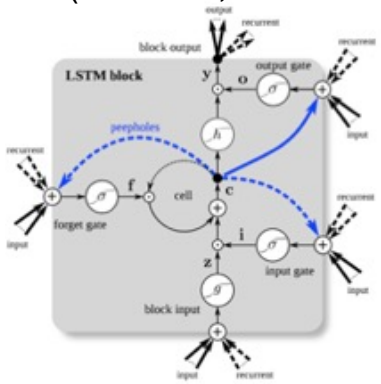
Convolutional NNs (Bach'15, Arras'17 …)



Local Renormalization Layers (Binder'16)

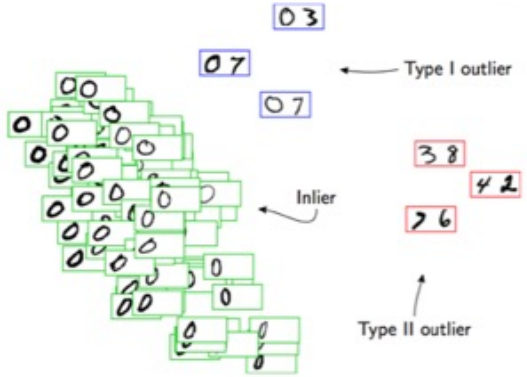


LSTM (Arras'17, Thomas'18)



Bag-of-words / Fisher Vector models (Bach'15, Arras'16, Lapuschkin'17, Binder'18)

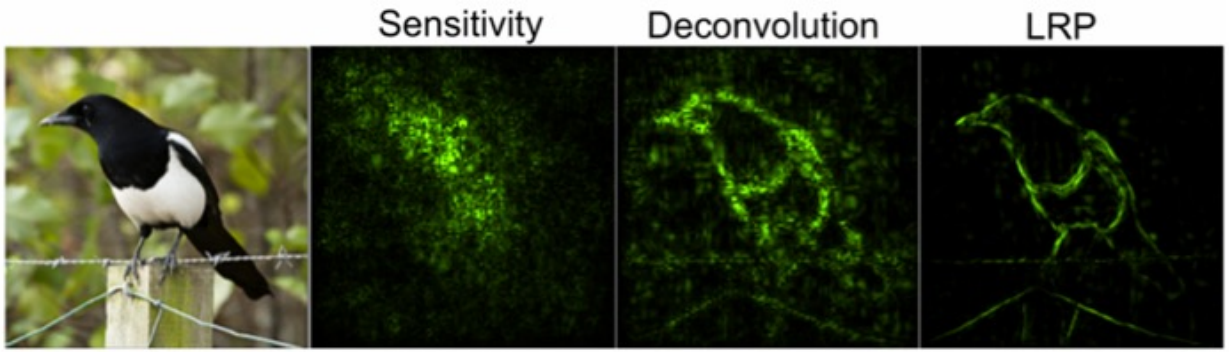
One-class SVM (Kauffmann'18)



LRP & Others

Evaluating Heatmap Quality

Compare Explanation Methods



Can we objectively measure which heatmap is best ?

Idea: Compare selectivity (Bach'15, Samek'17):

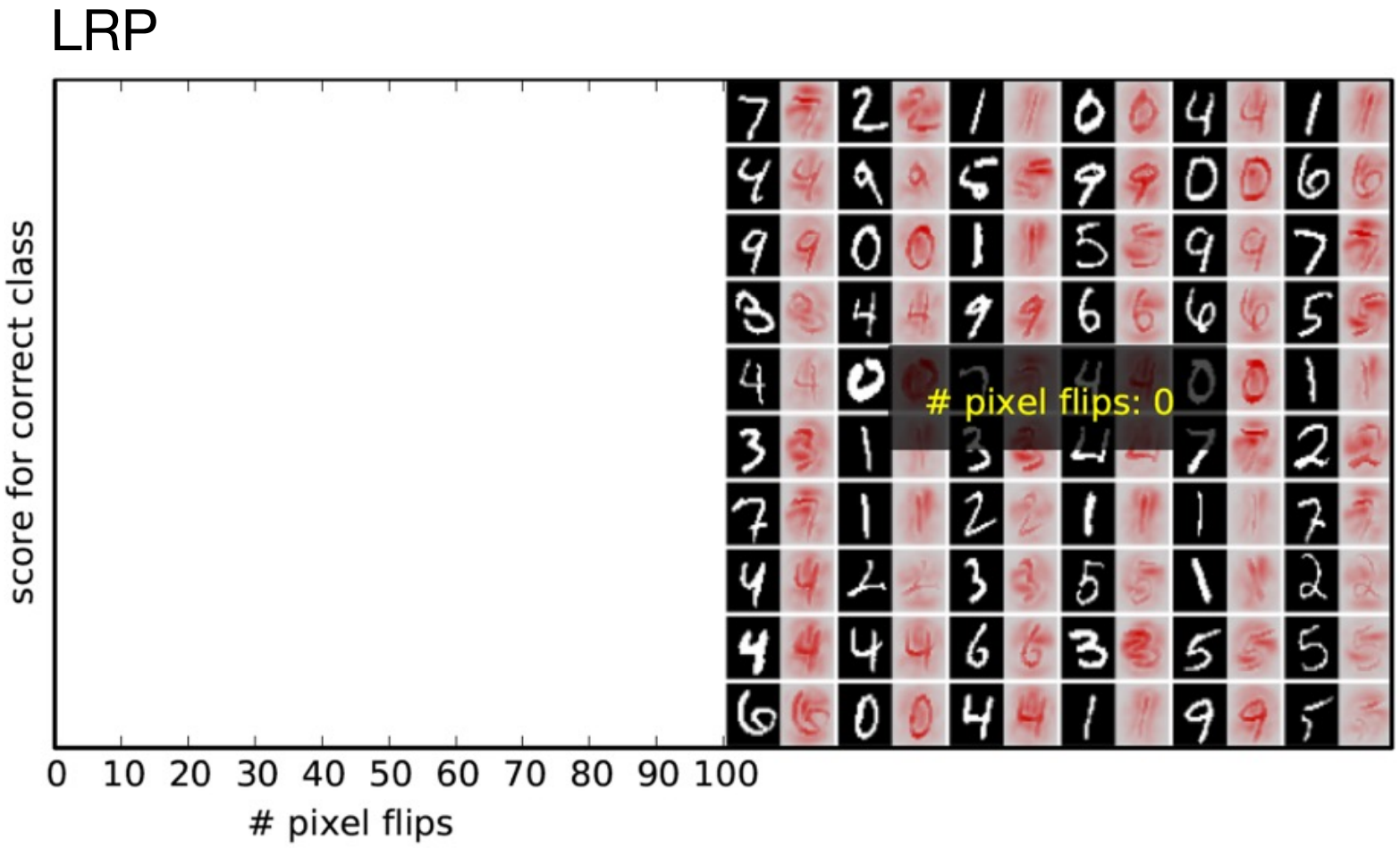
"If input features are deemed relevant, removing them should reduce evidence at the output of the network."

Algorithm ("Pixel Flipping")

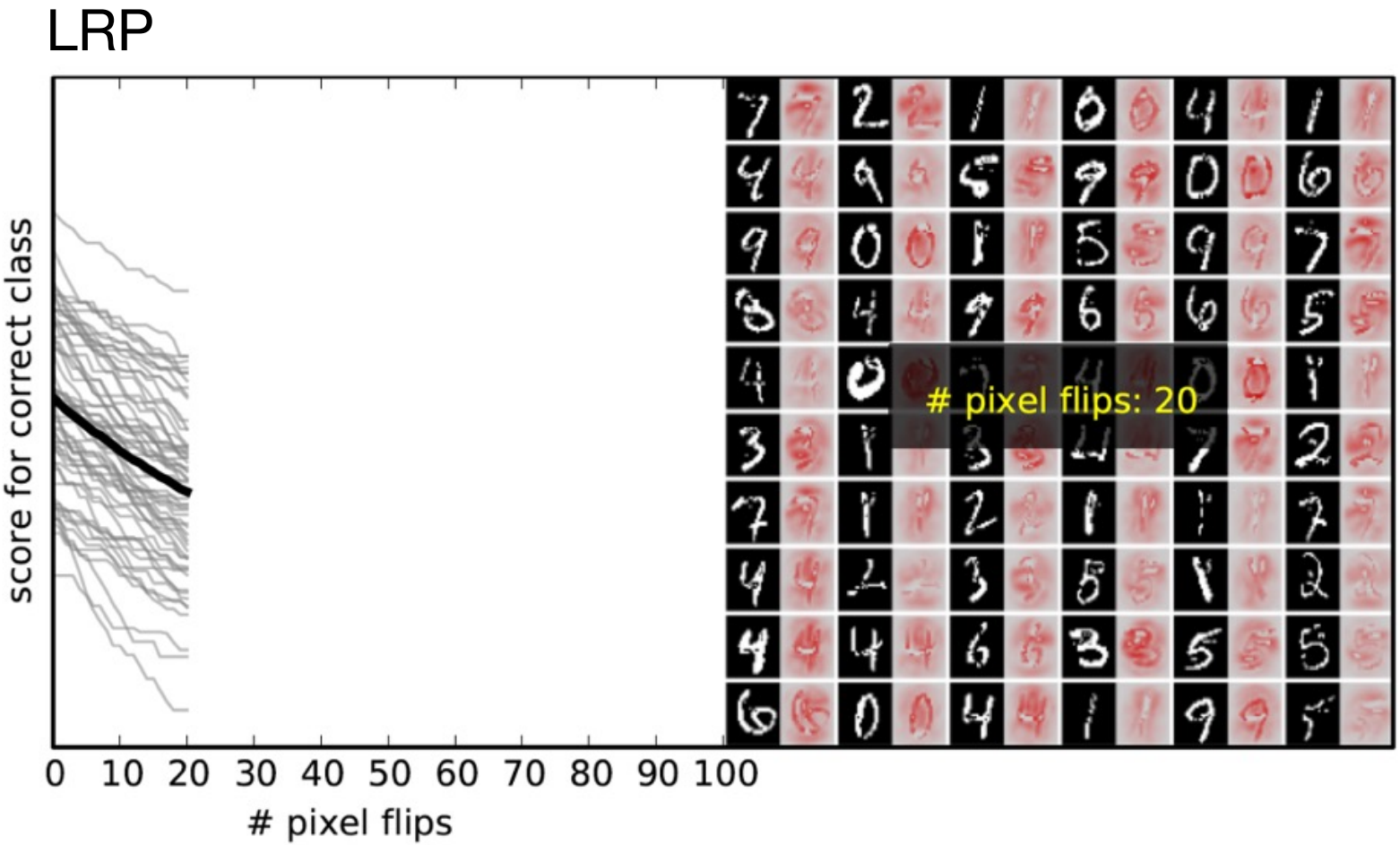
```
Sort pixels / patches by relevance
Iterate
  destroy pixel / patch
  evaluate f(x)
Measure decrease of f(x)
```

Important: Remove information in a non-specific manner (e.g. sample from uniform distribution)

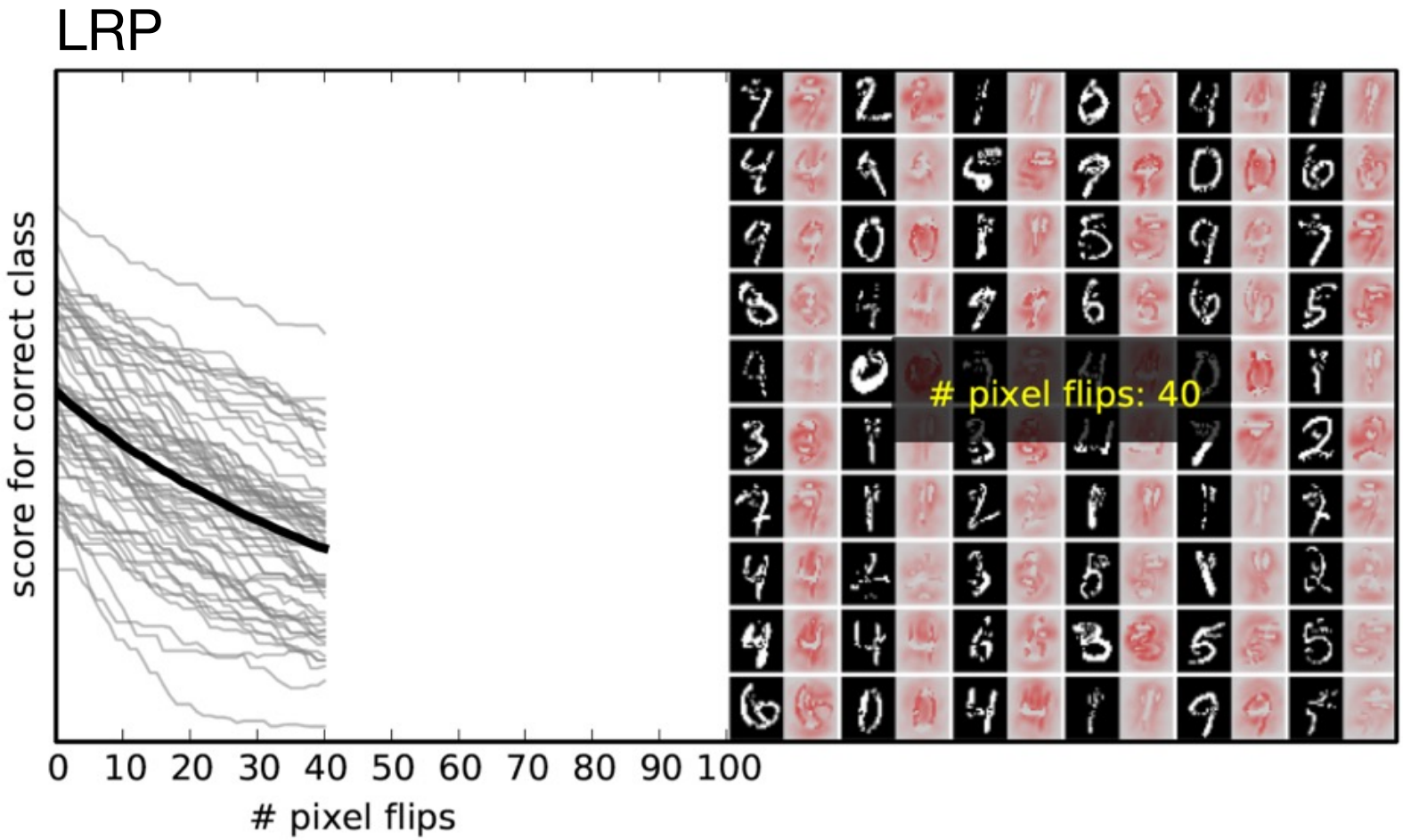
Compare Explanation Methods



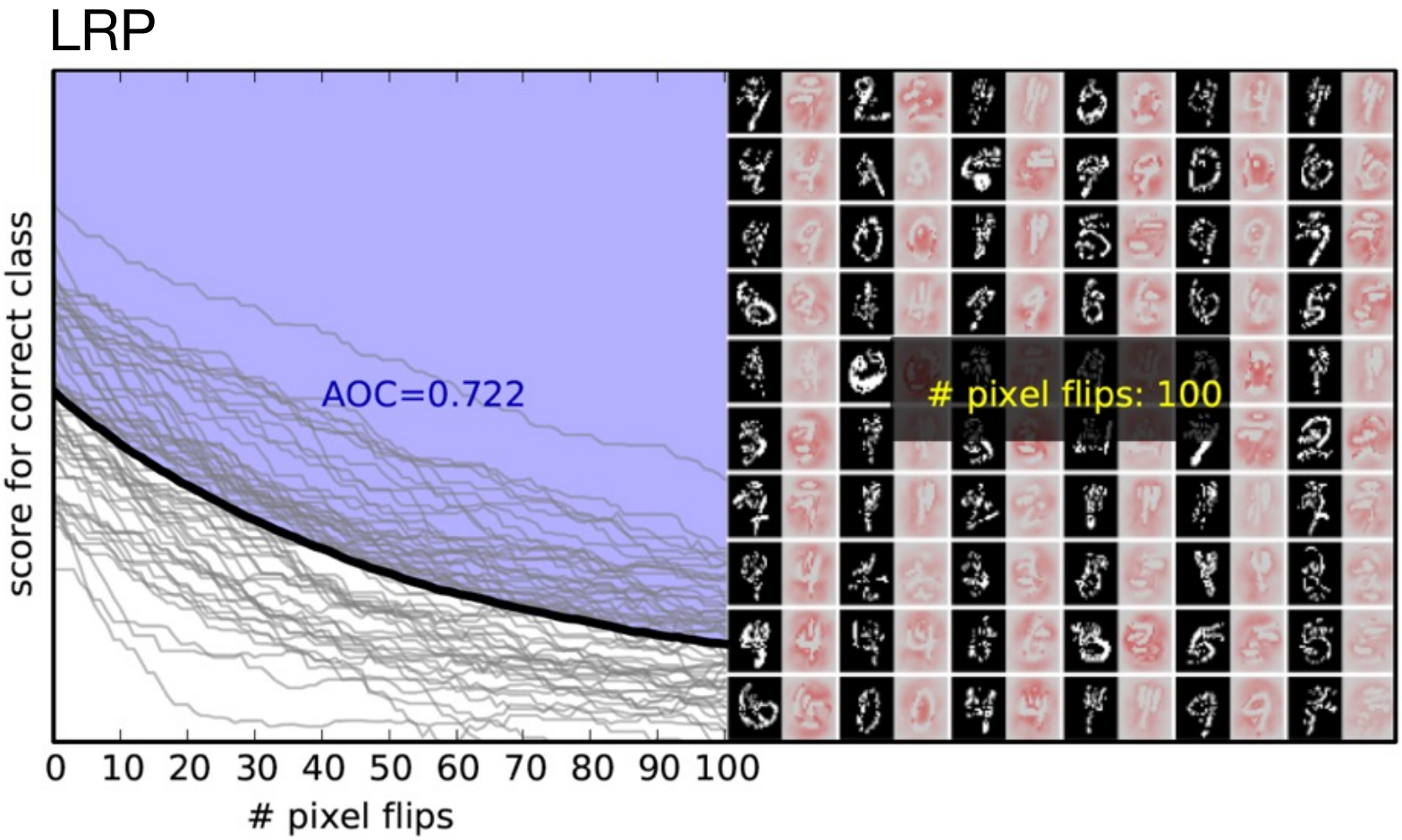
Compare Explanation Methods



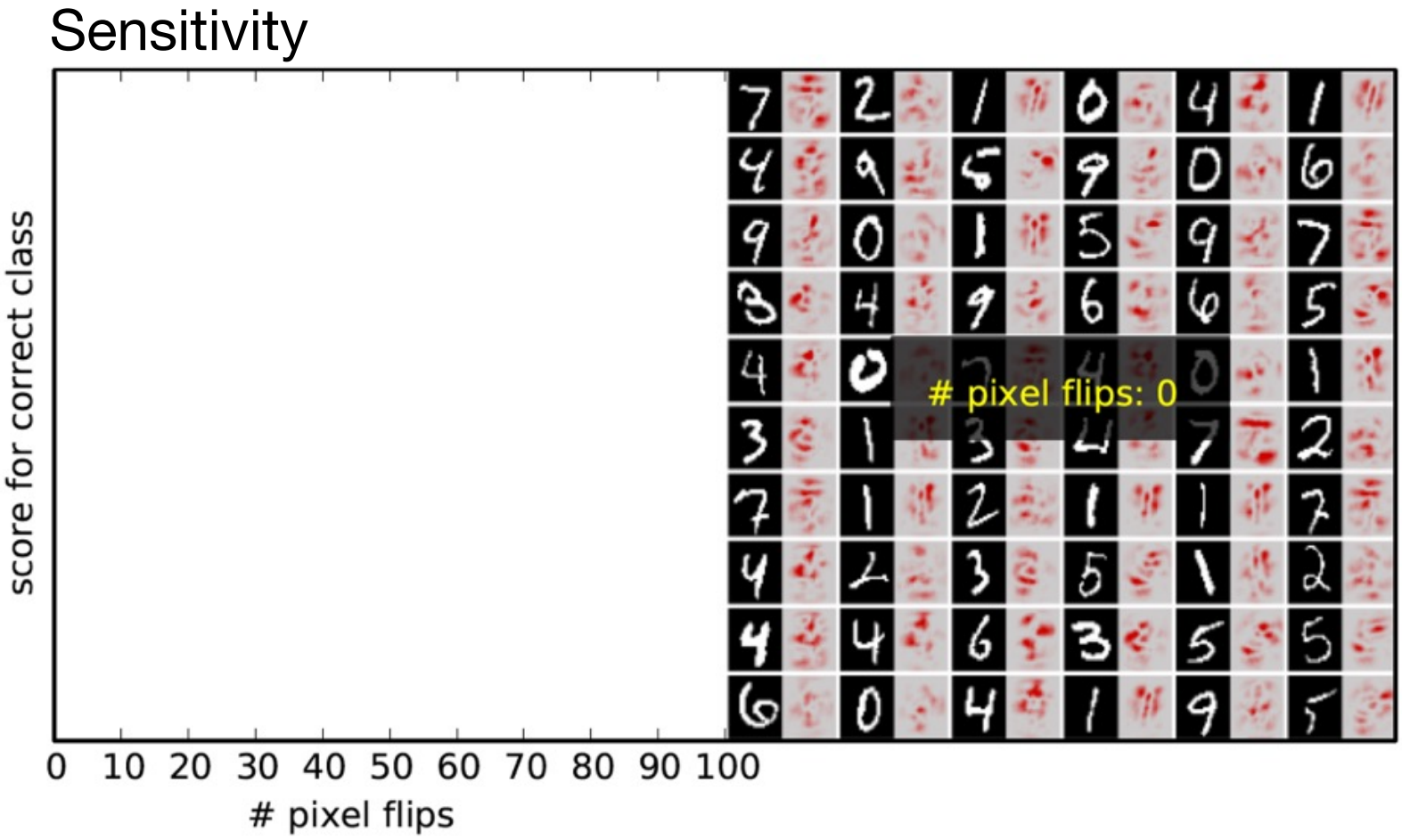
Compare Explanation Methods



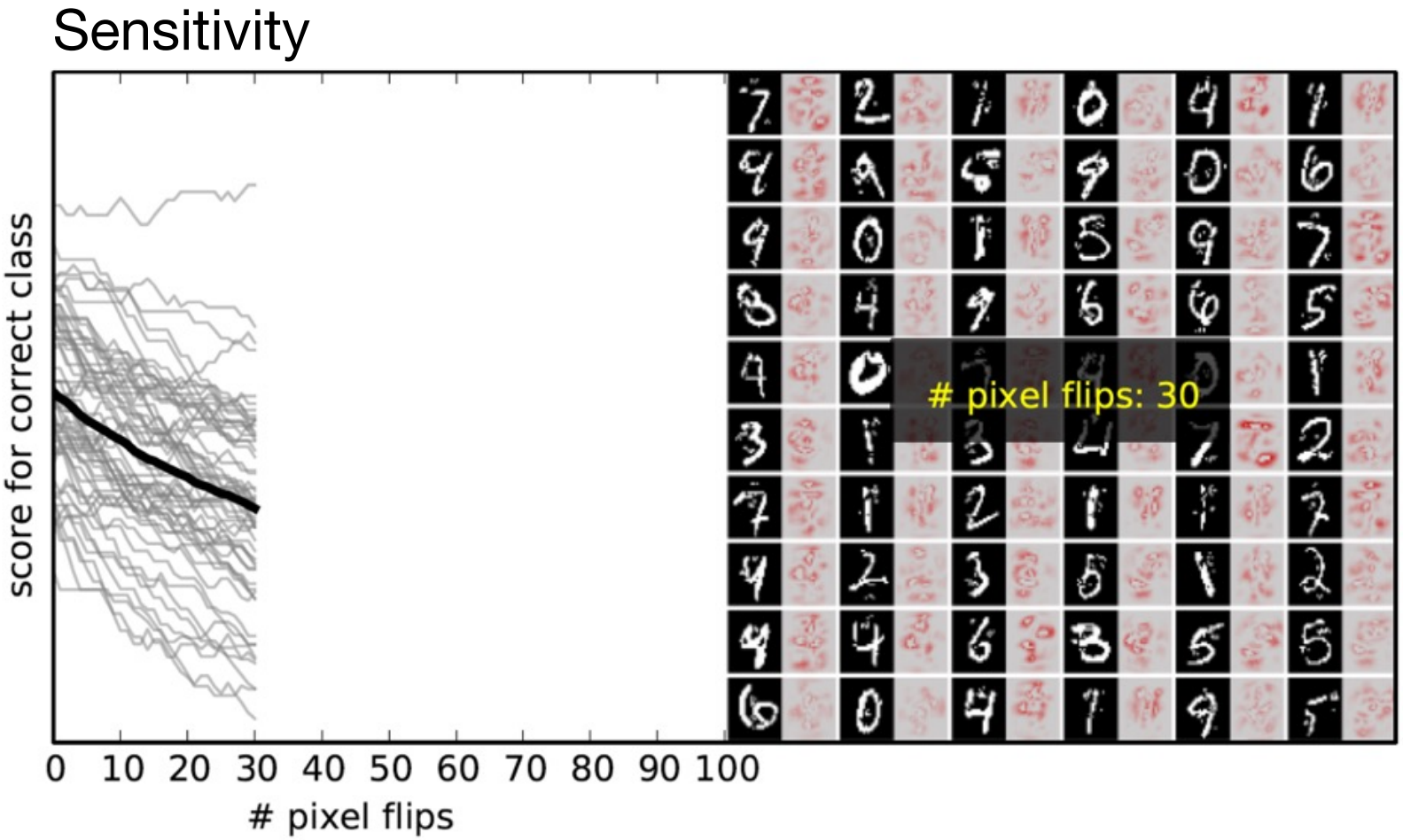
Compare Explanation Methods



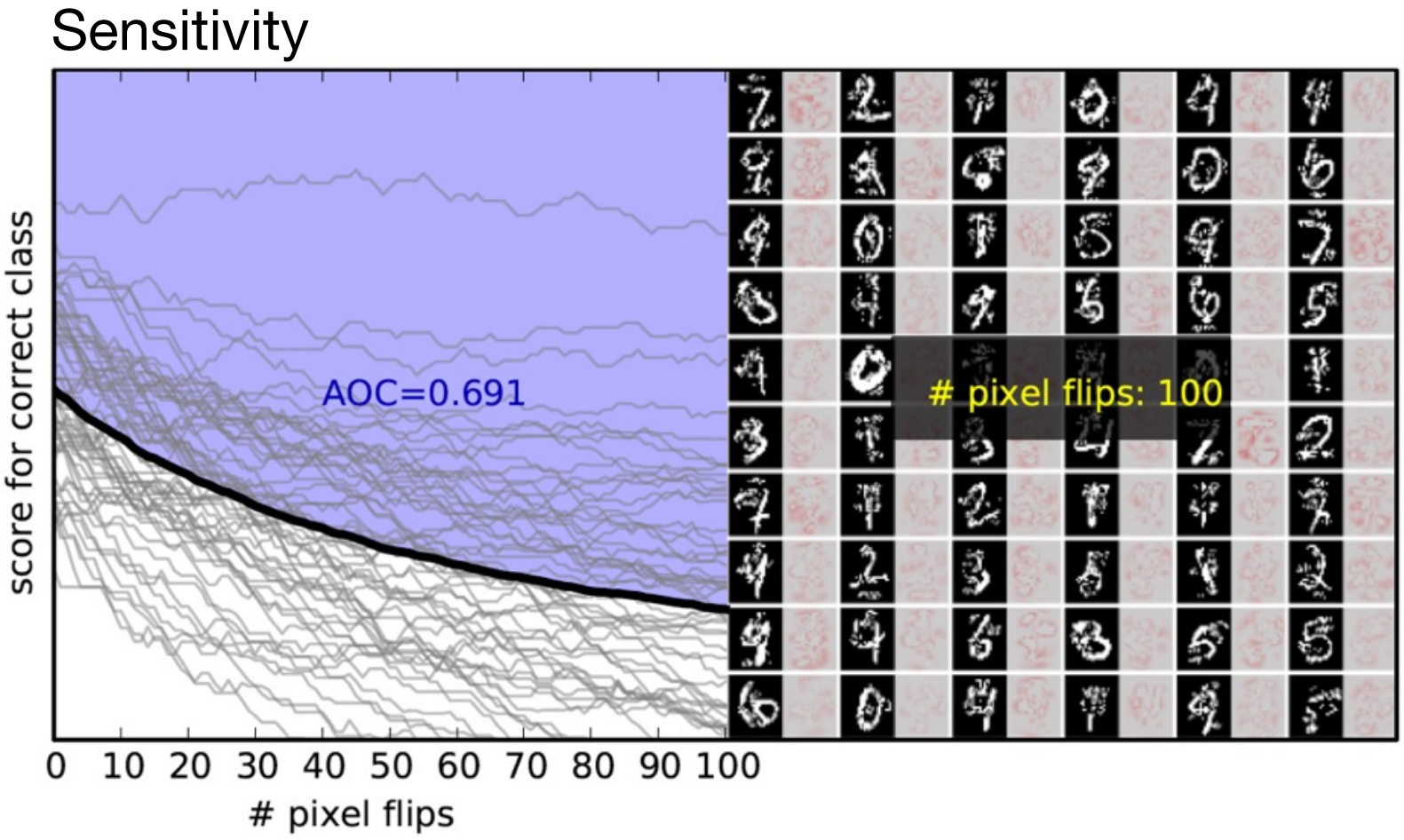
Compare Explanation Methods



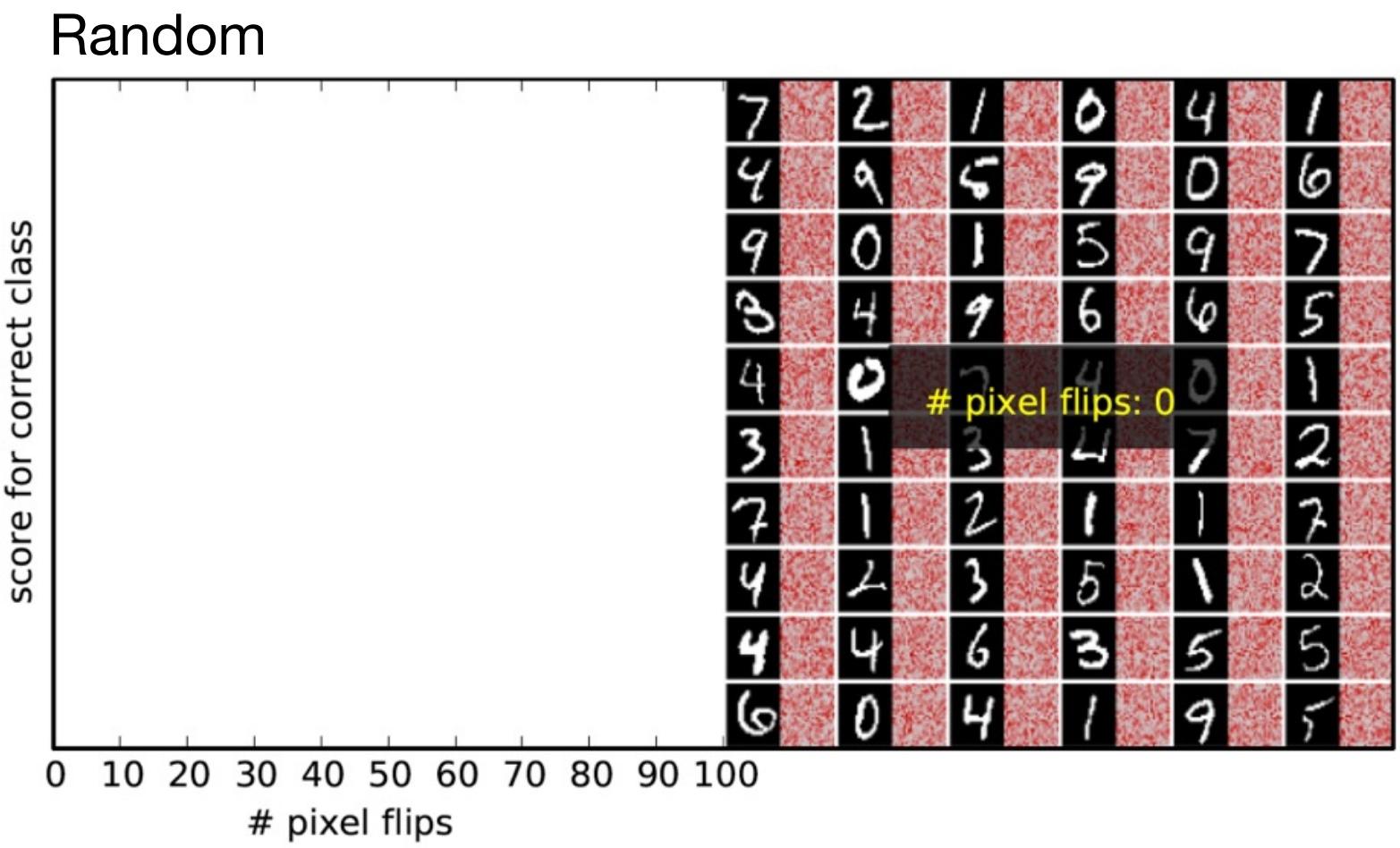
Compare Explanation Methods



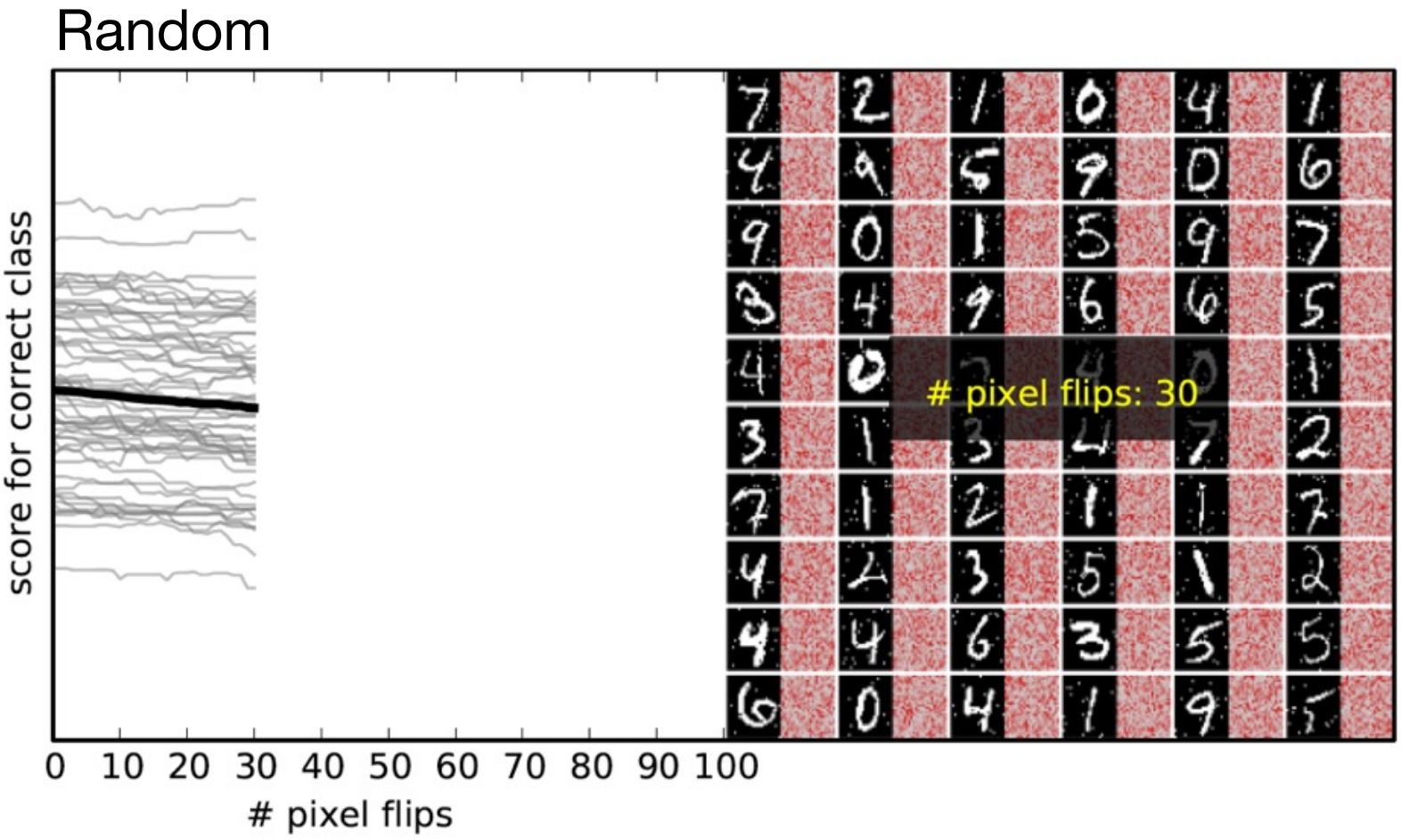
Compare Explanation Methods



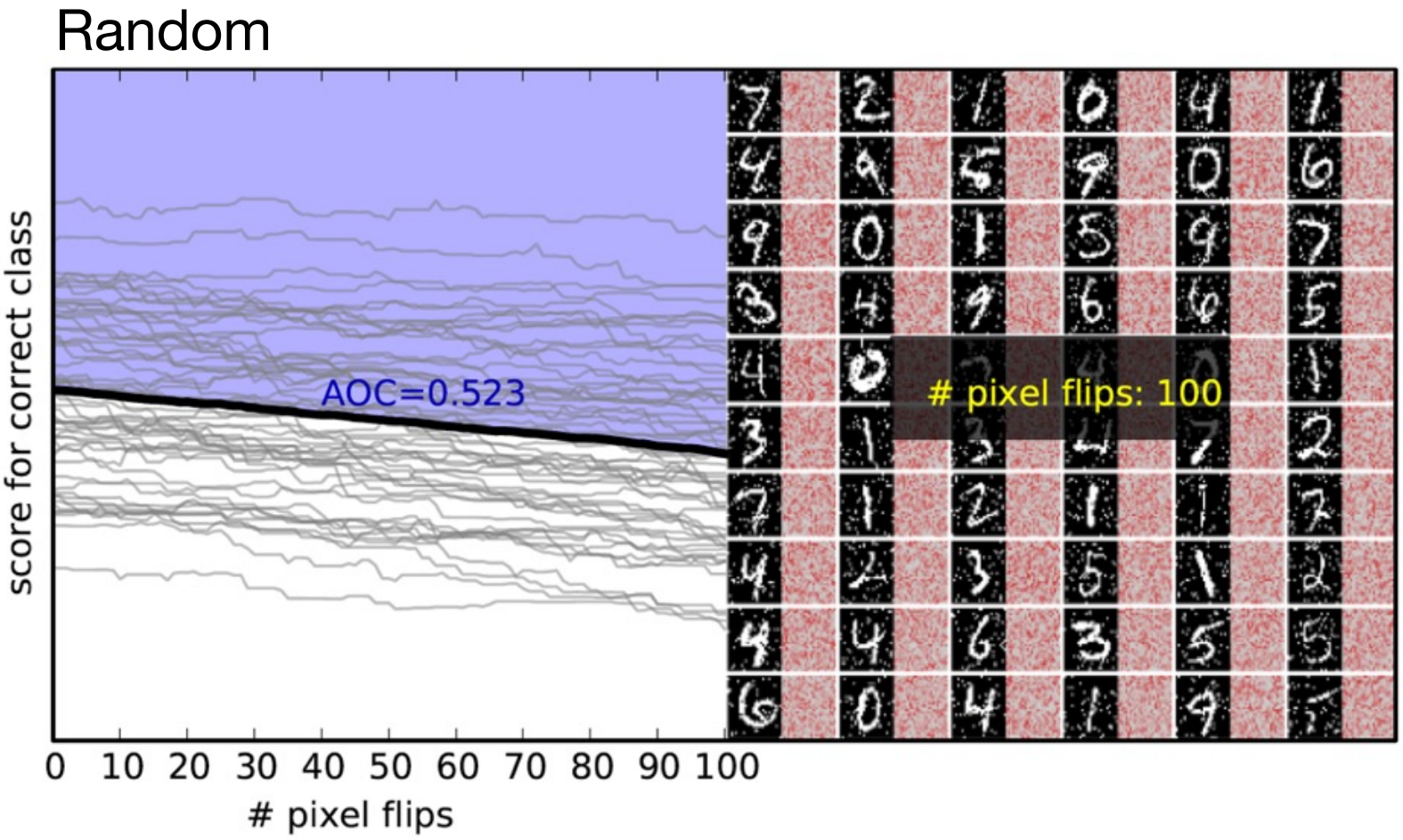
Compare Explanation Methods



Compare Explanation Methods



Compare Explanation Methods



Compare Explanation Methods

LRP:	**0.722**
Sensitivity:	0.691
Random:	0.523

LRP produces quantitatively better heatmaps than sensitivity analysis and random.

What about more complex datasets ?

SUN397

397 scene categories
(108,754 images in total)

ILSVRC2012

1000 categories
(1.2 million training images)

MIT Places

205 scene categories
(2.5 millions of images)

Compare Explanation Methods

Sensitivity Analysis (Simonyan et al. 2014)

Deconvolution Method (Zeiler & Fergus 2014)

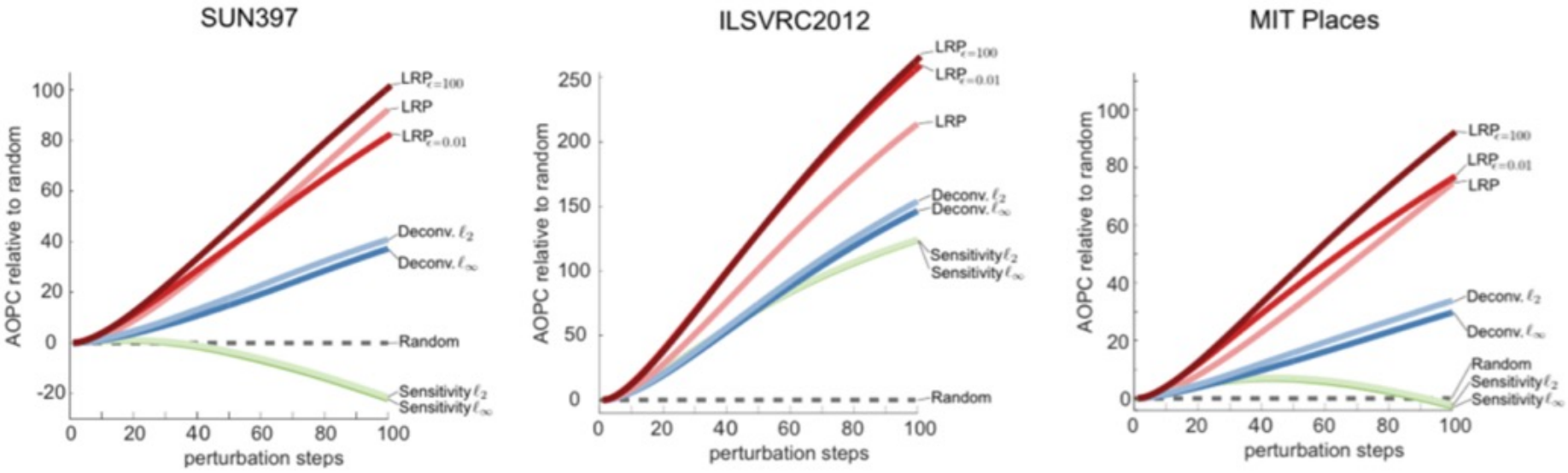
LRP Algorithm (Bach et al. 2015)

(Samek et al. 2017)

Compare Explanation Methods

Red: LRP method
Blue: Deconvolution method (Zeiler & Fergus, 2014)
Green: Sensitivity method (Simonyan et al., 2014)

- ImageNet: Caffe reference model
- Places & SUN: Classifier from MIT
- AOPC averages over 5040 images
- perturb 9 × 9 nonoverlapping regions
- 100 steps (15.7% of the image)
- uniform sampling in pixel space



(Samek et al. 2017)

LRP produces better heatmaps

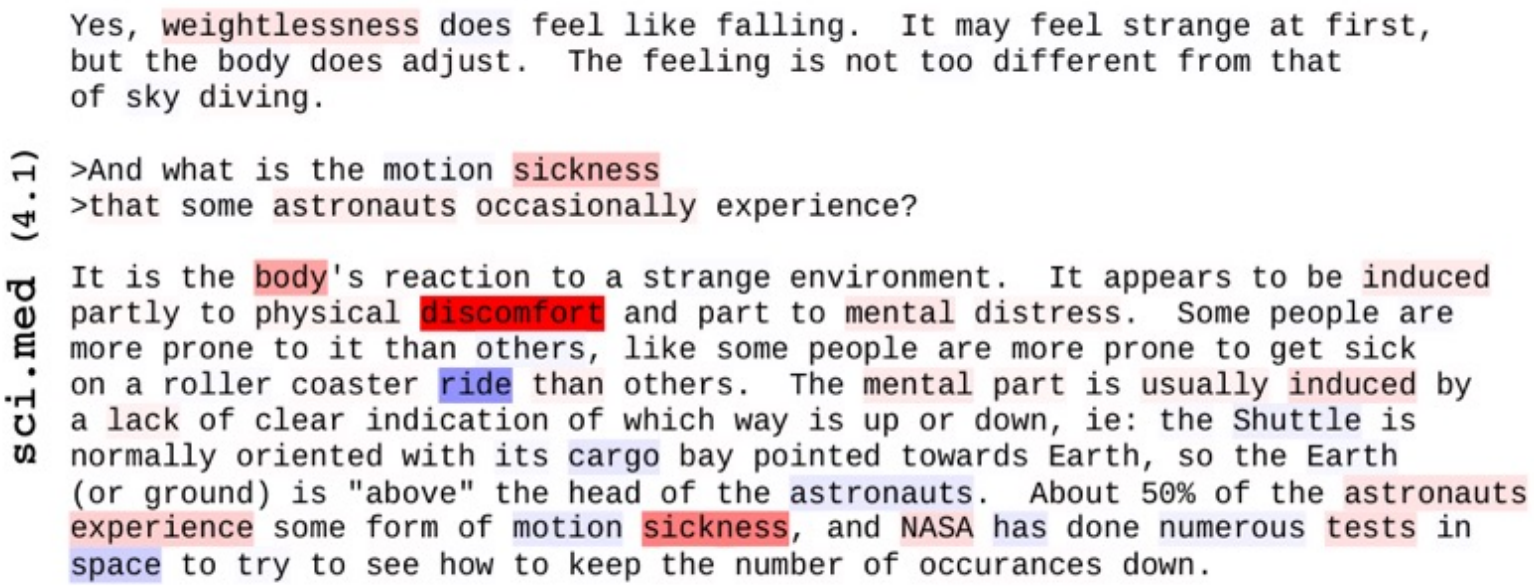
- Sensitivity heatmaps are noisy (gradient shuttering)
- Deconvolution and sensitivity analysis solve a different problem

Compare Explanation Methods

Same idea can be applied for other domains (e.g. text document classification)

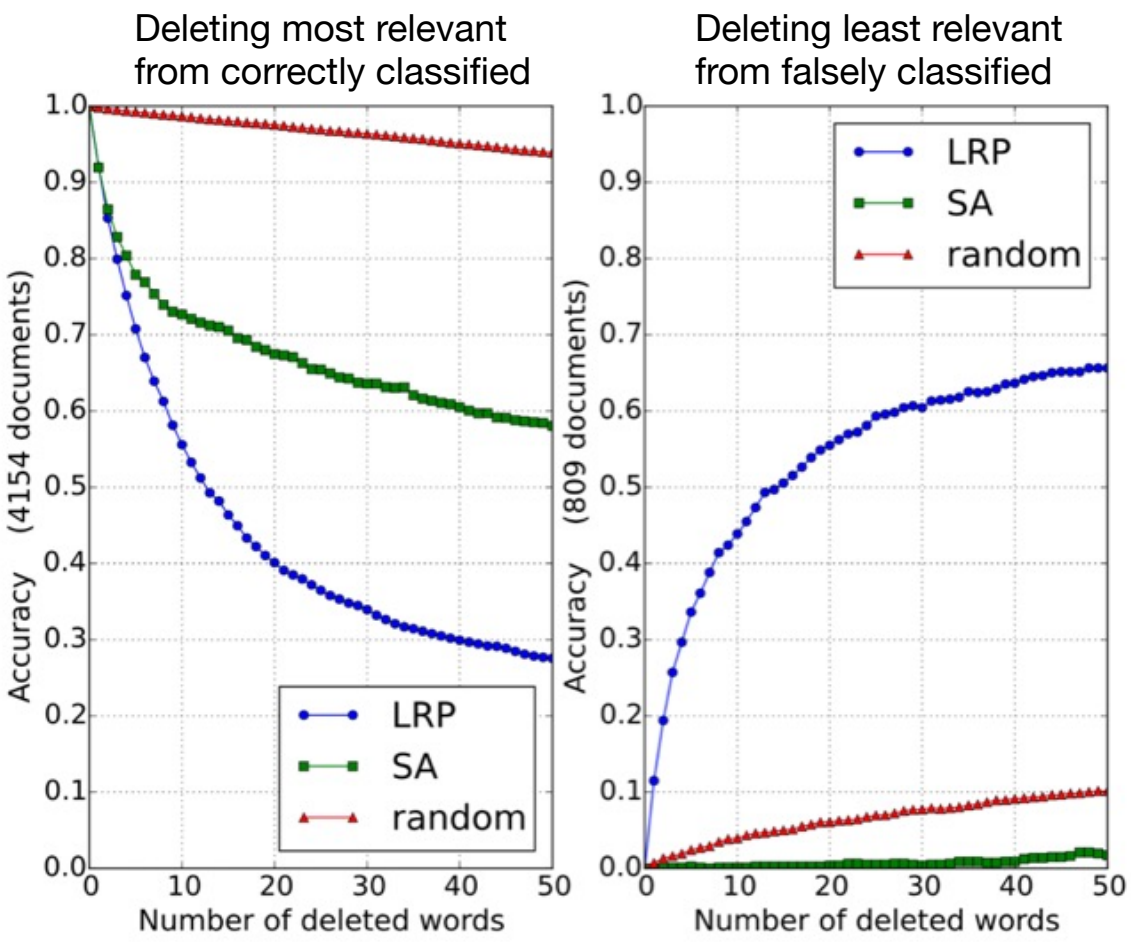
"Pixel flipping"
=
"Word deleting"

Text classified as "sci.med" —> LRP identifies most relevant words.



(Arras et al. 2017)

Compare Explanation Methods



- word2vec / CNN model
- Conv → ReLU → 1-Max-Pool → FC
- trained on 20Newsgroup Dataset
- accuracy: 80.19%

LRP better than SA

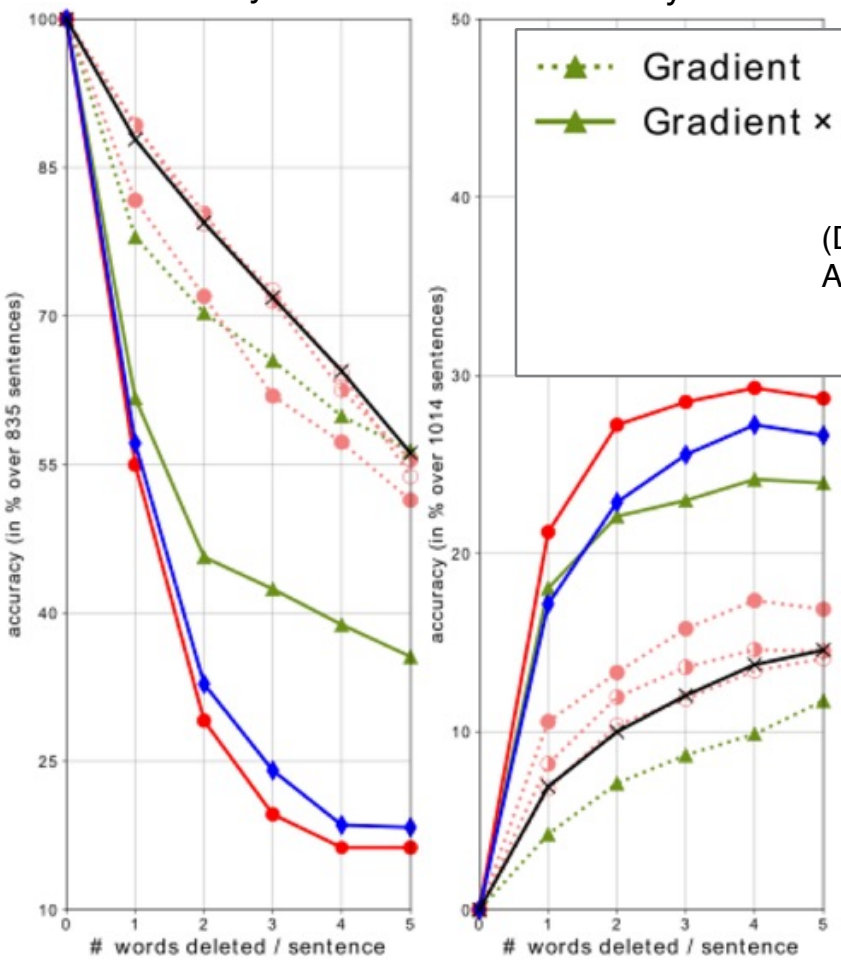
LRP distinguishes between positive and negative evidence

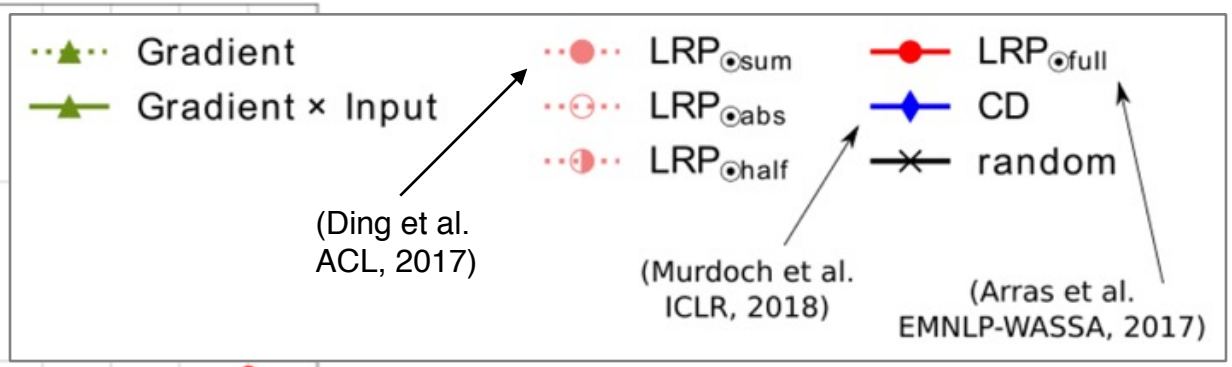
(Arras et al. 2016)

Compare Explanation Methods

Deleting most relevant from correctly classified

Deleting least relevant from falsely classified





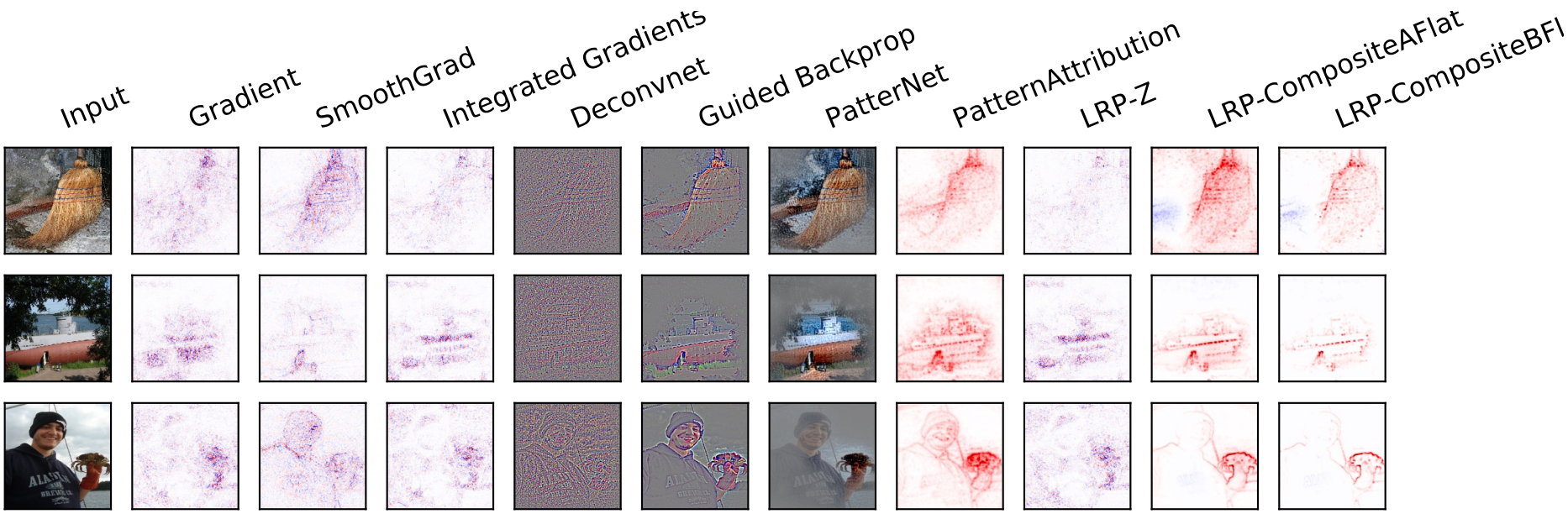
- bidirectional LSTM model (Li'16)
- Stanford Sentiment Treebank dataset
- delete up to 5 words per sentence

LRP outperforms baselines (also recently proposed contextual decomposition)

LRP ≠ Gradient x Input

(Arras et al. 2018)

Compare Explanation Methods



Highly efficient (e.g., 0.01 sec per VGG16 explanation) !

New Keras Toolbox available for explanation methods:
`https://github.com/albermax/innvestigate`

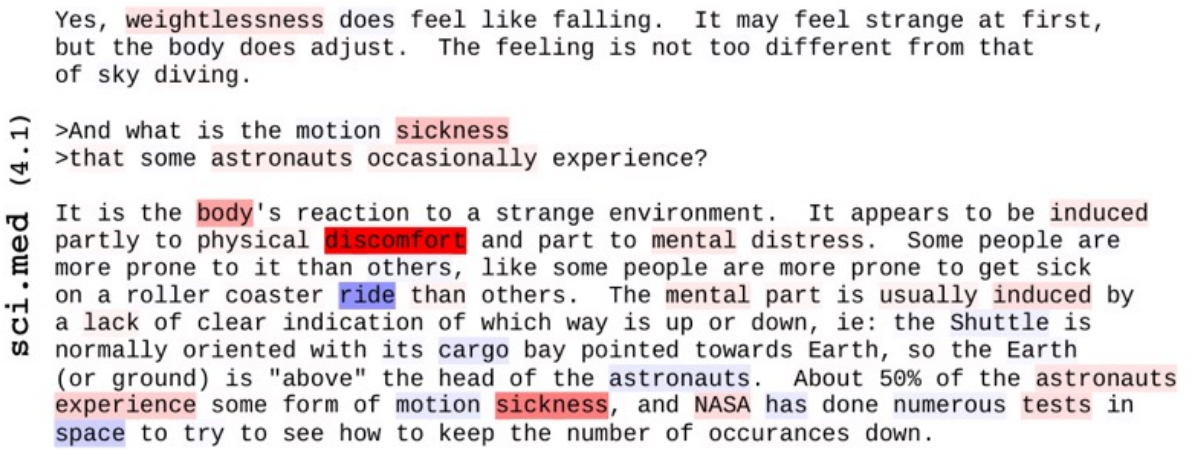
Application of LRP
Compare models

Application: Compare Classifiers

word2vec/CNN:

Performance: 80.19%

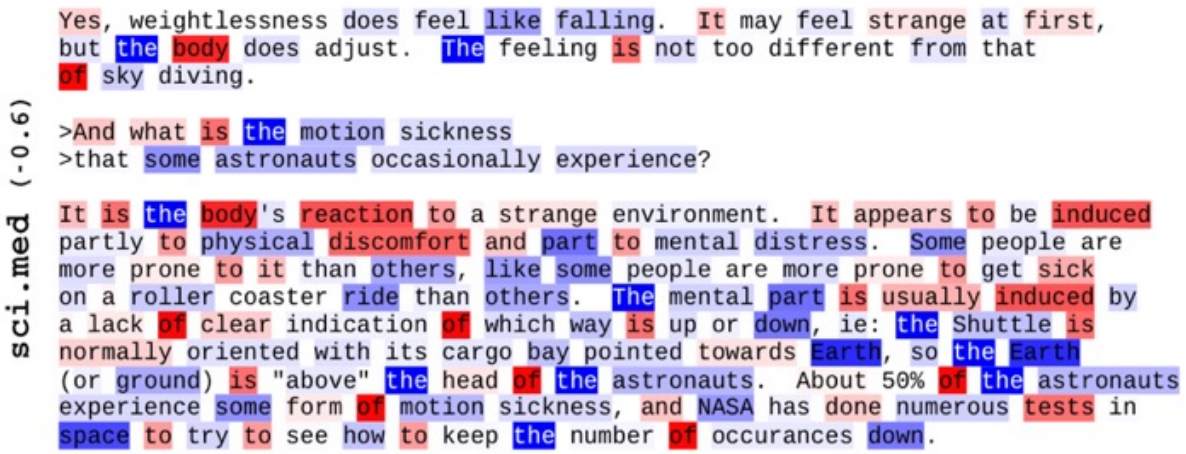
Strategy to solve the problem: identify semantically meaningful words related to the topic.



BoW/SVM:

Performance: 80.10%

Strategy to solve the problem: identify statistical patterns, i.e., use word statistics



(Arras et al. 2016 & 2017)

Application: Compare Classifiers

word2vec / CNN model

sci.med

symptoms (7.3), treatments (6.6), medication (6.4), osteopathy (6.3), ulcers (6.2), sciatica (6.0), hypertension (6.0), herb (5.6), doctor (5.4), physician (5.1), Therapy (5.1), antibiotics (5.1), Asthma (5.0), renal (5.0), medicines (4.9), caffeine (4.9), infection (4.9), gastrointestinal (4.8), therapy (4.8), homeopathic (4.7), medicine (4.7), allergic (4.7), dosages (4.7), esophagitis (4.7), inflammation (4.6), arrhythmias (4.6), cancer (4.6), disease (4.6), migraine (4.6), patients (4.5).

BoW/SVM model

sci.med

cancer (1.4), photography (1.0), doctor (1.0), msg (0.9), disease (0.9), medical (0.8), sleep (0.8), radiologist (0.7), eye (0.7), treatment (0.7), prozac (0.7), vitamin (0.7), epilepsy (0.7), health (0.6), yeast (0.6), skin (0.6), pain (0.5), liver (0.5), physician (0.5), she (0.5), needles (0.5), dn (0.5), circumcision (0.5), syndrome (0.5), migraine (0.5), antibiotic (0.5), water (0.5), blood (0.5), fat (0.4), weight (0.4).

Words with maximum relevance

(Arras et al. 2016 & 2017)

LRP in Practice

Visual Object Classes Challenge: 2005 - 2012

	aeroplane	bicycle	bird	boat	bottle	bus	car	cat	chair	cow	diningtable	dog	horse	motorbike	person	pottedplant	sheep	sofa	train	tvmonitor
INRIA_Flat	*74.8*	*62.5*	*51.2*	*69.4*	*29.2*	*60.4*	*76.3*	*57.6*	*53.1*	*41.1*	*54.0*	*42.8*	*76.5*	*62.3*	*84.5*	*35.3*	*41.3*	*50.1*	*77.6*	*49.3*
INRIA_Genetic	**77.5**	**63.6**	**56.1**	**71.9**	**33.1**	**60.6**	**78.0**	**58.8**	**53.5**	**42.6**	**54.9**	**45.8**	**77.5**	**64.0**	**85.9**	**36.3**	**44.7**	*50.6*	**79.2**	**53.2**
INRIA_Larlus	62.6	54.0	32.8	47.5	17.8	46.4	69.6	44.2	44.6	26.0	38.1	34.0	66.0	55.1	77.2	13.1	29.1	36.7	62.7	43.3
MPI_BOW	58.9	46.0	31.3	59.0	16.9	40.5	67.2	40.2	44.3	28.3	31.9	34.4	63.6	53.5	75.7	22.3	26.6	35.4	60.6	40.6
PRIPUVA	48.6	20.9	21.3	17.2	6.4	14.2	45.0	31.4	27.4	12.3	14.3	23.7	30.1	13.3	62.0	10.0	12.4	13.3	26.7	26.2
QMUL_HSLS	70.6	54.8	35.7	64.5	27.8	51.1	71.4	54.0	46.6	36.6	34.4	39.9	71.5	55.4	80.6	15.8	35.8	41.5	73.1	45.5
QMUL_LSPCH	71.6	55.0	41.1	65.5	27.2	51.1	72.2	*55.1*	47.4	35.9	37.4	41.5	71.5	57.9	80.8	15.6	33.3	41.9	*76.5*	45.9
TKK	71.4	51.7	48.5	63.4	27.3	49.9	70.1	51.2	51.7	32.3	46.3	41.5	72.6	*60.2*	82.2	31.7	30.1	39.2	71.1	41.0
ToshCam_rdf	59.9	36.8	29.9	40.0	23.6	33.3	60.2	33.0	41.0	17.8	33.2	33.7	63.9	53.1	77.9	29.0	27.3	31.2	50.1	37.6
ToshCam_svm	54.0	27.1	30.3	35.6	17.0	22.3	58.0	34.6	38.0	19.0	27.5	32.4	48.0	40.7	78.1	23.4	21.8	28.0	45.5	31.8
Tsinghua	62.9	42.4	33.9	49.7	23.7	40.7	62.0	35.2	42.7	21.0	38.9	34.7	65.0	48.1	76.9	16.9	30.8	32.8	58.9	33.1
UVA_Bigrams	61.2	33.2	29.4	45.0	16.5	37.6	54.6	31.3	39.9	17.2	31.4	30.6	61.6	42.4	74.6	14.5	20.9	23.5	49.9	30.0
UVA_FuseAll	67.1	48.1	43.3	58.1	19.9	46.3	61.8	41.9	48.4	27.8	41.9	38.5	69.8	51.4	79.4	32.5	31.9	36.0	66.2	40.3
UVA_MCIP	66.5	47.9	41.0	58.0	16.8	44.0	61.2	40.5	48.5	27.8	41.7	37.1	66.4	50.1	78.6	31.2	32.3	31.9	66.6	40.3
UVA_SFS	66.3	49.7	43.5	60.7	18.8	44.9	64.8	41.9	46.8	24.9	42.3	33.9	71.5	53.4	80.4	29.7	31.2	31.8	67.4	43.5
UVA_WGT	59.7	33.7	34.9	44.5	22.2	32.9	55.9	36.3	36.8	20.6	25.2	34.7	65.1	40.1	74.2	26.4	26.9	25.1	50.7	29.7
XRCE	*72.3*	*57.5*	*53.2*	*68.9*	*28.5*	*57.5*	*75.4*	50.3	*52.2*	*39.0*	*46.8*	*45.3*	*75.7*	58.5	*84.0*	*32.6*	*39.7*	**50.9**	75.1	*49.5*

Application: Compare Classifiers

Test error for various classes:

	aeroplane	bicycle	bird	boat	bottle	bus	car
Fisher	79.08%	66.44%	45.90%	70.88%	27.64%	69.67%	80.96%
DeepNet	88.08%	79.69%	80.77%	77.20%	35.48%	72.71%	86.30%
	cat	**chair**	**cow**	**diningtable**	**dog**	**horse**	**motorbike**
Fisher	59.92%	51.92%	47.60%	58.06%	42.28%	80.45%	69.34%
DeepNet	81.10%	51.04%	61.10%	64.62%	76.17%	81.60%	79.33%
	person	**pottedplant**	**sheep**	**sofa**	**train**	**tvmonitor**	**mAP**
Fisher	85.10%	28.62%	49.58%	49.31%	82.71%	54.33%	59.99%
DeepNet	92.43%	49.99%	74.04%	49.48%	87.07%	67.08%	72.12%

(Lapuschkin et al. 2016)

Application: Compare Classifiers

Test error for various classes:

	aeroplane	bicycle	bird	boat	bottle	bus	car
Fisher	79.08%	66.44%	45.90%	70.88%	27.64%	69.67%	80.96%
DeepNet	88.08%	79.69%	80.77%	77.20%	35.48%	72.71%	86.30%
	cat	**chair**	**cow**	**diningtable**	**dog**	**horse**	**motorbike**
Fisher	59.92%	51.92%	47.60%	58.06%	42.28%	80.45%	69.34%
DeepNet	81.10%	51.04%	61.10%	64.62%	76.17%	81.60%	79.33%
	person	**pottedplant**	**sheep**	**sofa**	**train**	**tvmonitor**	**mAP**
Fisher	85.10%	28.62%	49.58%	49.31%	82.71%	54.33%	59.99%
DeepNet	92.43%	49.99%	74.04%	49.48%	87.07%	67.08%	72.12%

same performance —> same strategy ?

(Lapuschkin et al. 2016)

Application: Compare Classifiers

Test error for various classes:

	aeroplane	bicycle	bird	boat	bottle	bus	car
Fisher	79.08%	66.44%	45.90%	70.88%	27.64%	69.67%	80.96%
DeepNet	88.08%	79.69%	80.77%	77.20%	35.48%	72.71%	86.30%
	cat	**chair**	**cow**	**diningtable**	**dog**	**horse**	**motorbike**
Fisher	59.92%	51.92%	47.60%	58.06%	42.28%	80.45%	69.34%
DeepNet	81.10%	51.04%	61.10%	64.62%	76.17%	81.60%	79.33%
	person	**pottedplant**	**sheep**	**sofa**	**train**	**tvmonitor**	**mAP**
Fisher	85.10%	28.62%	49.58%	49.31%	82.71%	54.33%	59.99%
DeepNet	92.43%	49.99%	74.04%	49.48%	87.07%	67.08%	72.12%

Image

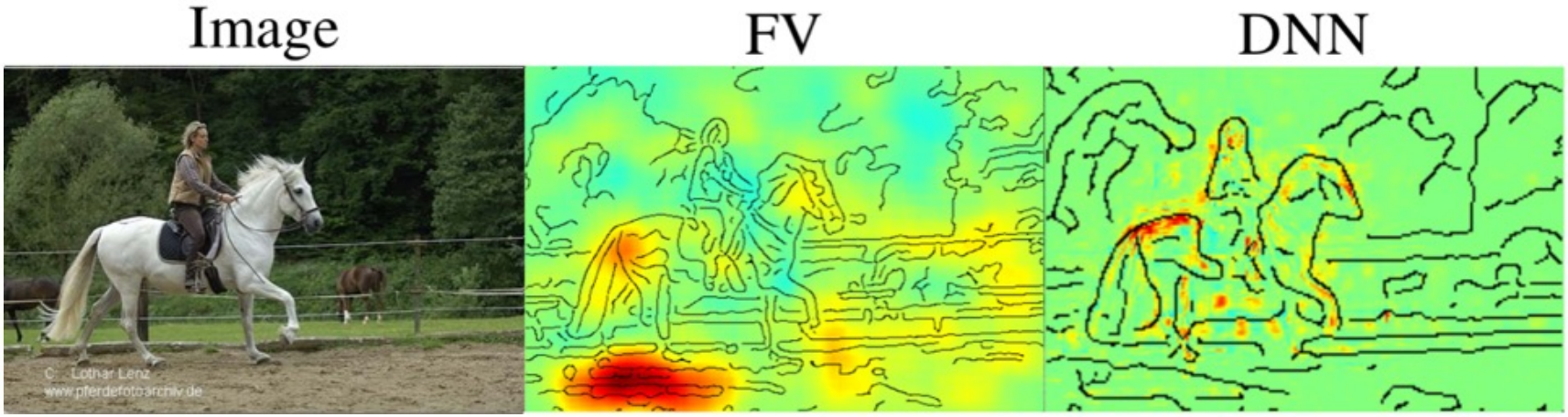
same performance —> same strategy ?

(Lapuschkin et al. 2016)

Application: Compare Classifiers

Test error for various classes:

	aeroplane	bicycle	bird	boat	bottle	bus	car
Fisher	79.08%	66.44%	45.90%	70.88%	27.64%	69.67%	80.96%
DeepNet	88.08%	79.69%	80.77%	77.20%	35.48%	72.71%	86.30%
	cat	**chair**	**cow**	**diningtable**	**dog**	**horse**	**motorbike**
Fisher	59.92%	51.92%	47.60%	58.06%	42.28%	80.45%	69.34%
DeepNet	81.10%	51.04%	61.10%	64.62%	76.17%	81.60%	79.33%
	person	**pottedplant**	**sheep**	**sofa**	**train**	**tvmonitor**	**mAP**
Fisher	85.10%	28.62%	49.58%	49.31%	82.71%	54.33%	59.99%
DeepNet	92.43%	49.99%	74.04%	49.48%	87.07%	67.08%	72.12%

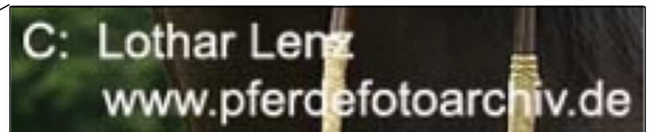


same performance —> same strategy ?

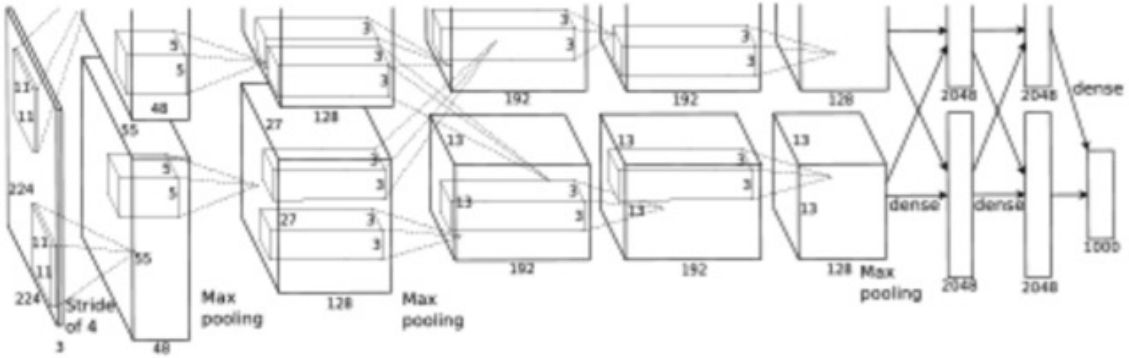
(Lapuschkin et al. 2016)

Application: Compare Classifiers

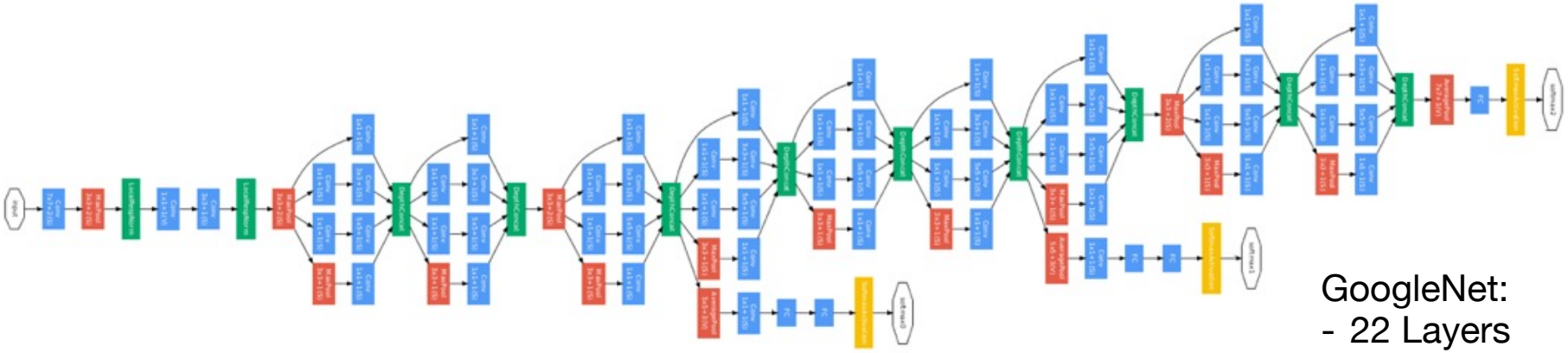
‘horse’ images in PASCAL VOC 2007



Application: Compare Classifiers



BVLC:
- 8 Layers
- ILSRCV: 16.4%



GoogleNet:
- 22 Layers
- ILSRCV: 6.7%
- Inception layers

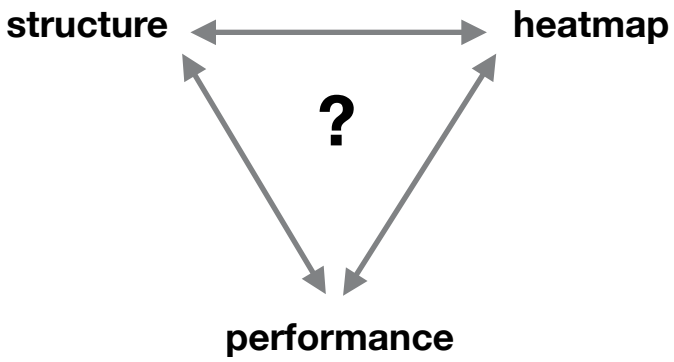
Application: Compare Classifiers

GoogleNet focuses on faces of animal.
—> suppresses background noise

BVLC CaffeNet heatmaps are much more noisy.

Is it related to the architecture ?

Is it related to the performance ?



(Binder et al. 2016)

Application of LRP
Quantify Context Use

Application: Measure Context Use

how important is context ?

how important is context ?

classifier

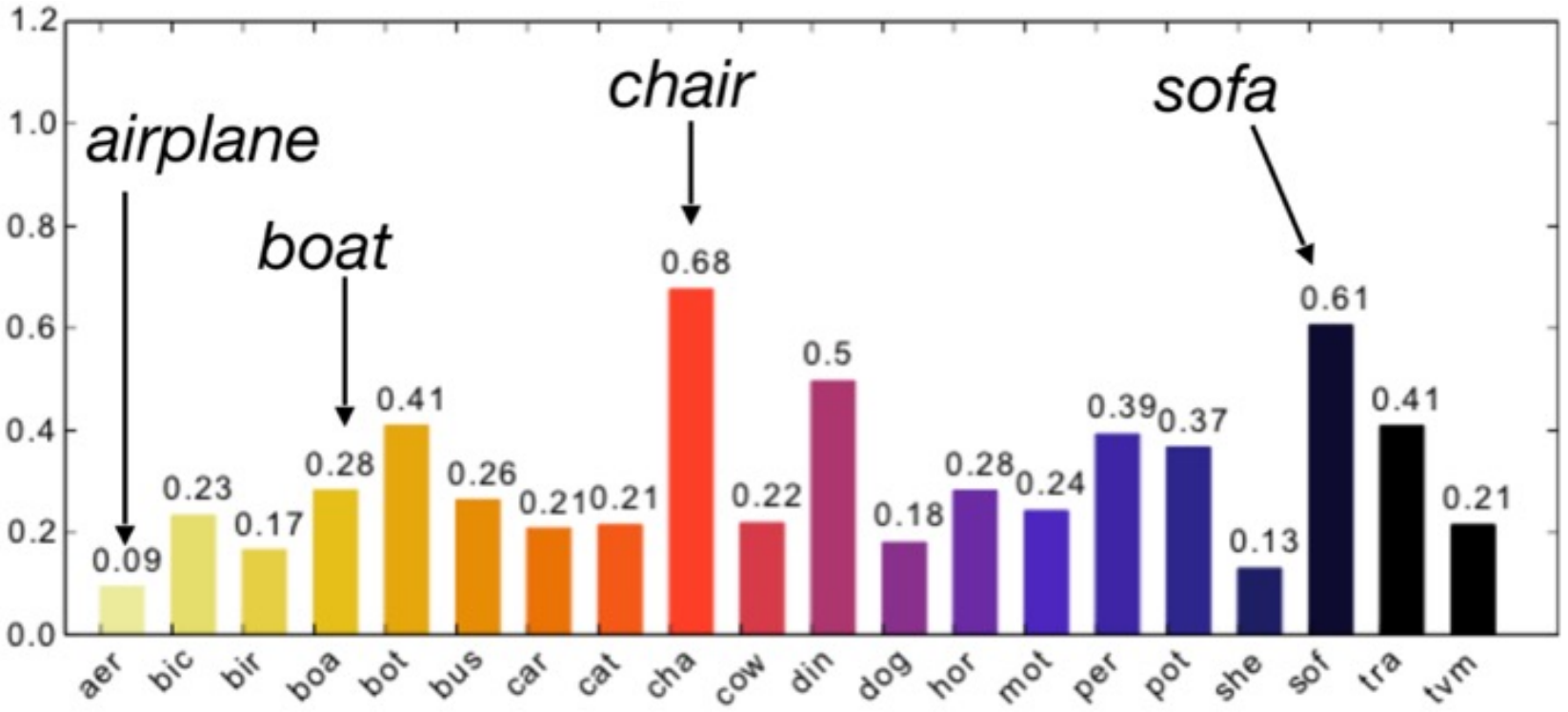
LRP decomposition allows meaningful pooling over bbox !

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

$$\text{importance of context} = \frac{\text{relevance outside bbox}}{\text{relevance inside bbox}}$$

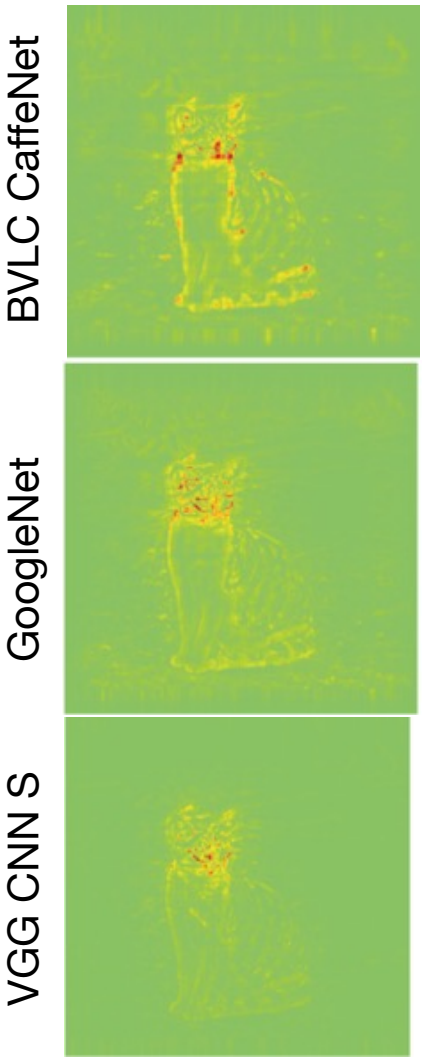
Application: Measure Context Use

- BVLC reference model + fine tuning
- PASCAL VOC 2007

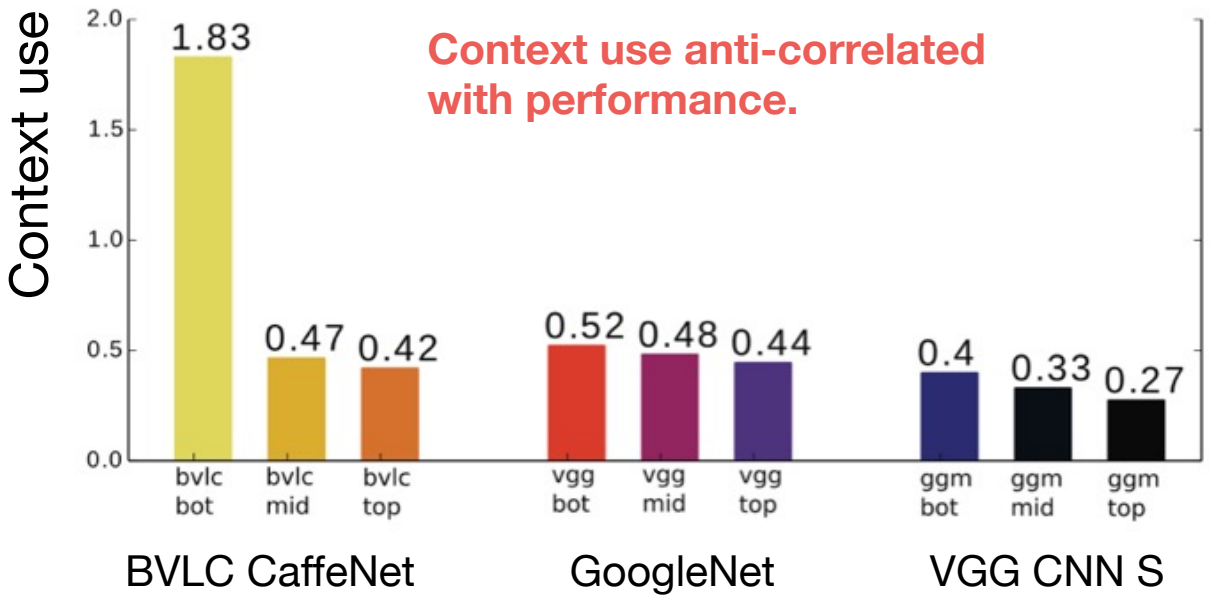


(Lapuschkin et al., 2016)

Application: Measure Context Use



- Differen models (BVLC CaffeNet, GoogleNet, VGG CNN S)
- ILSVCR 2012



Context use anti-correlated with performance.

(Lapuschkin et al. 2016)

Application of LRP
Detect Biases & Improve Models

Application: Face analysis

- Compare AdienceNet, CaffeNet, GoogleNet, VGG-16
- Adience dataset, 26,580 images

Age classification

	A	C	G	V
[i]	51.4 87.0	52.1 87.9	54.3 89.1	–
[r]	51.9 87.4	52.3 88.9	53.3 89.9	–
[m]	53.6 88.4	54.3 89.7	56.2 90.7	–
[i,n]	–	51.6 87.4	56.2 90.9	53.6 88.2
[r,n]	–	52.1 87.0	57.4 91.9	–
[m,n]	–	52.8 88.3	58.5 92.6	56.5 90.0
[i,w]	–	–	–	59.7 94.2
[r,w]	–	–	–	–
[m,w]	–	–	–	62.8 95.8

Gender classification

	A	C	G	V
[i]	88.1	87.4	87.9	–
[r]	88.3	87.8	88.9	–
[m]	89.0	88.8	89.7	–
[i,n]	–	89.9	**91.0**	**92.0**
[r,n]	–	90.6	**91.6**	–
[m,n]	–	90.6	**91.7**	**92.6**
[i,w]	–	–	–	90.5
[r,w]	–	–	–	–
[m,w]	–	–	–	**92.2**

A = AdienceNet
C = CaffeNet
G = GoogleNet
V = VGG-16

[i] = in-place face alignment
[r] = rotation based alignment
[m] = mixing aligned images for training
[n] = initialization on Imagenet
[w] = initialization on IMDB-WIKI

(Lapuschkin et al., 2017)

Application: Face analysis

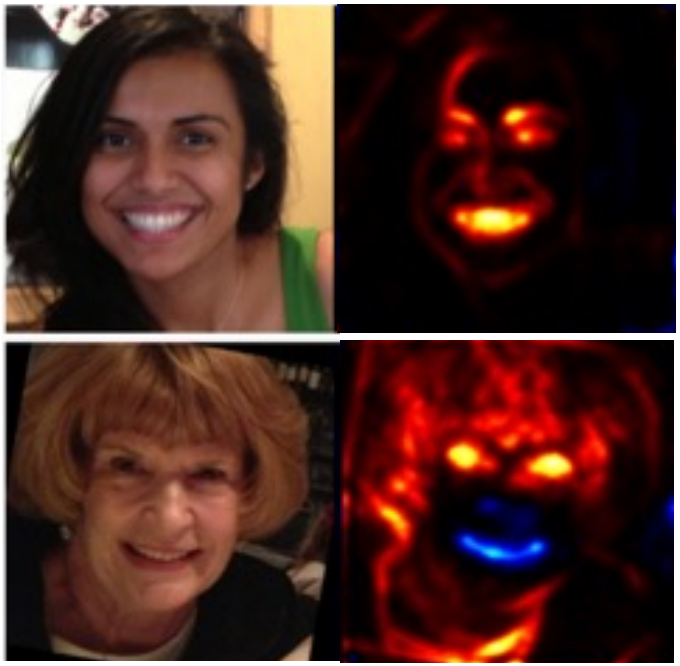
Gender classification

Strategy to solve the problem: Focus on chin / beard, eyes & hear, but without pretraining the model overfits

(Lapuschkin et al., 2017)

Application: Face analysis

Age classification



Predictions

25-32 years old

Strategy to solve the problem:
Focus on the laughing …

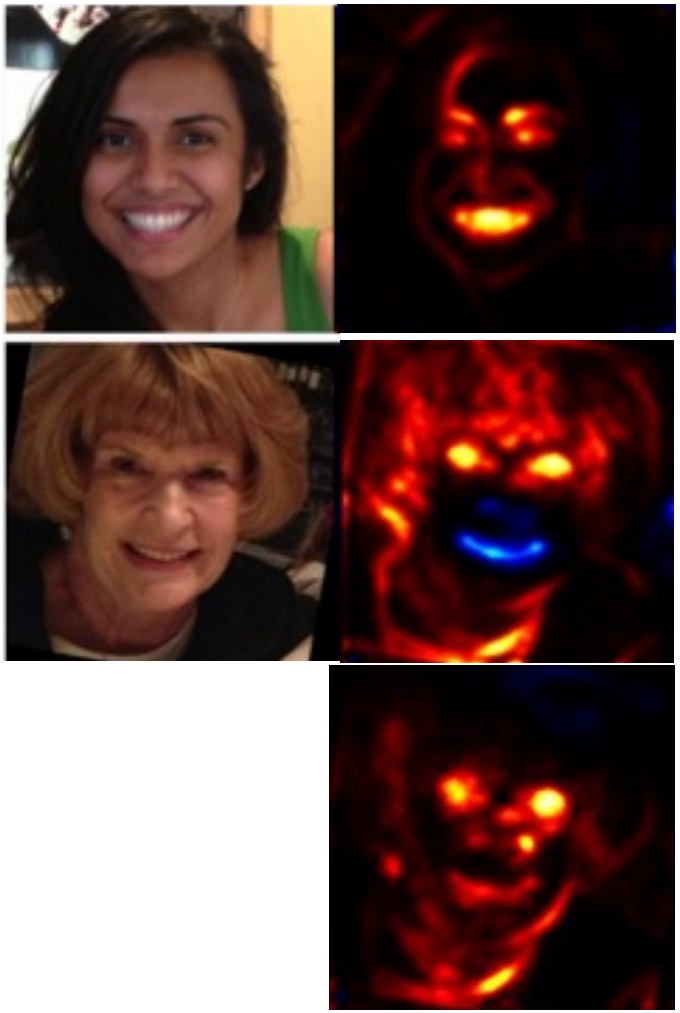
60+ years old

laughing speaks against 60+ (i.e., model learned that old people do not laugh)

(Lapuschkin et al., 2017)

Application: Face analysis

Age classification



Predictions

25-32 years old

<u>Strategy to solve the problem</u>:
Focus on the laughing …

60+ years old

pretraining on ImageNet

laughing speaks against 60+ (i.e., model learned that old people do not laugh)

pretraining on IMDB-WIKI

(Lapuschkin et al., 2017)

Application: Face analysis

real person | fake person | real person

- 1,900 images of different individuals
- pretrained VGG19 model
- different ways to train the models

Different training methods

	naive	one morphed	complex morphs	multiclass
true positive	95%	90%	93%	92%
true negative	98%	95%	95%	99%
EER	3.1%	7.2%	6.1%	2.8%

50% genuine images,
50% complete morphs

50% genuine images,
10% complete morphs and
4 × 10% one region morphed

50% genuine images,
10% complete morphs,
partial morphs with 10%
one, two, three and four
region morphed

partial morphs with zero,
one, two, three or four
morphed regions,
for two class classification
last layer reinitialized

(Seibold et al., 2018)

Application: Face analysis

Semantic attack on the model

Table 4. Robustness against partial morphs.

	left eye	right eye	nose	mouth	average
naive	25%	21%	14%	13%	20%
one morphed	81%	89%	79%	71%	80%
complex morphs	78%	74%	73%	54%	70%
multiclass	86%	93%	90%	79%	87%

Black box adversarial attack on the model

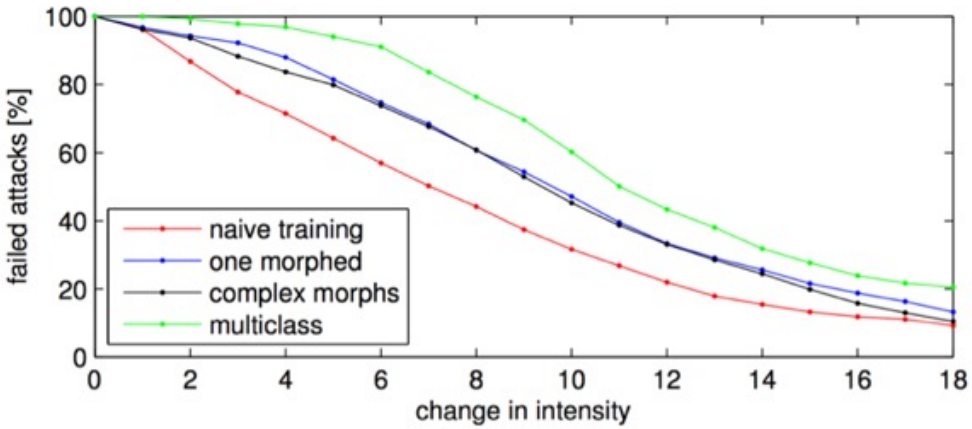


Fig. 5. Robustness against fast gradient sign attacks.

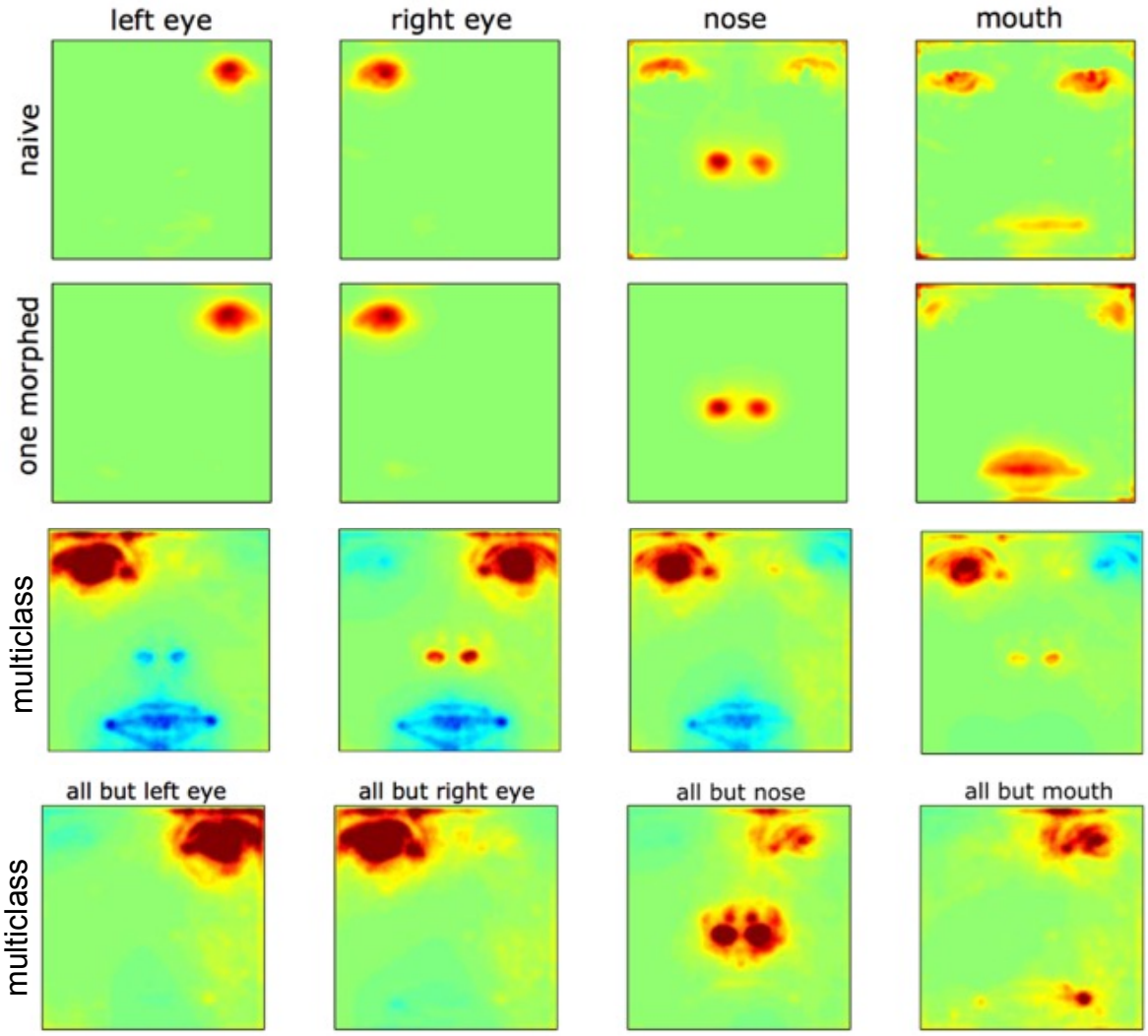
Application: Face analysis

morphed region	relative amount of relevance per region: naive				one morphed			
	left eye	right eye	nose	mouth	left eye	right eye	nose	mouth
left eye	0.84	0.00	0.02	0.14	0.96	0.00	0.01	0.04
right eye	0.00	0.91	0.05	0.05	0.00	0.92	0.01	0.07
nose	0.21	0.28	0.47	0.04	0.00	0.01	0.97	0.02
mouth	0.34	0.27	0.04	0.35	0.17	0.12	0.04	0.68

	complex morphs				multiclass			
	left eye	right eye	nose	mouth	left eye	right eye	nose	mouth
left eye	0.98	0.00	0.00	0.02	0.00	0.98	0.00	0.01
right eye	0.00	0.92	0.00	0.08	0.98	0.00	0.02	0.00
nose	0.02	0.03	0.92	0.02	0.01	0.10	0.19	0.70
mouth	0.06	0.00	0.41	0.53	0.11	0.18	0.58	0.13

(Seibold et al., 2018)

Application: Face analysis



Different models have different strategies !

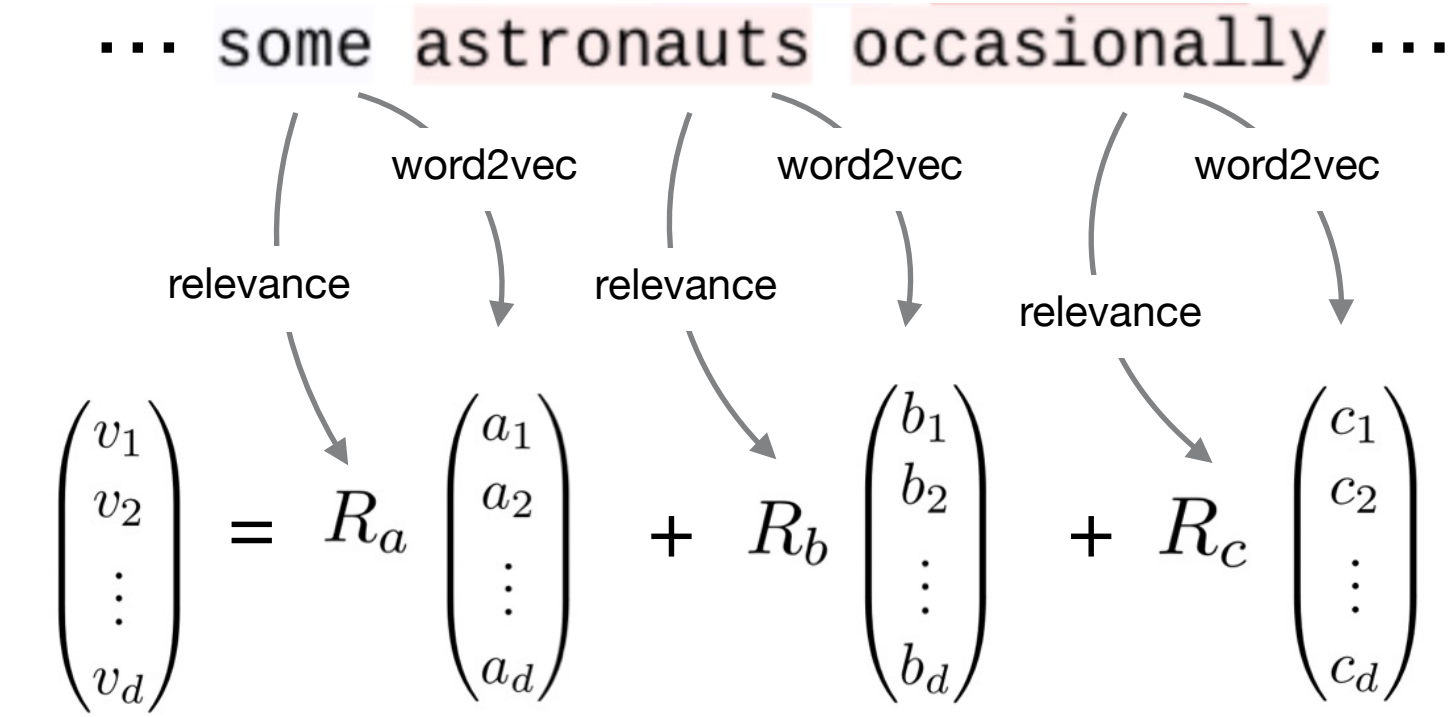
network seems to compare different structures

network seems to identify “original” parts

(Seibold et al., 2018)

Application of LRP
Learn new Representations

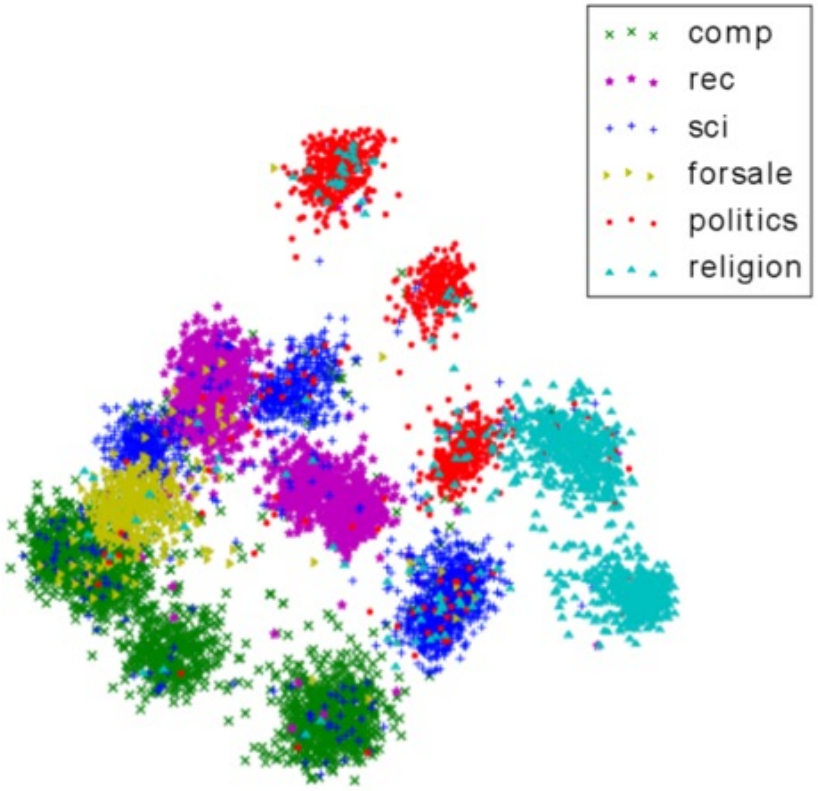
Application: Learn new Representations



(Arras et al. 2016 & 2017)

Application: Learn new Representations

2D PCA projection of document vectors



uniform

TFIDF

Document vector computation is unsupervised (given we have a classifier).

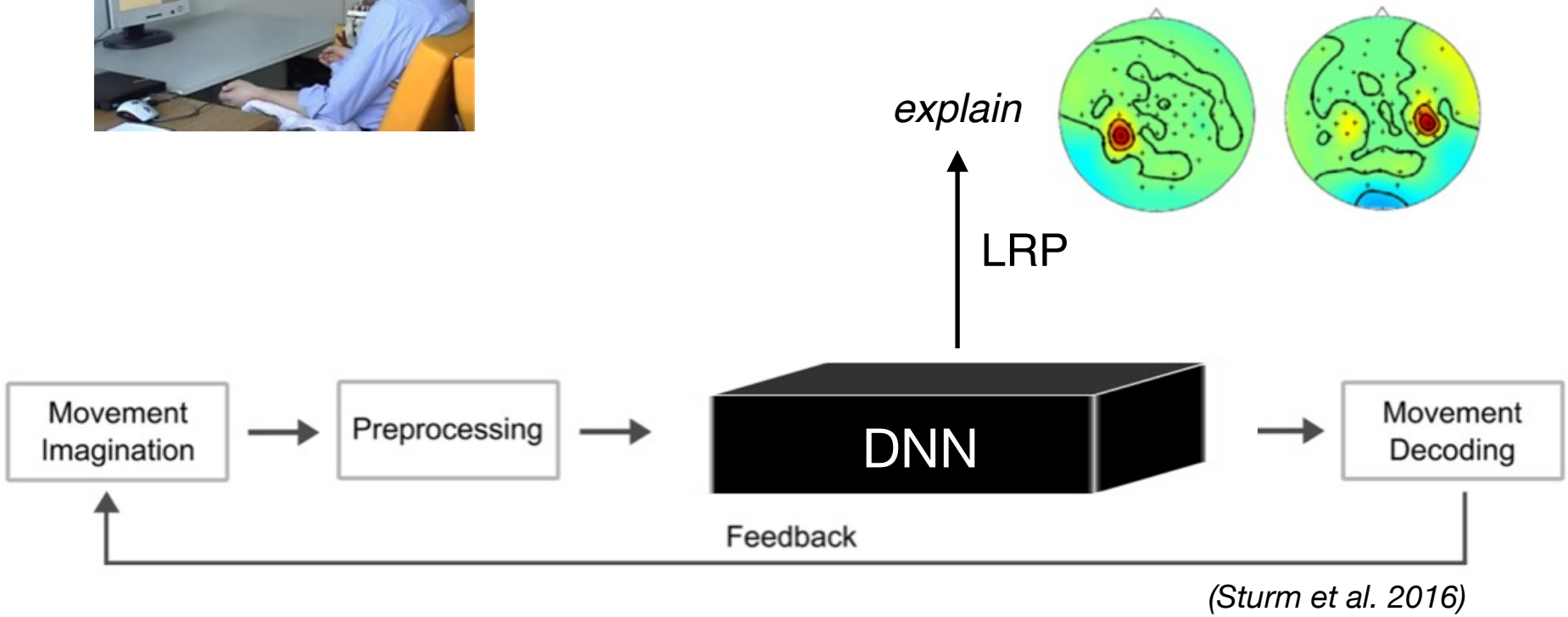
(Arras et al. 2016 & 2017)

Application of LRP
Interpreting Scientific Data

Application: EEG Analysis

Brain-Computer Interfacing

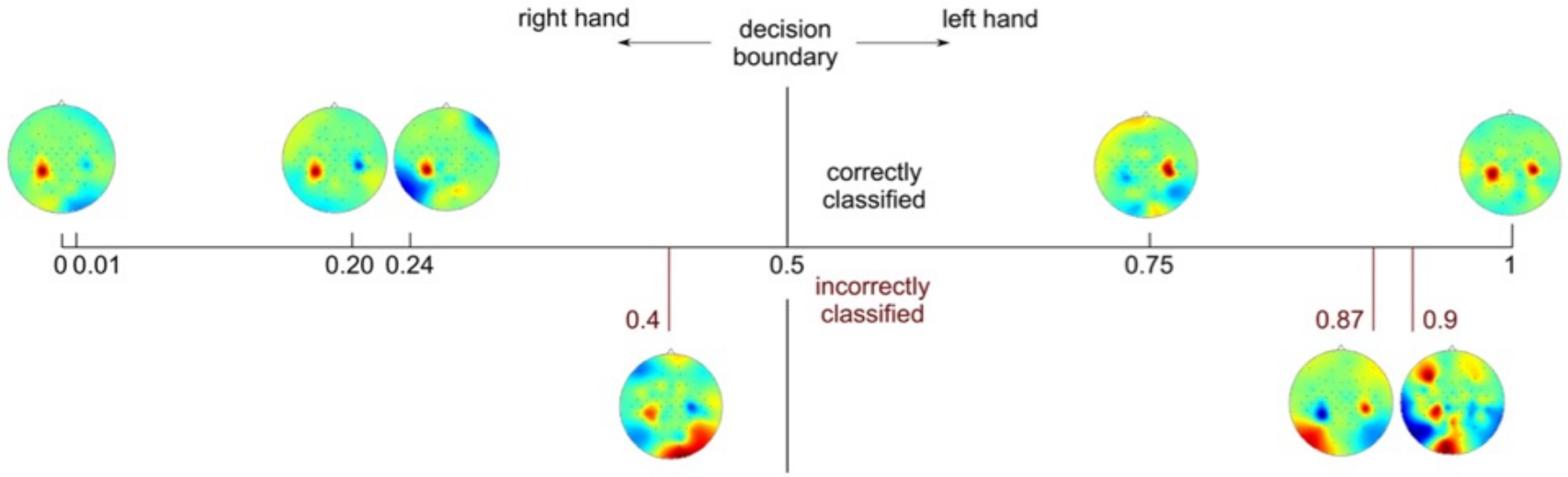
Neural network learns that:
Left hand movement imagination leads to desynchronization over right sensorimotor cortext (and vice versa).



(Sturm et al. 2016)

Application: EEG Analysis

Our neural networks are interpretable:
We can see for every trial "why" it is classified the way it is.



(Sturm et al. 2016)

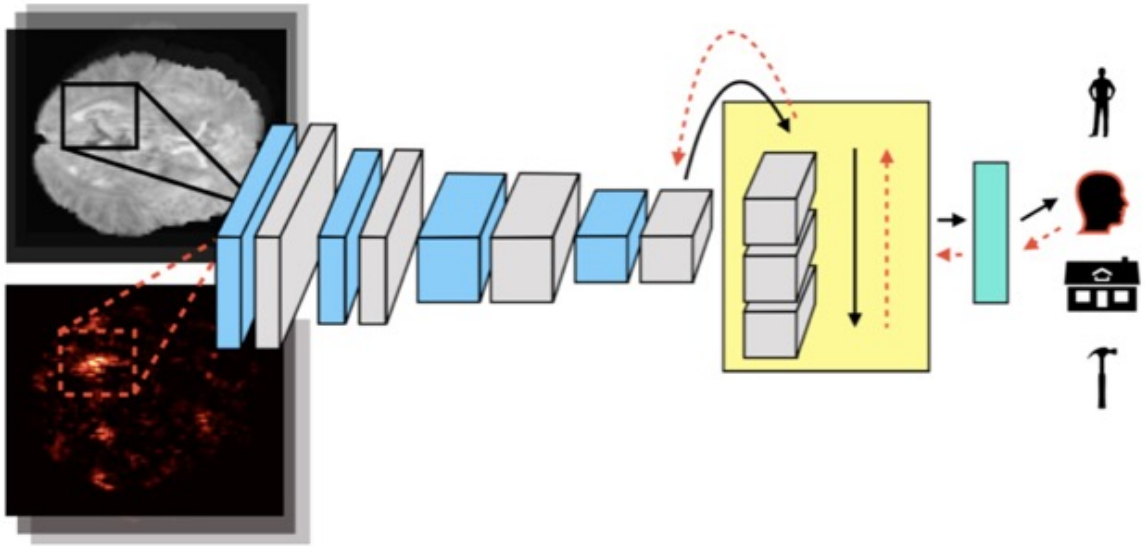
Application: fMRI Analysis

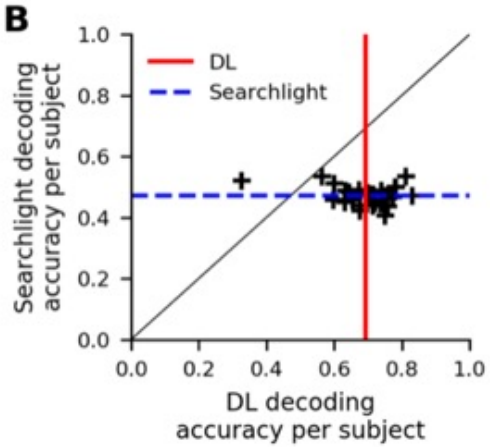
Difficulty to apply deep learning to fMRI :

- high dimensional data (100 000 voxels), but only few subjects
- results must be interpretable (key in neuroscience)

Our approach:

- Recurrent neural networks (CNN + LSTM) for whole-brain analysis
- LRP allows to interpret the results



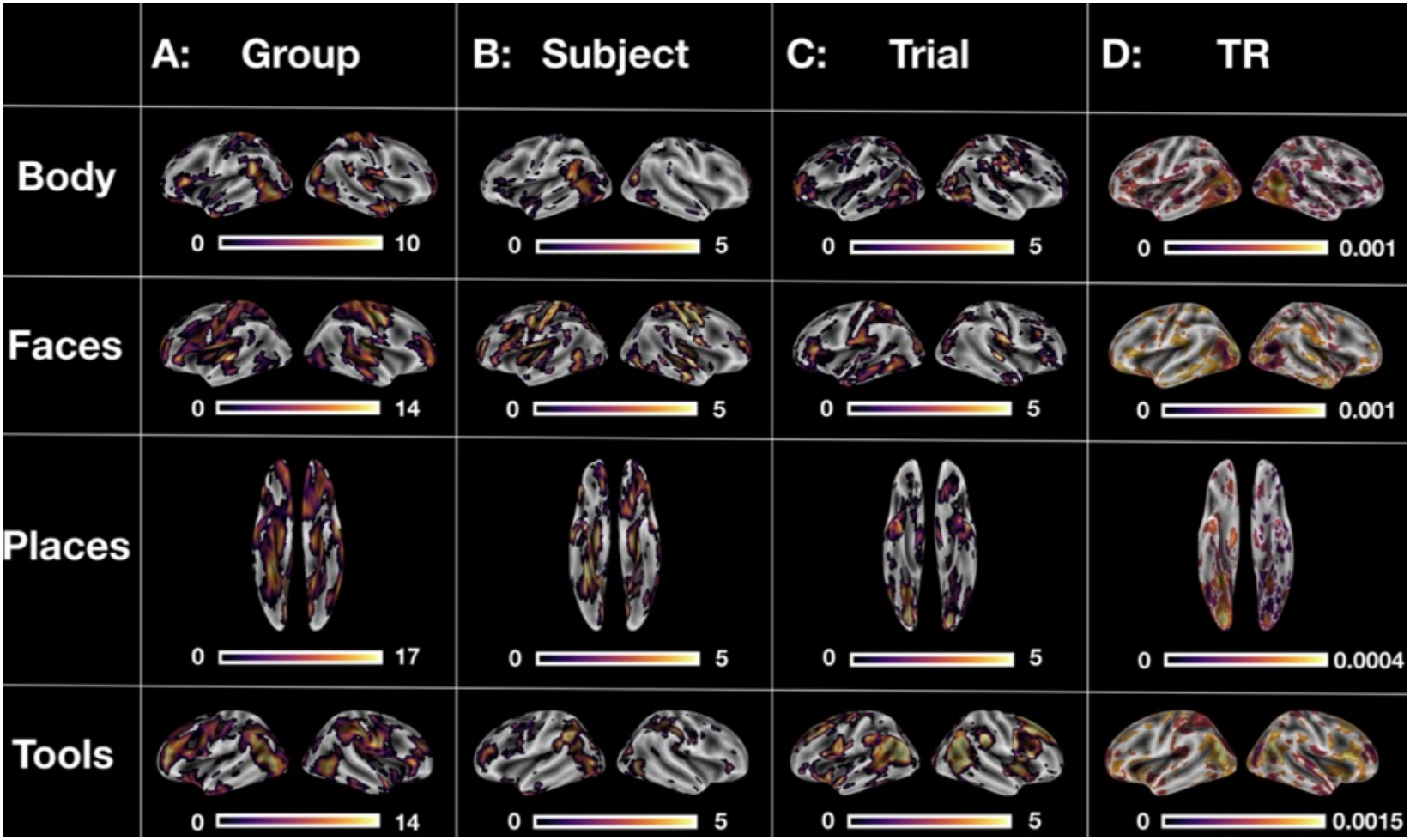


Dataset:

- 100 subjects from Human Connectome Project
- N-back task (faces, places, tools and body parts)

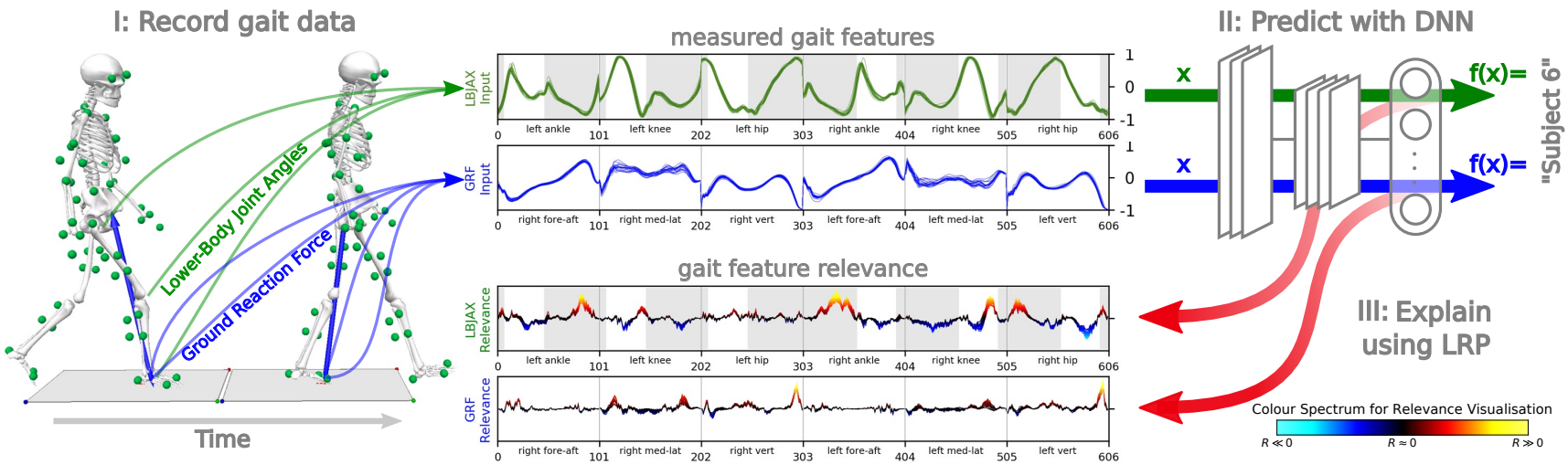
(Thomas et al. 2018)

Application: fMRI Analysis



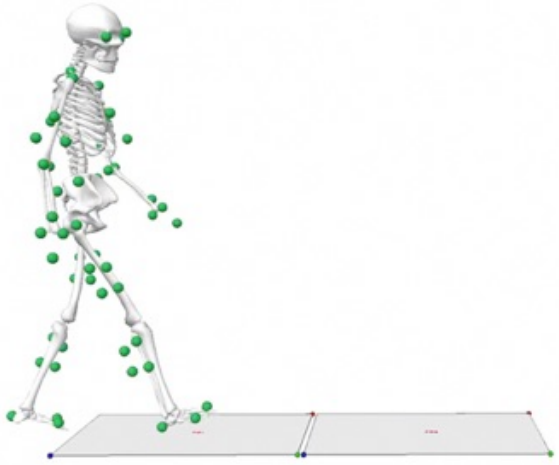
(Thomas et al. 2018)

Application: Gait Analysis



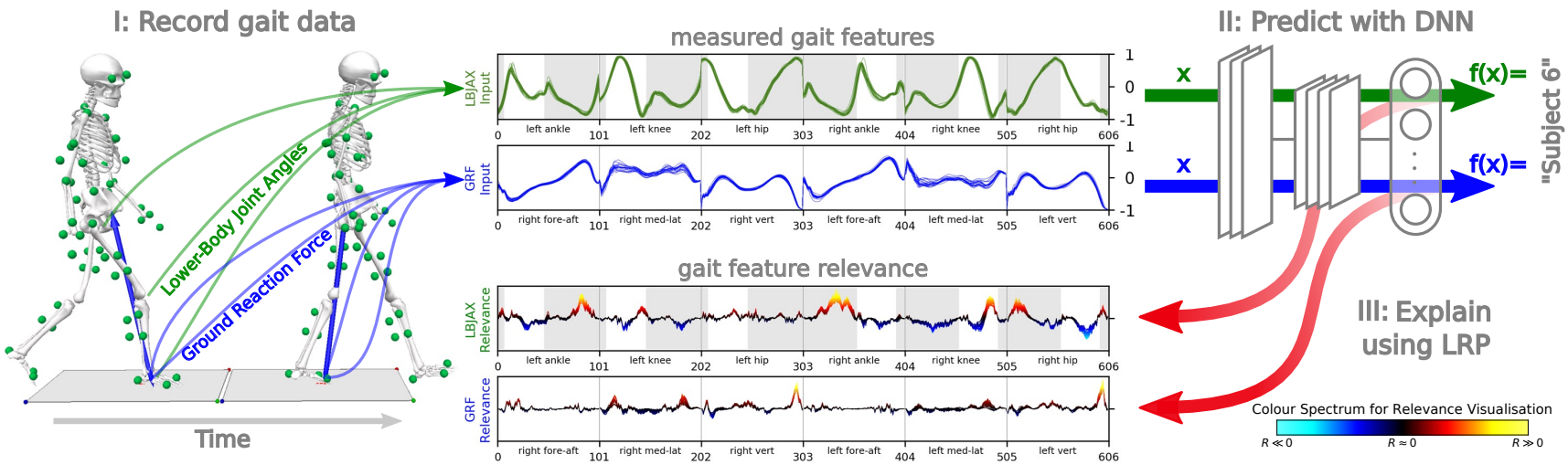
Our approach:

- Classify & explain individual gait patterns
- Important for understanding diseases such as Parkinson



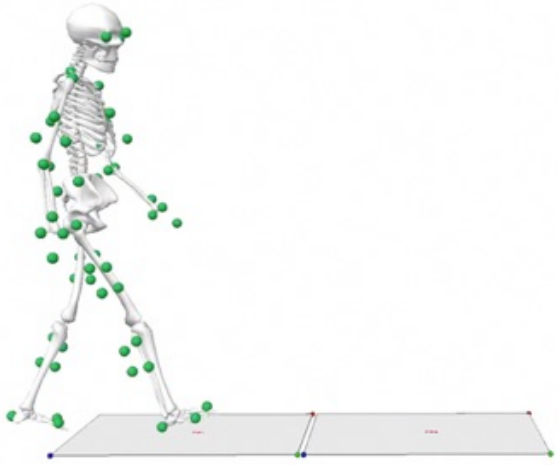
(Horst et al. 2018)

Application: Gait Analysis



Our approach:

- Classify & explain individual gait patterns
- Important for understanding diseases such as Parkinson

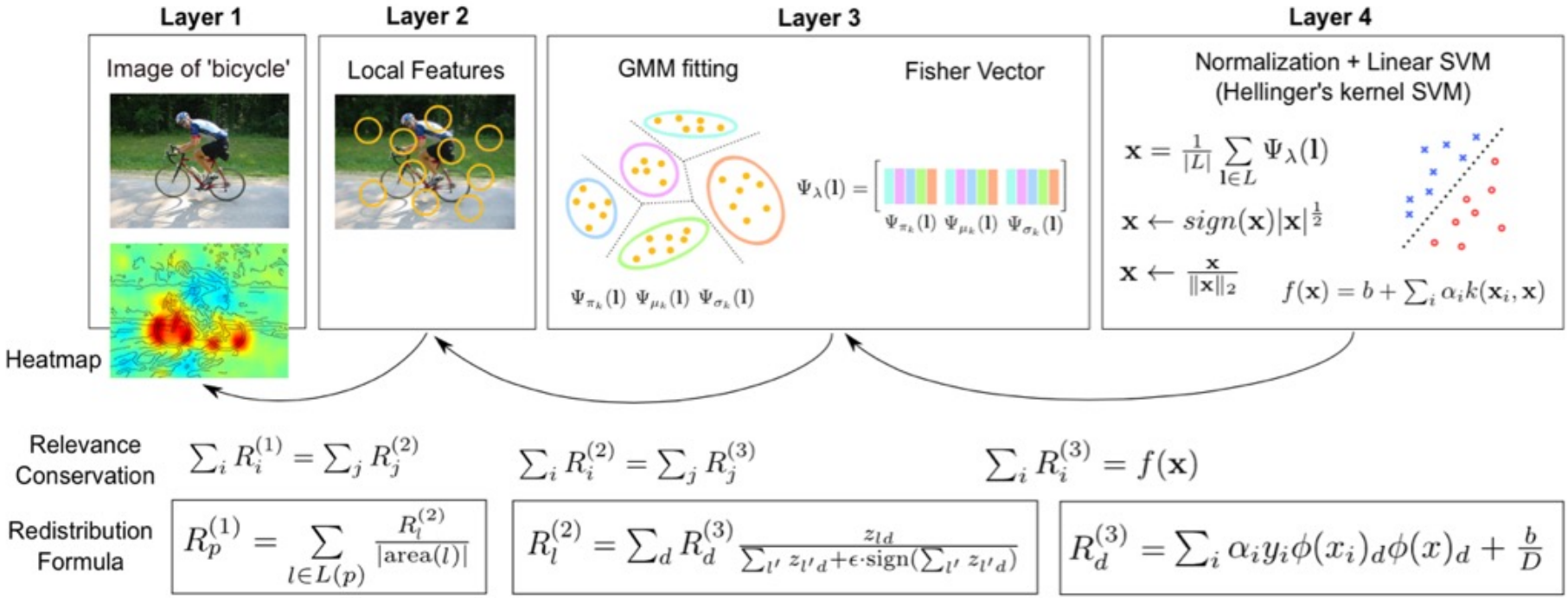


(Horst et al. 2018)

Application of LRP
Understand Model & Obtain new Insights

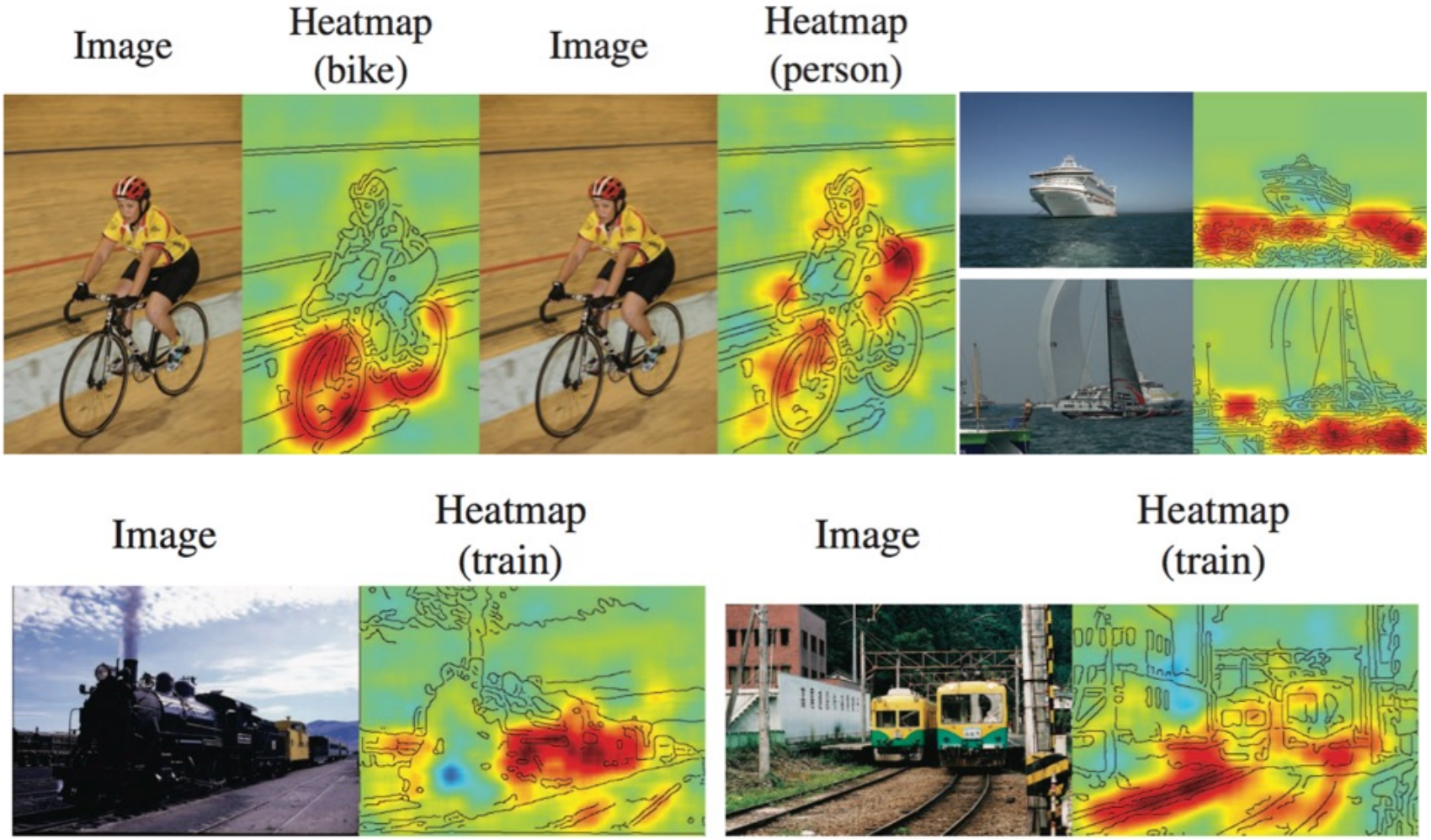
Application: Understand the model

- Fisher Vector / SVM classifier
- PASCAL VOC 2007



(Lapuschkin et al. 2016)

Application: Understand the model

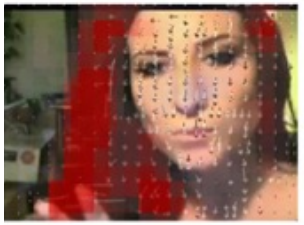


(Lapuschkin et al. 2016)

Application: Understand the model

Motion vectors can be extracted from the compressed video
-> allows very efficient analysis

- Fisher Vector / SVM classifier
- Model of Kantorov & Laptev, (CVPR'14)
- Histogram Of Flow, Motion Boundary Histogram
- HMDB51 dataset

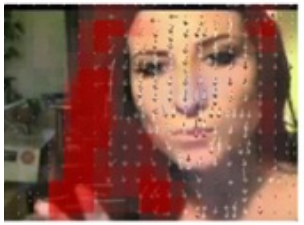


(Srinivasan et al. 2017)

Application: Understand the model

Motion vectors can be extracted from the compressed video
-> allows very efficient analysis

- Fisher Vector / SVM classifier
- Model of Kantorov & Laptev, (CVPR'14)
- Histogram Of Flow, Motion Boundary Histogram
- HMDB51 dataset



(Srinivasan et al. 2017)

Application: Understand the model

movie review:
++, −

- bidirectional LSTM model (Li'16)
- Stanford Sentiment Treebank dataset

How to handle multiplicative interactions ?

$$z_j = z_g \cdot z_s$$

$$R_g = 0 \quad R_s = R_j$$

gate neuron indirectly affect relevance distribution in forward pass

Negative sentiment

... too slow , too boring , and occasionally annoying .

it 's neither as romantic nor as thrilling as it should be .

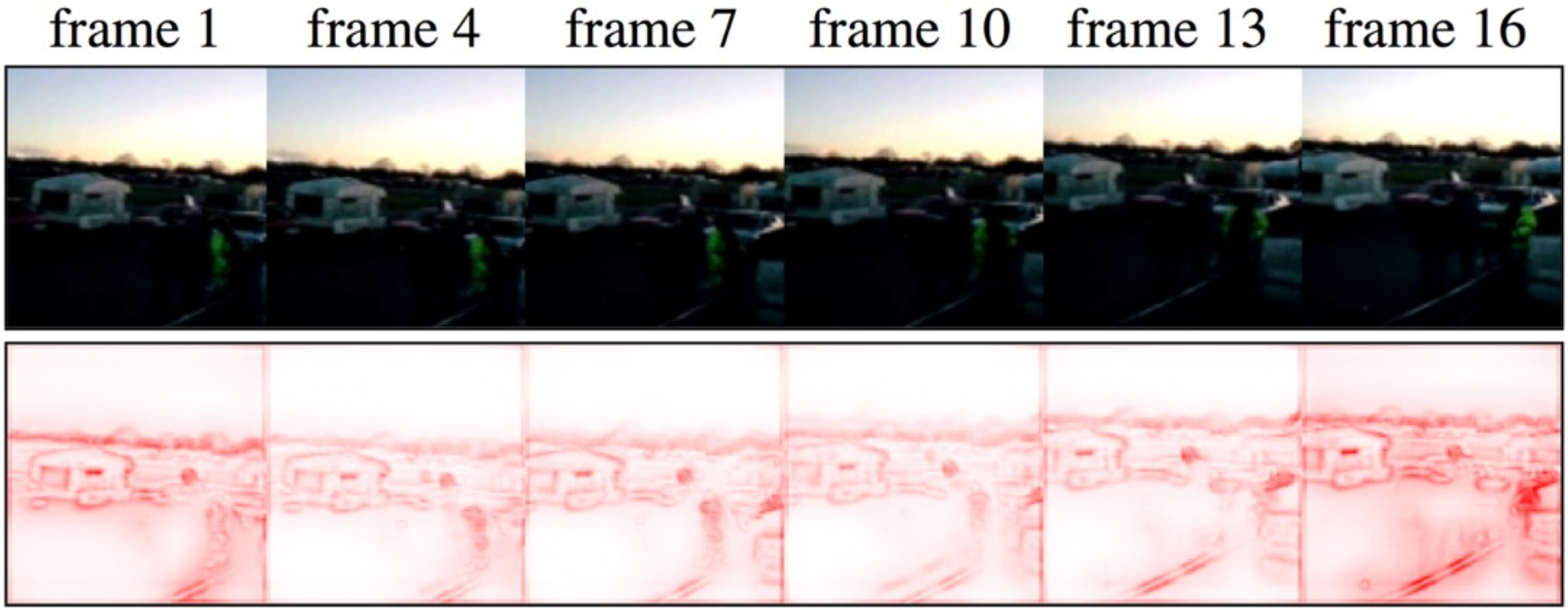
neither funny nor suspenseful nor particularly well-drawn .

Model understands negation !

(Arras et al., 2017 & 2018)

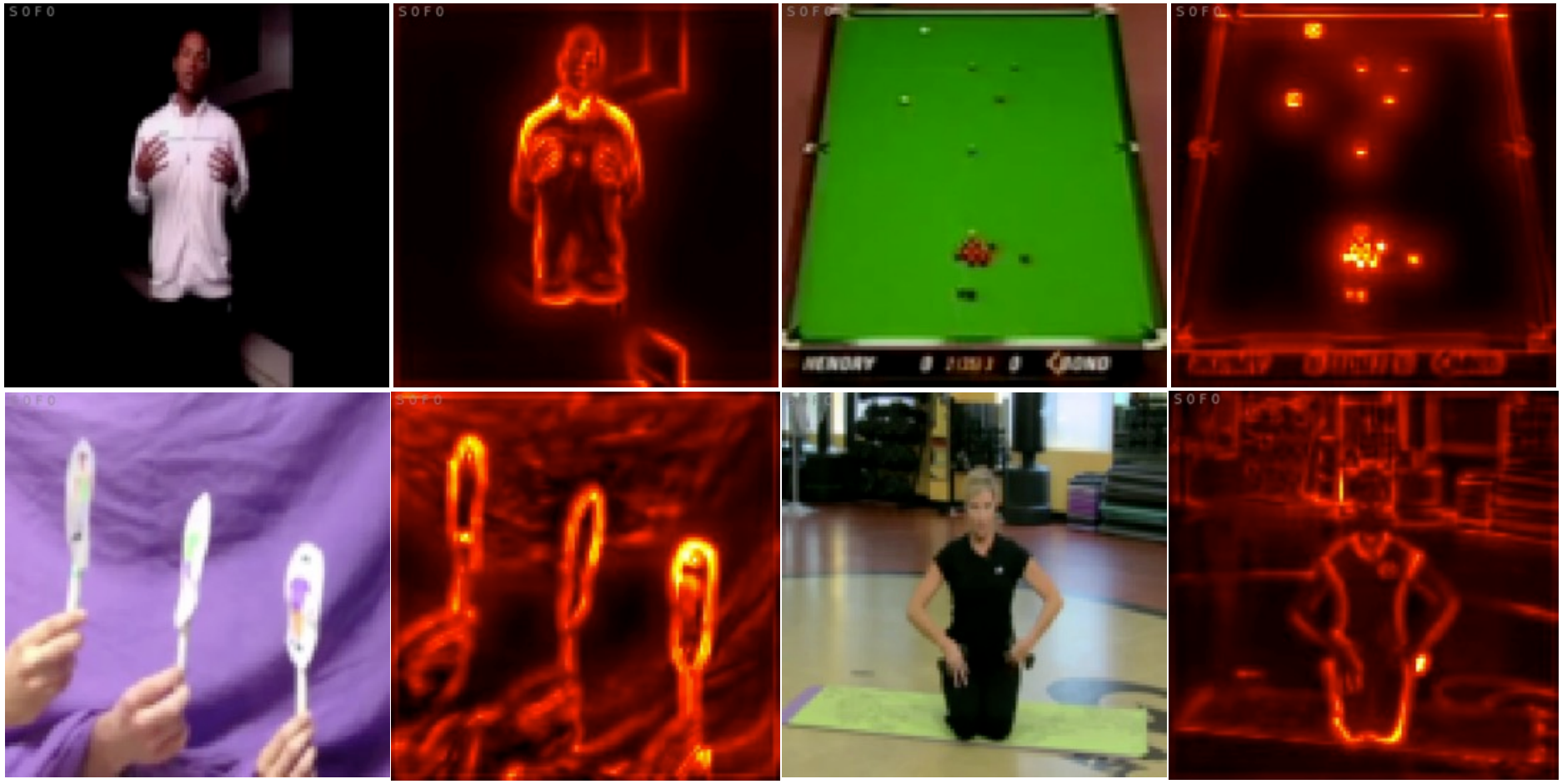
Application: Understand the model

- 3-dimensional CNN (C3D)
- trained on Sports-1M
- explain predictions for 1000 videos from the test set



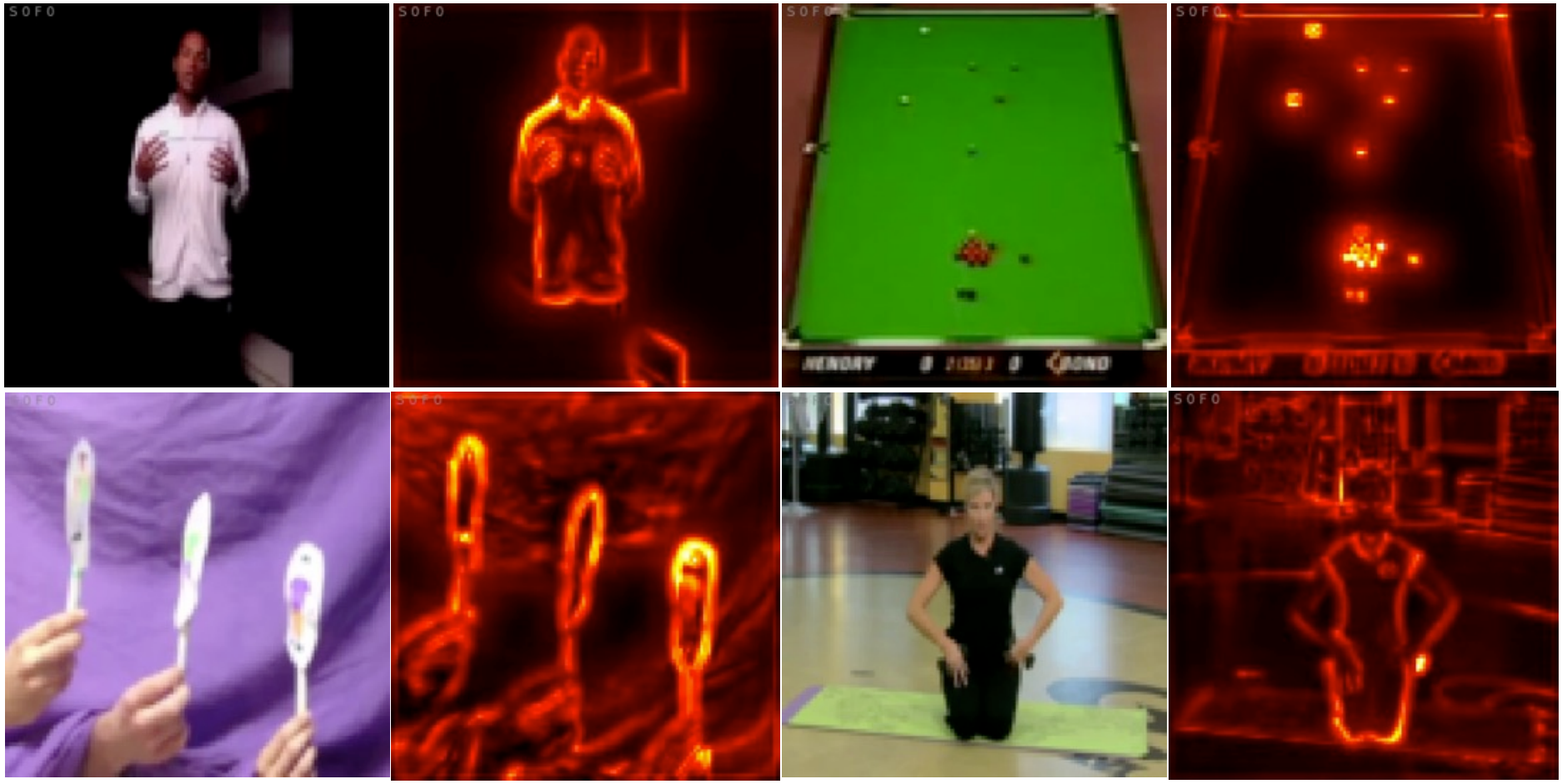
(Anders et al., 2018)

Application: Understand the model



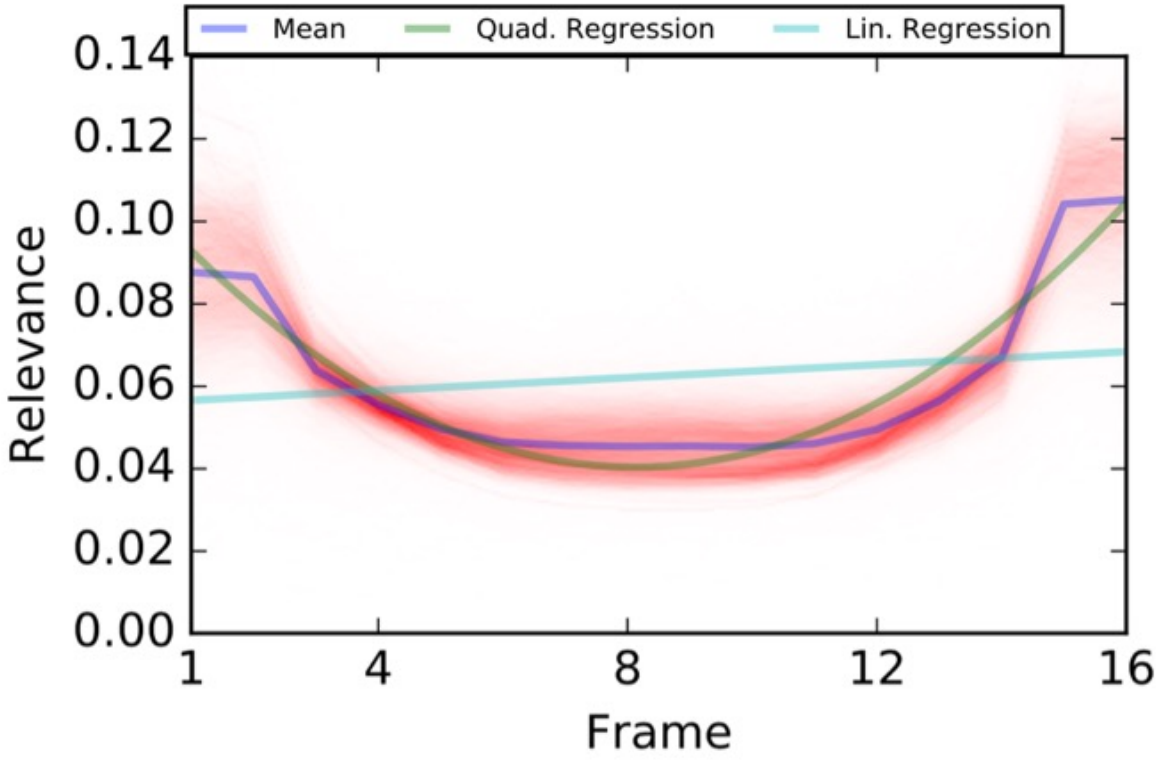
(Anders et al., 2018)

Application: Understand the model



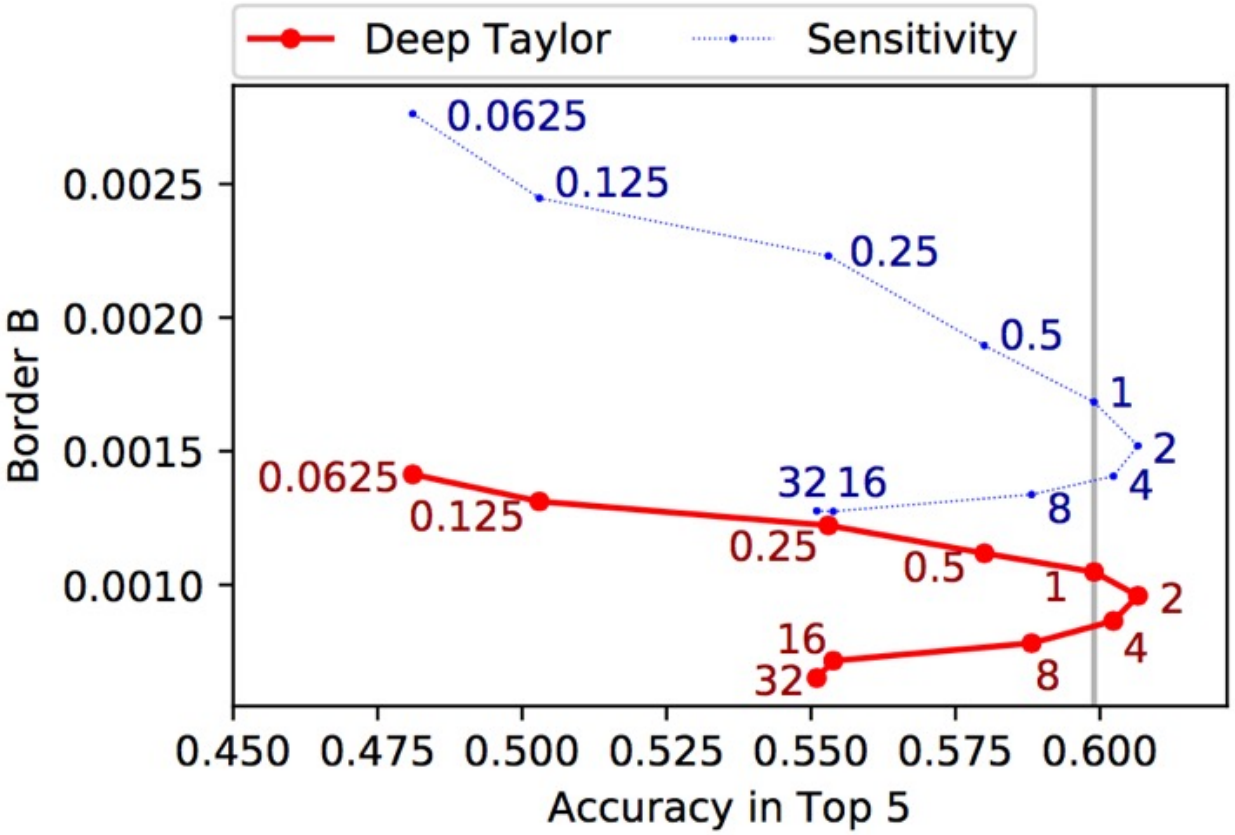
(Anders et al., 2018)

Application: Understand the model



Observation: Explanations focus on the bordering of the video, as if it wants to watch more of it.

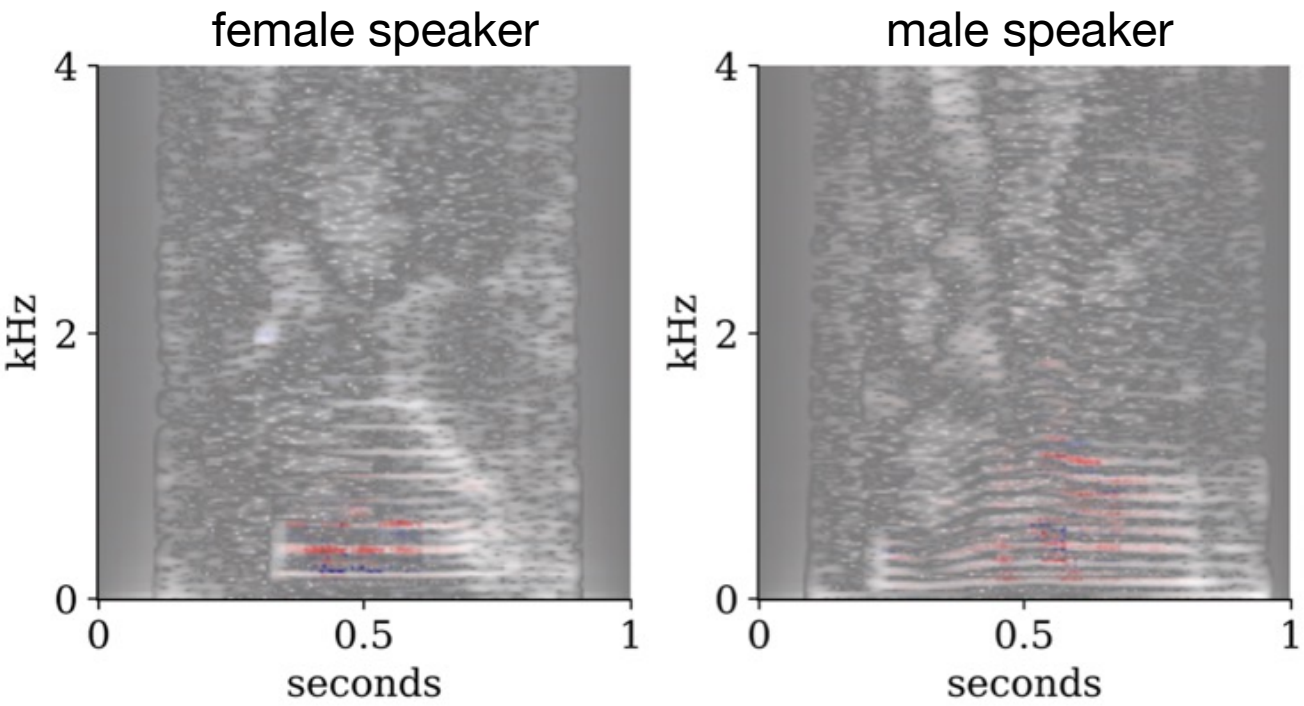
Application: Understand the model



Idea: Play video in fast forward (without retraining) and then the classification accuracy improves.

Application: Understand the model

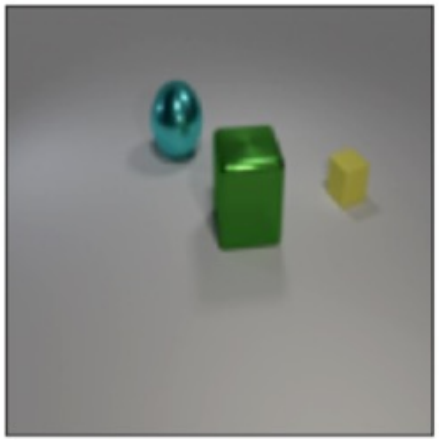
- AlexNet model
- trained on spectrograms
- spoken digits dataset (AudioMNIST)

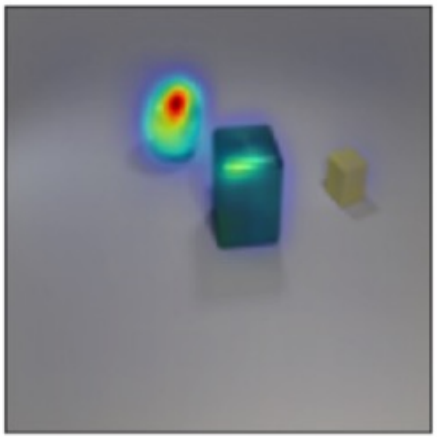


model classifies gender based on the fundamental frequency and its immediate harmonics (see also Traunmüller & Eriksson 1995) *(Becker et al., 2018)*

Application: Understand the model

Question	LRP
there is a metallic cube ; are there any large cyan metallic objects behind it ?	there is a metallic cube ; are there any large cyan metallic objects behind it ?





- reimplement model of (Santoro et al., 2017)
- test accuracy of 91,0%
- CLEVR dataset

Question Type	LRP
equal_size	size 2279, the 1939, cylinder 1201 (0, 1000, 2000, 3000, 4000)
greater_than	number 2177, more 1656, than 1515 (0, 1000, 2000, 3000, 4000)
query_color	color 12819, what 6837, is 5735 (0, 10000, 20000)
count	what 10823, how 9310, many 5691 (0, 10000, 20000, 30000)

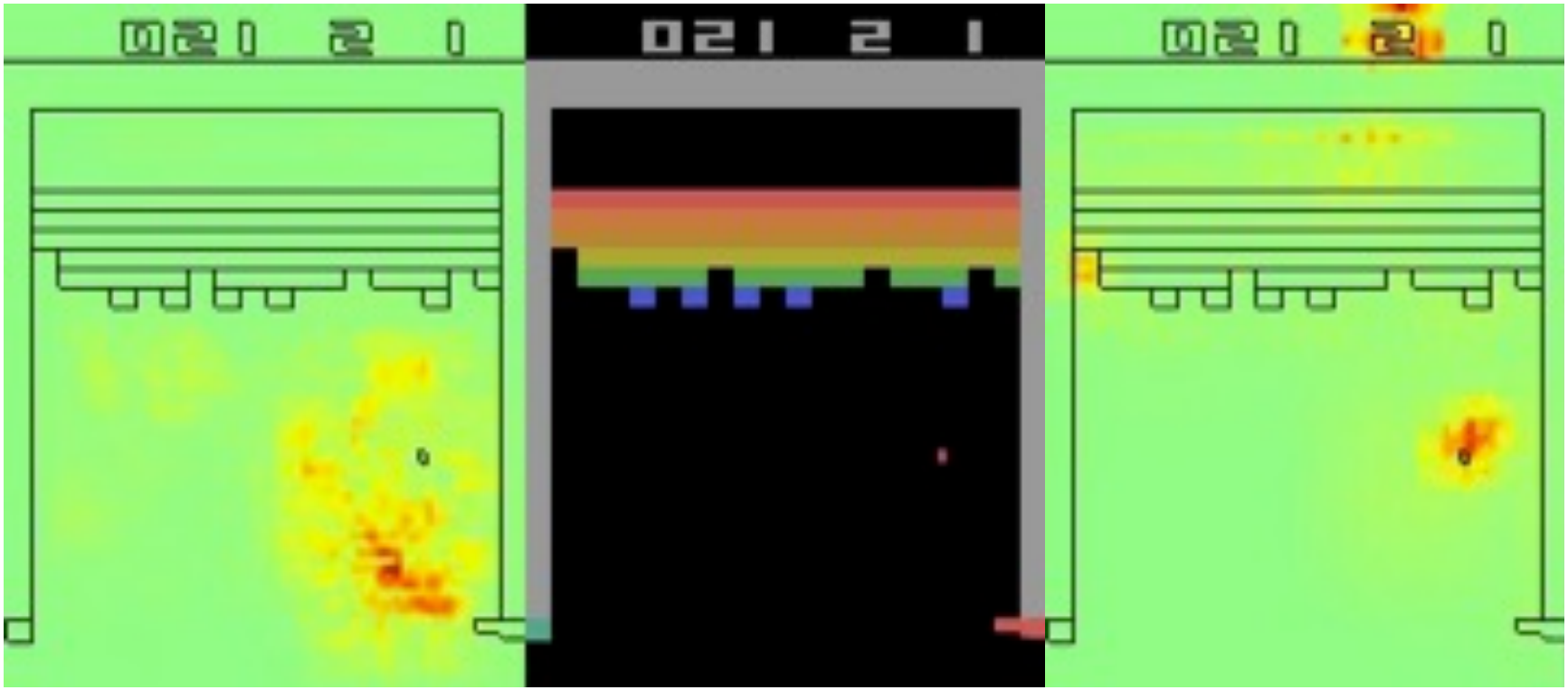
model understands the question and correctly identifies the object of interest

(Arras et al., 2018)

Application: Understand the model

Sensitivity Analysis

LRP



does not focus on where the ball is, but on where the ball could be in the next frame

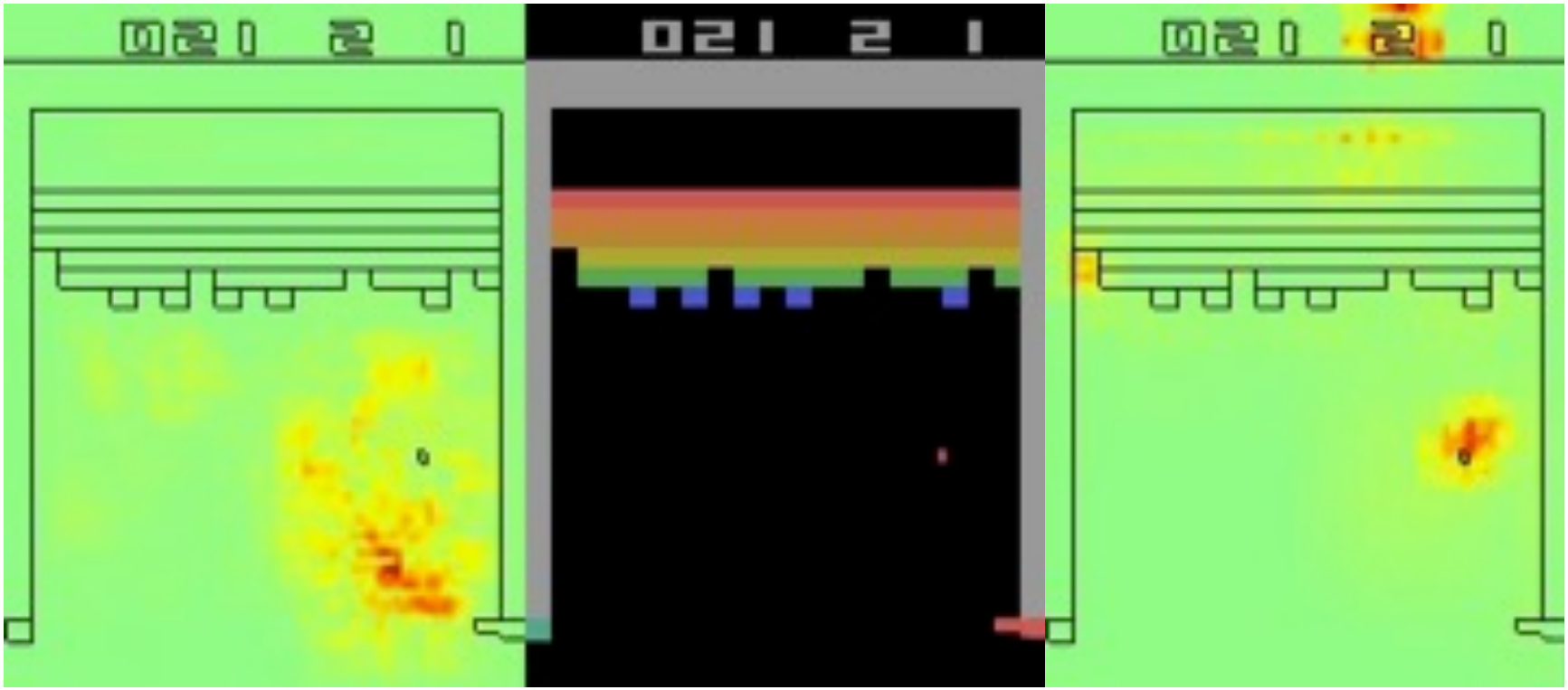
LRP shows that that model tracks the ball

(Lapuschkin et al., in prep.)

Application: Understand the model

Sensitivity Analysis

LRP



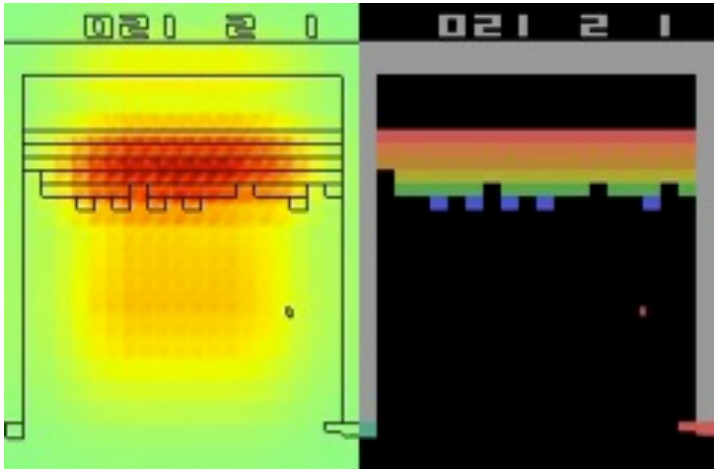
does not focus on where the ball is, but on where the ball could be in the next frame

LRP shows that that model tracks the ball

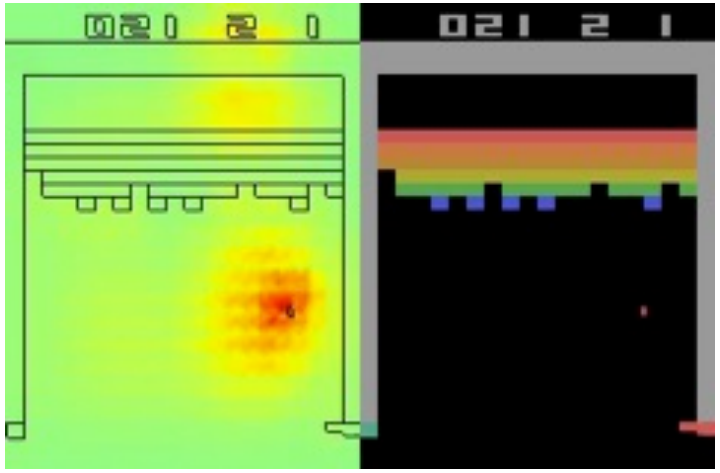
(Lapuschkin et al., in prep.)

Application: Understand the model

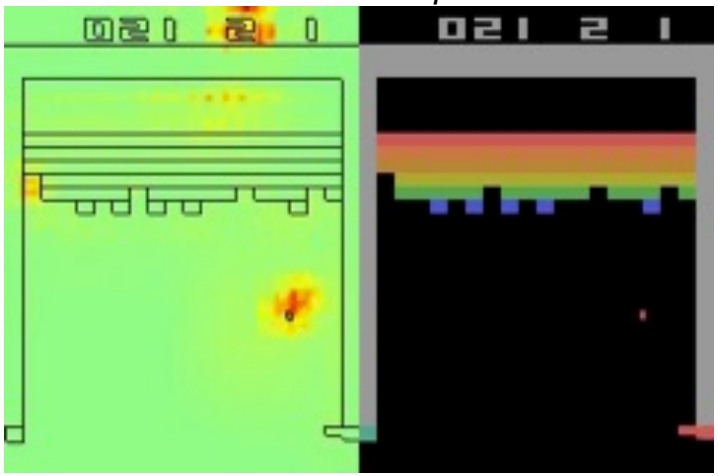
After 0 epochs



After 25 epochs



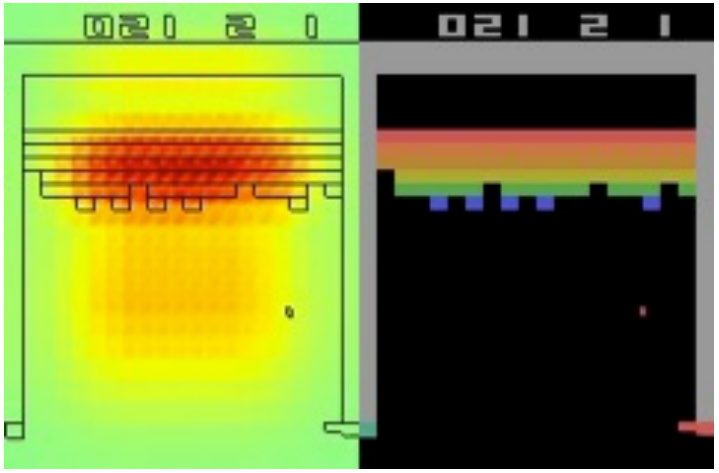
After 195 epochs



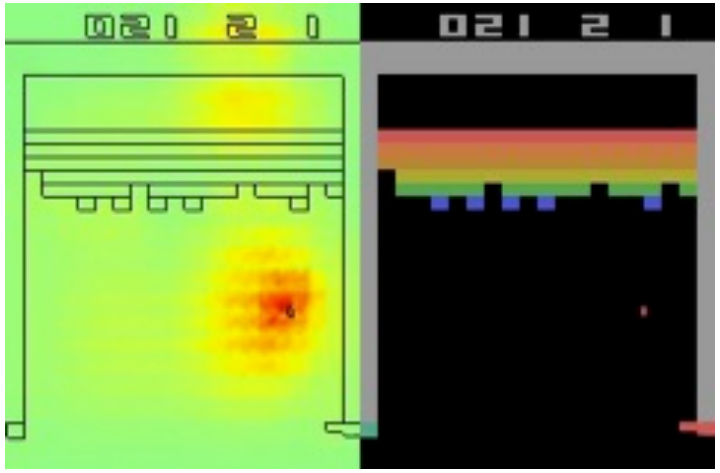
(Lapuschkin et al., in prep.)

Application: Understand the model

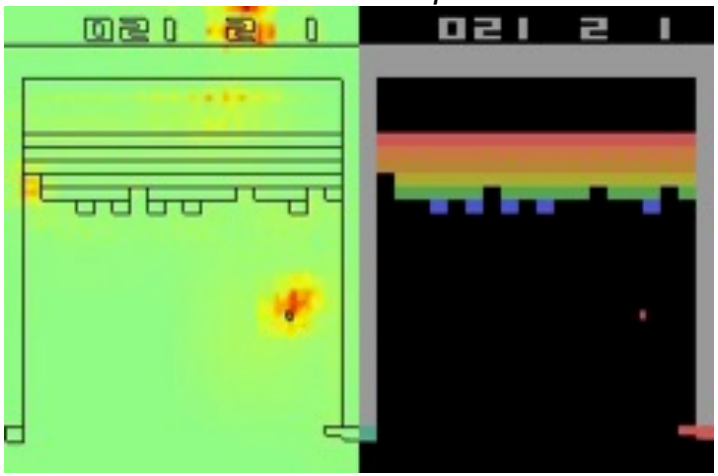
After 0 epochs



After 25 epochs

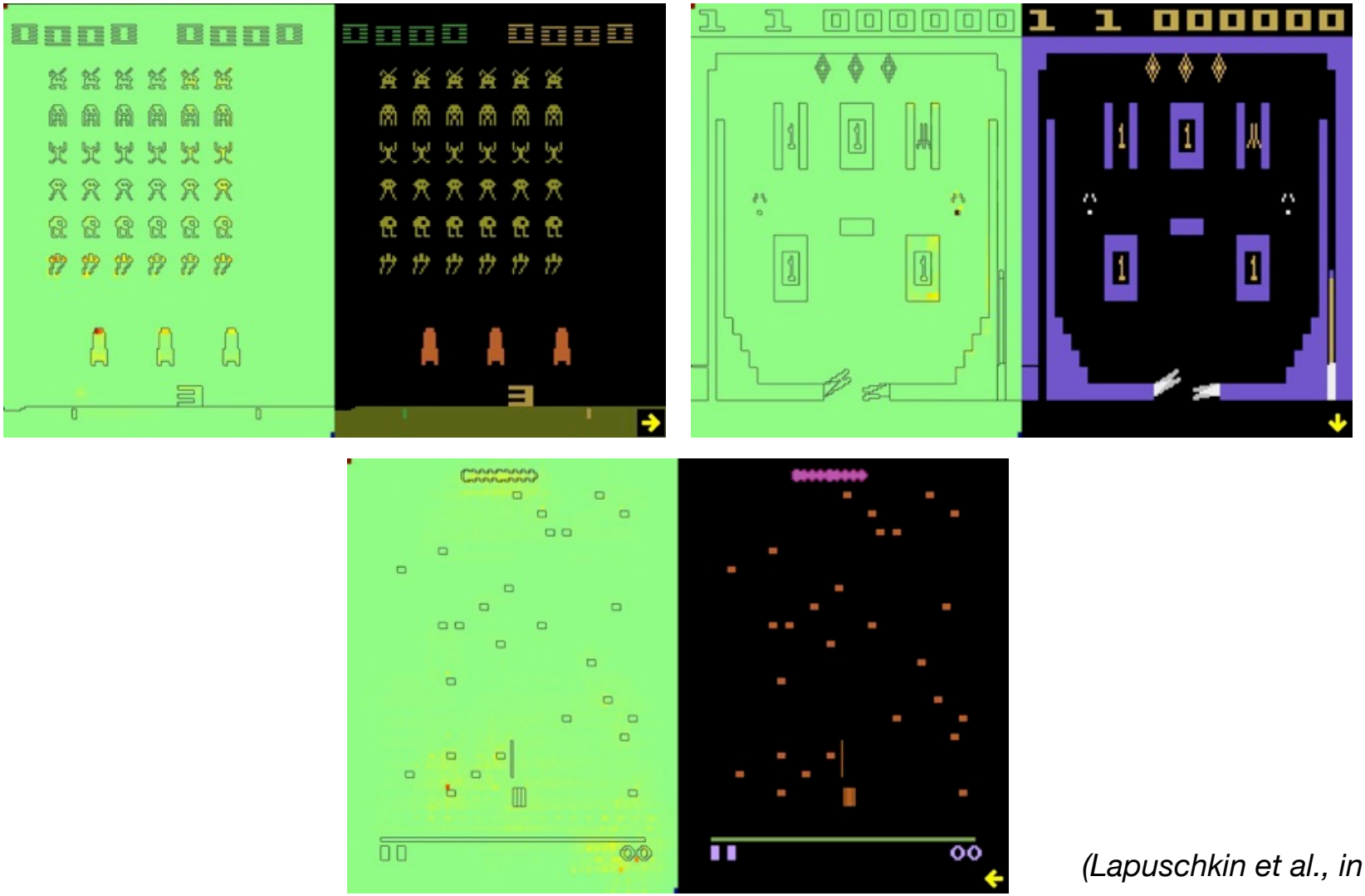


After 195 epochs



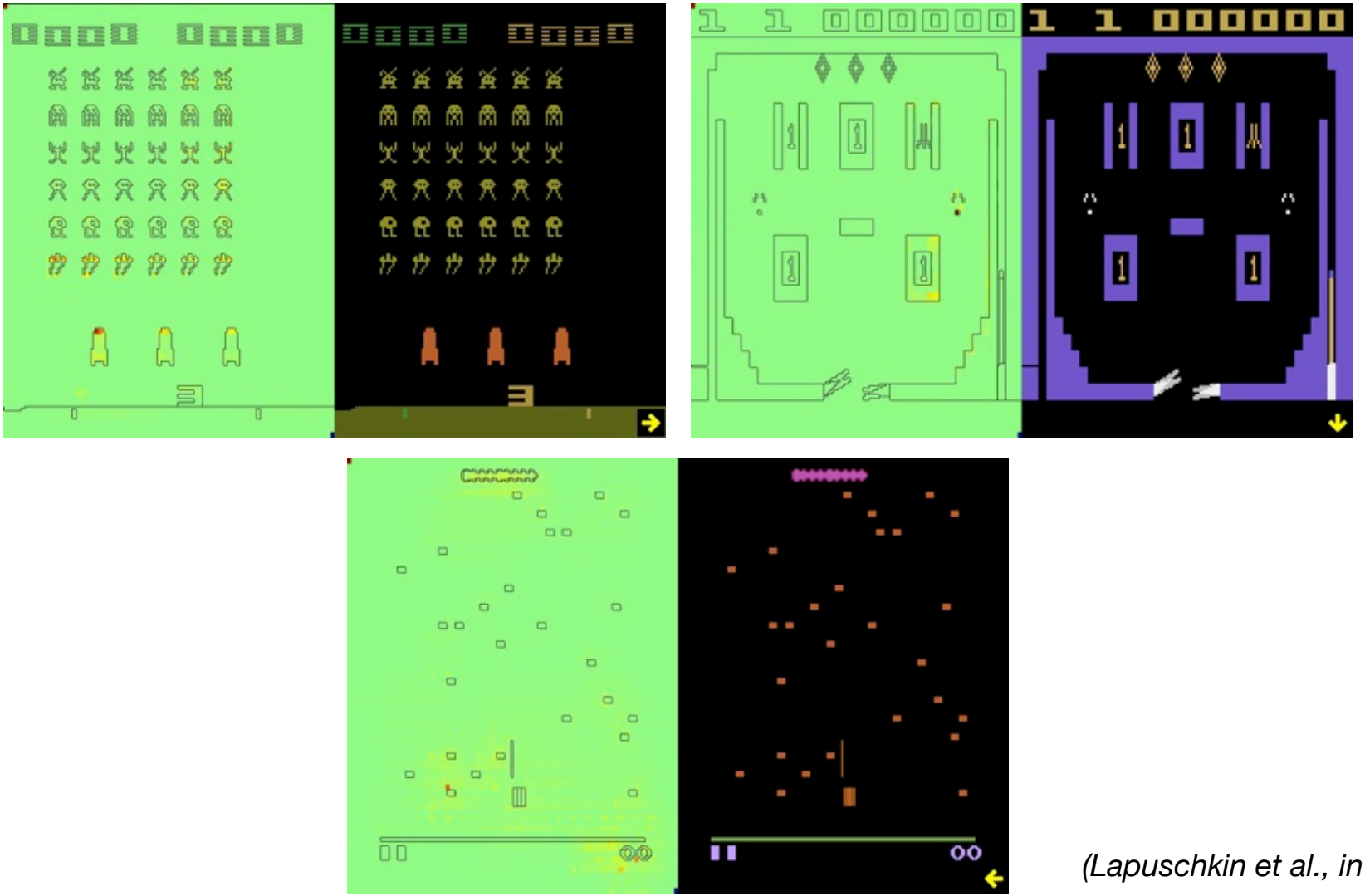
(Lapuschkin et al., in prep.)

Application: Understand the model



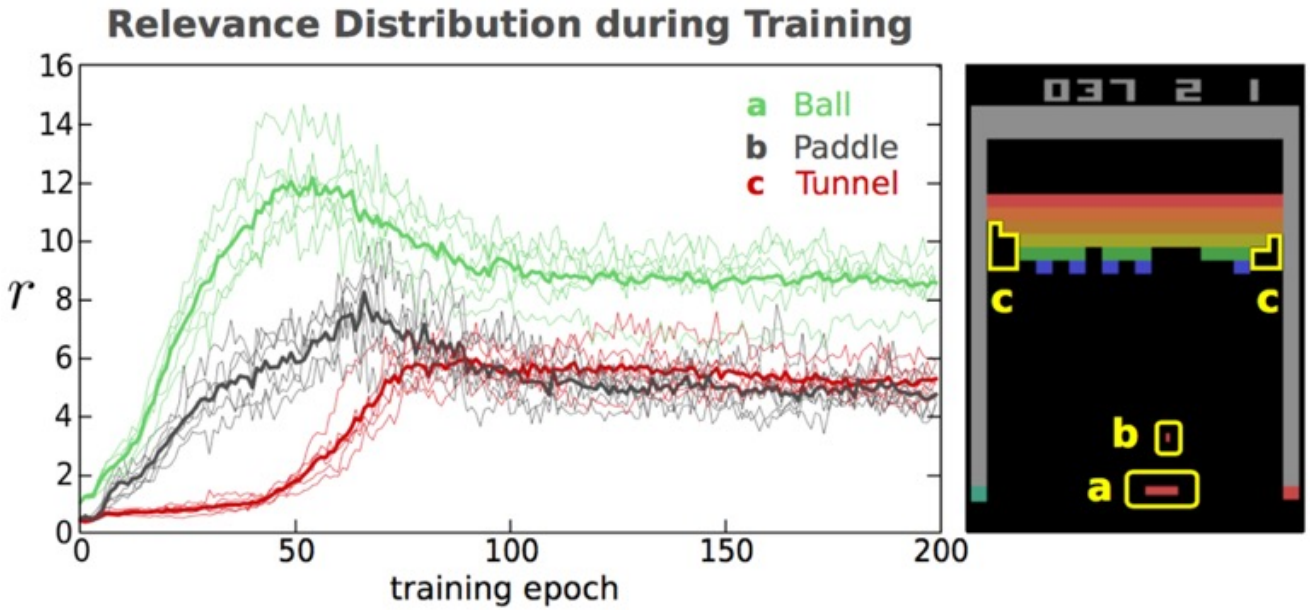
(Lapuschkin et al., in prep.)

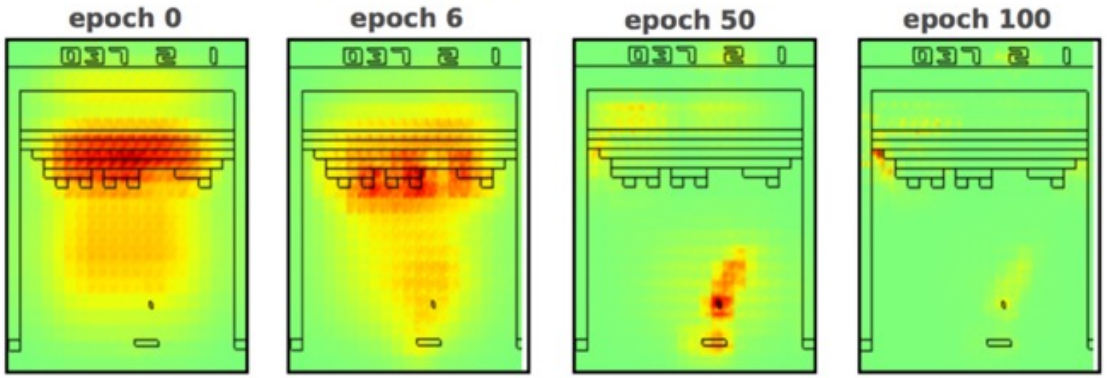
Application: Understand the model



(Lapuschkin et al., in prep.)

Application: Understand the model





model learns
1. track the ball
2. focus on paddle
3. focus on the tunnel

(Lapuschkin et al., in prep.)

Tutorial on Interpretable Machine Learning

Part 4: Case Study: Interpretable ML in Histopathology

Heatmapping - a quick case study in histopathology

Talk for MICCAI workshop on Interpretable ML, 2018.

Alexander Binder
Joint work with F. Klauschen, S. Lapuschkin (Bach), G. Montavon, K.-R. Müller, W. Samek

ISTD Pillar, Singapore University of Technology and Design (SUTD)

September 15, 2018

Deep Neural networks and (near-)human performance

Lipnet beats humans at lip reading

Human performance in Generic classification

IMAGENET

human performance in low-res (!) traffic sign recognition

DeepStack outplays Humans in poker

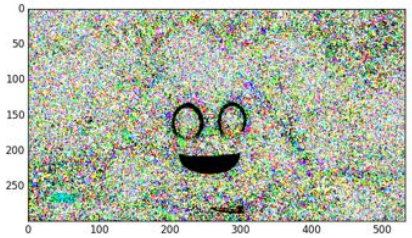
Computer outplays Humans in DOOM

Mimicking art styles: `https://deepart.io`

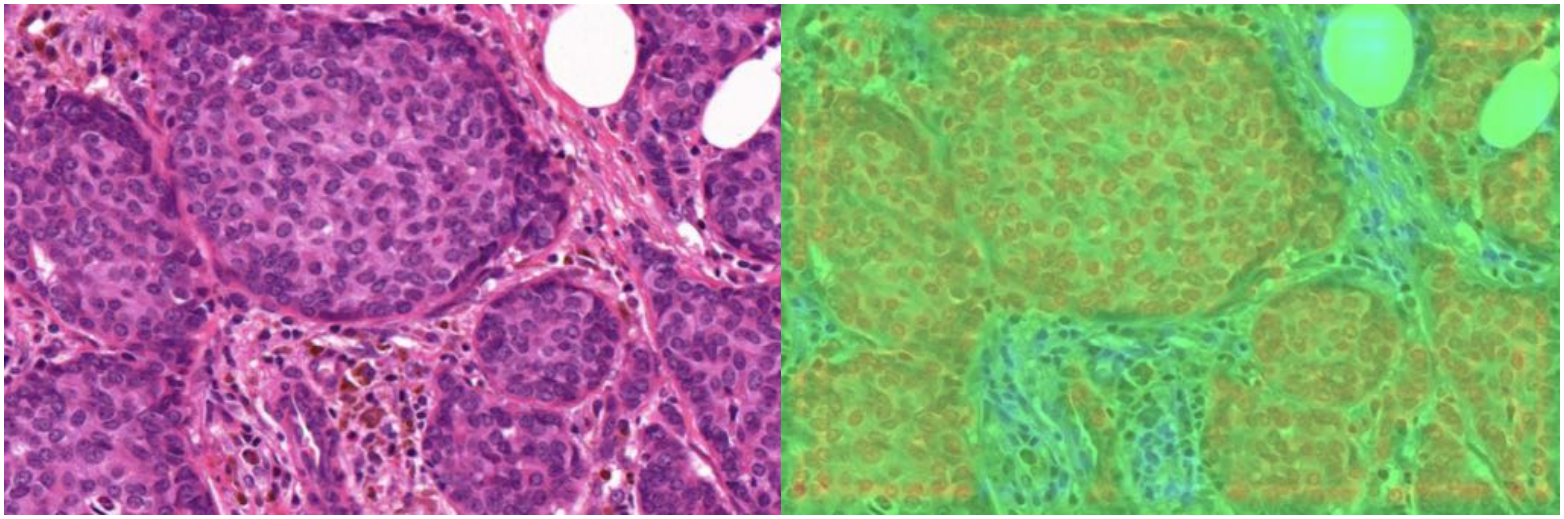
Deep Learning tops human average on a constrained (!) reading comprehension task (SQuAD Dataset)

Human-like performance $\neq$ Human-like reasoning

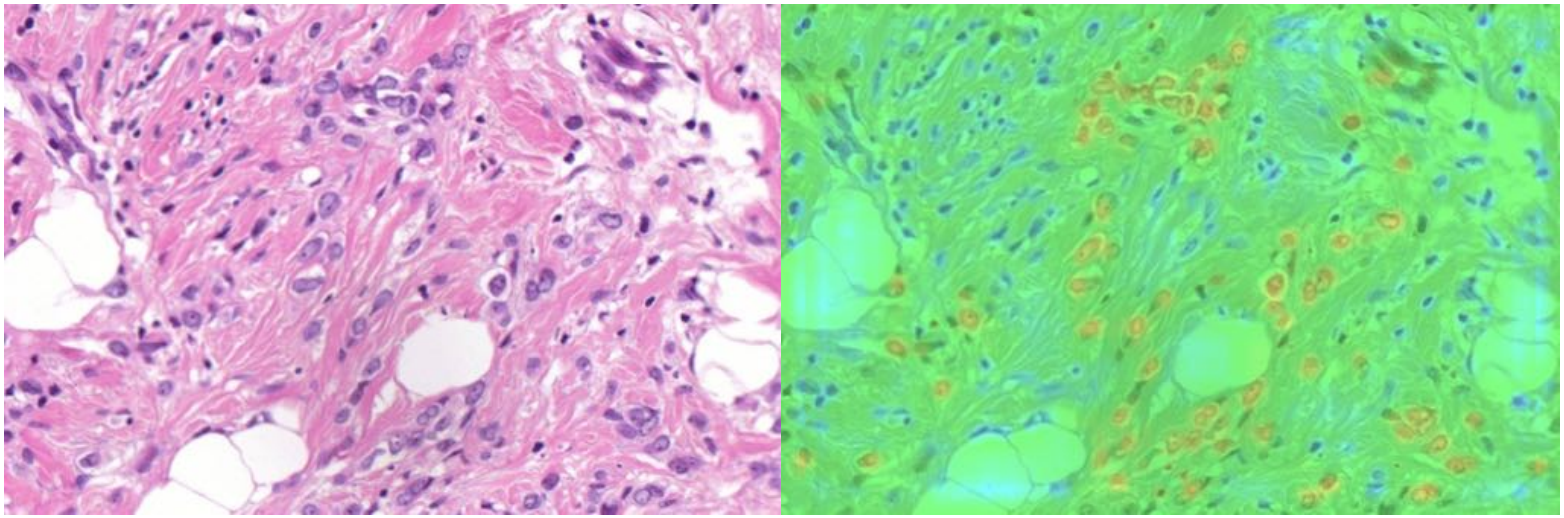
Adversarial attacks against deep neural networks are easy.



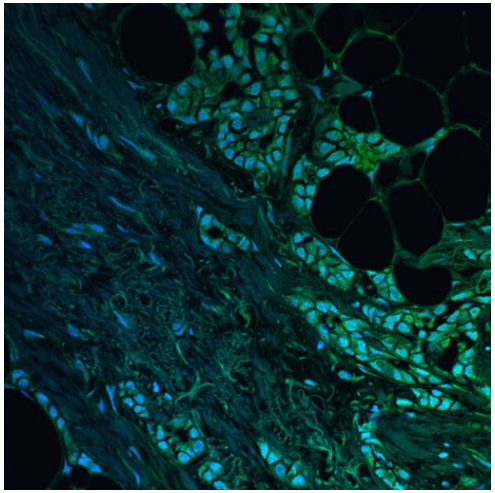
Can explanation be a useful tool beyond mere curiousity?



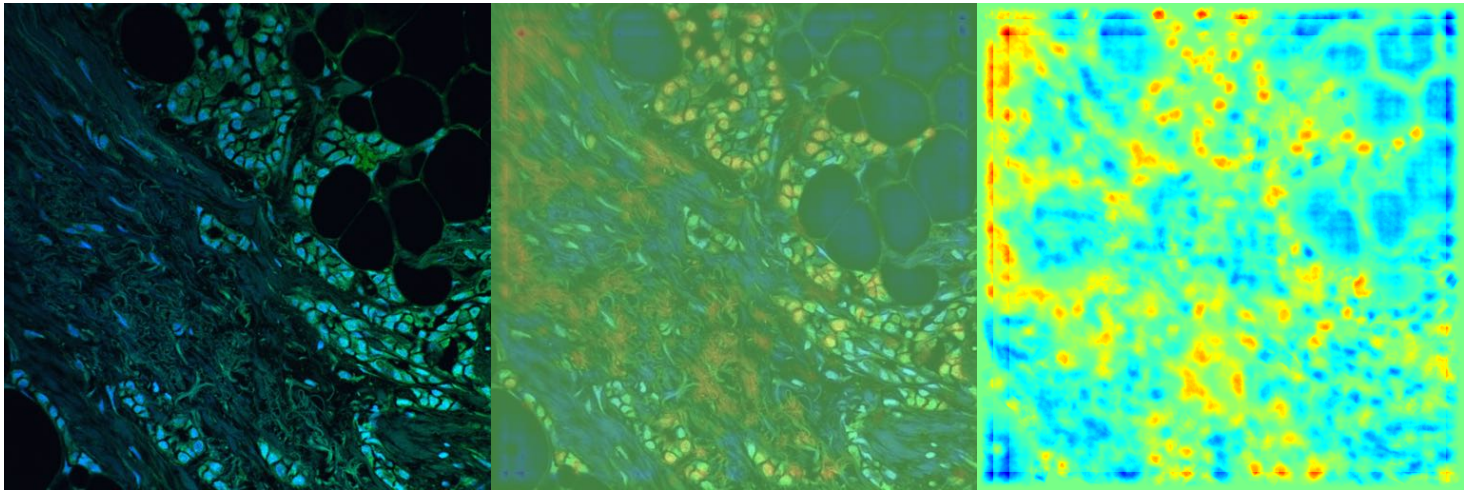
Can explanation be a useful tool beyond mere curiousity?



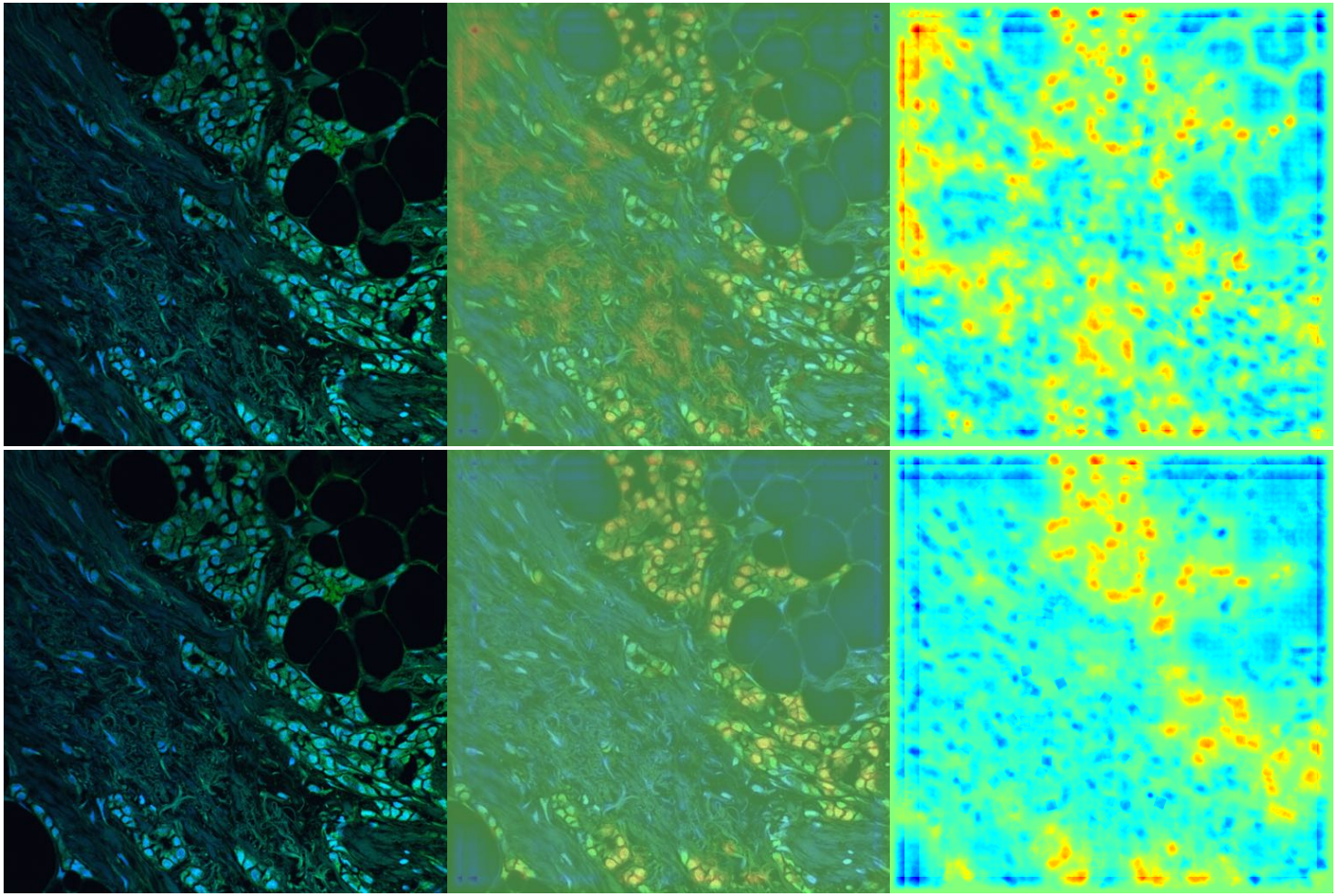
Application Idea: Finding Biases in Your Training Data



Application Idea: Finding Biases in Your Training Data



Application Idea: Finding Biases in Your Training Data



Can explanation be a useful tool beyond mere curiousity?

- BoW: heatmapping for cancer evidence
- BoW: heatmapping for molecular expression evidence
- Deep Learning: heatmapping for looking for biases.

Advertisement Warning

The next slides show authors own research. Views might be positively biased ;) . Nope I have not solved all interpretability problems with it.

BoW: Heatmapping for cancer evidence

BoW: heatmapping for cancer evidence

Why do we still talk about BoW?

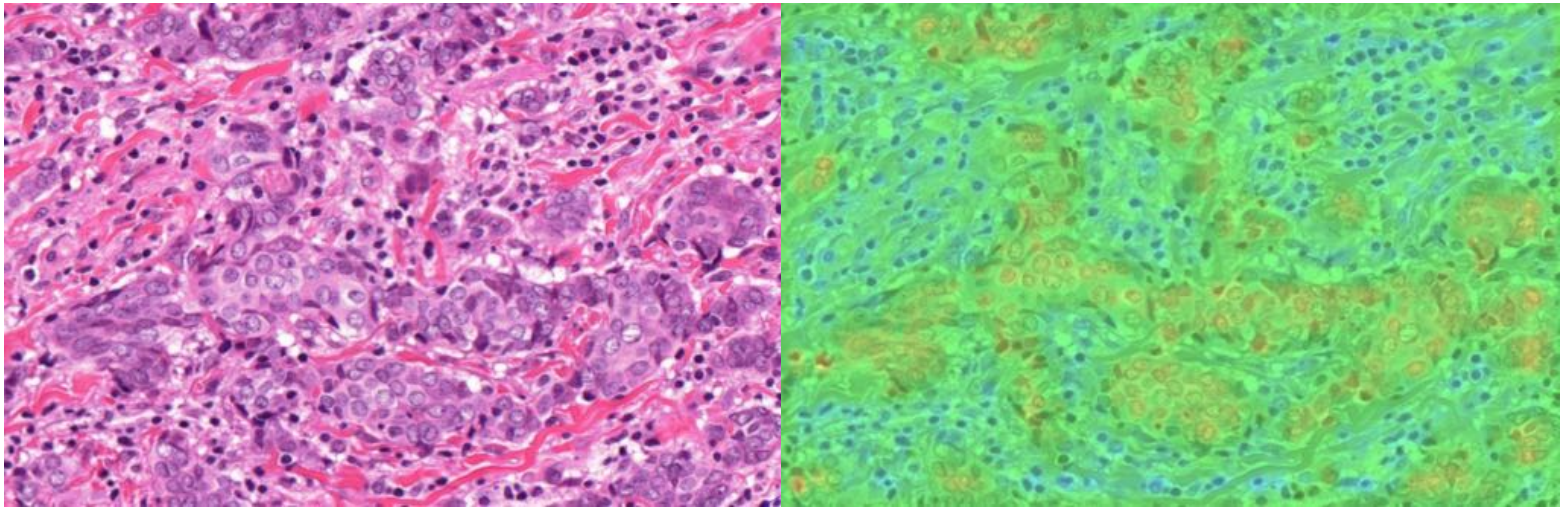
- Good performance for small sample sizes (samples per class $< 10^3$).
- Stable against small changes in data augmentation / choices of negative sampling.
- Heatmapping slower compared to DNN+GPU+innvestigate

BoW: heatmapping for cancer evidence

Useful for ?

BoW: heatmapping for cancer evidence

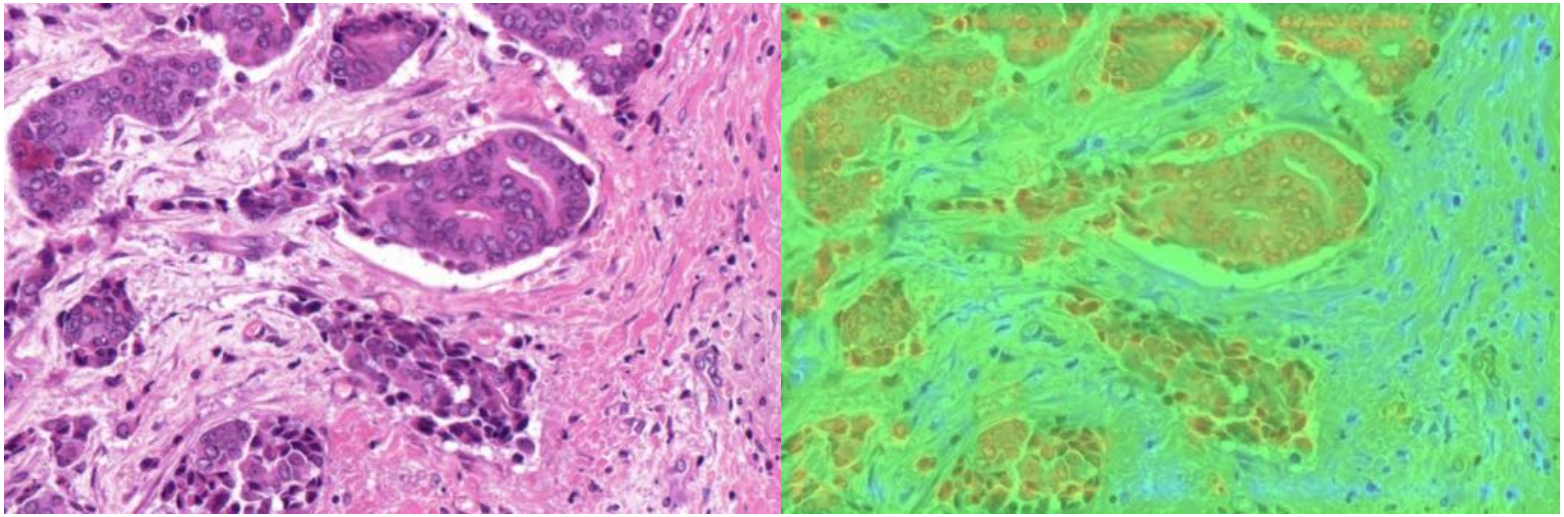
Useful for ?
Shows cases where heatmapping works well



No stain normalization was used here – stability.
Towards computational fluorescence microscopy: Machine learning-based integrated prediction of morphological and molecular tumor profiles, Binder et al., arxiv 2018

BoW: heatmapping for cancer evidence

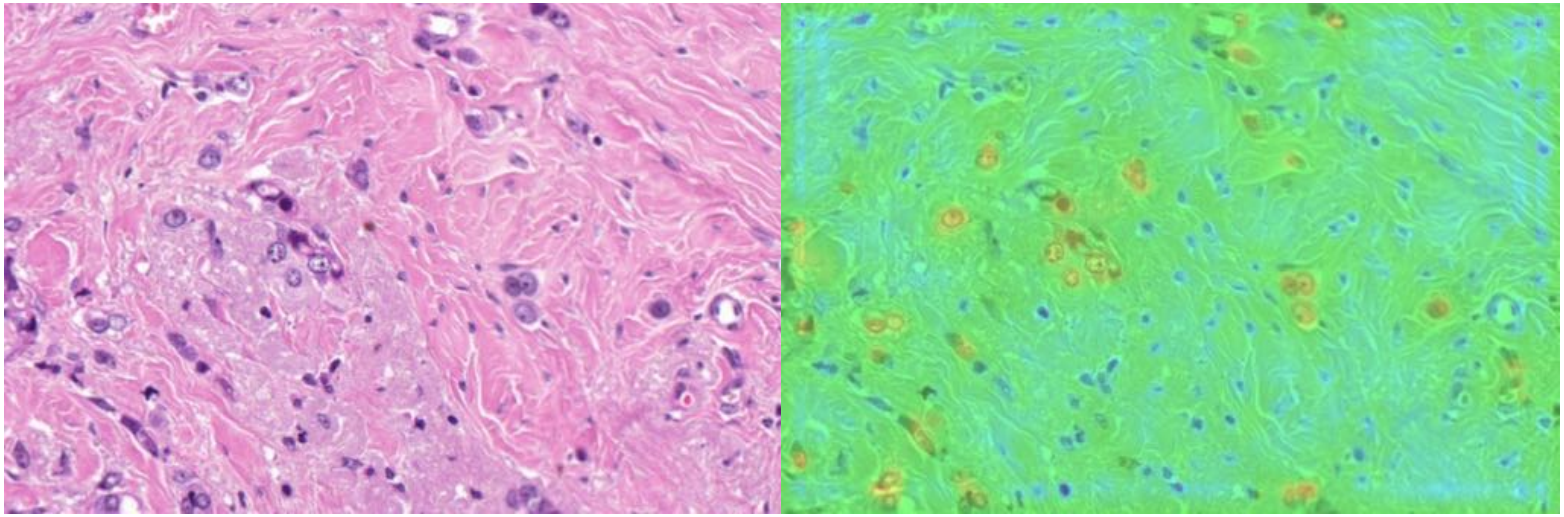
Useful for ?
Shows cases where heatmapping works well



Towards computational fluorescence microscopy: Machine learning-based integrated prediction of morphological and molecular tumor profiles, Binder et al., arxiv 2018

BoW: heatmapping for cancer evidence

Useful for ?
Shows cases where heatmapping works well

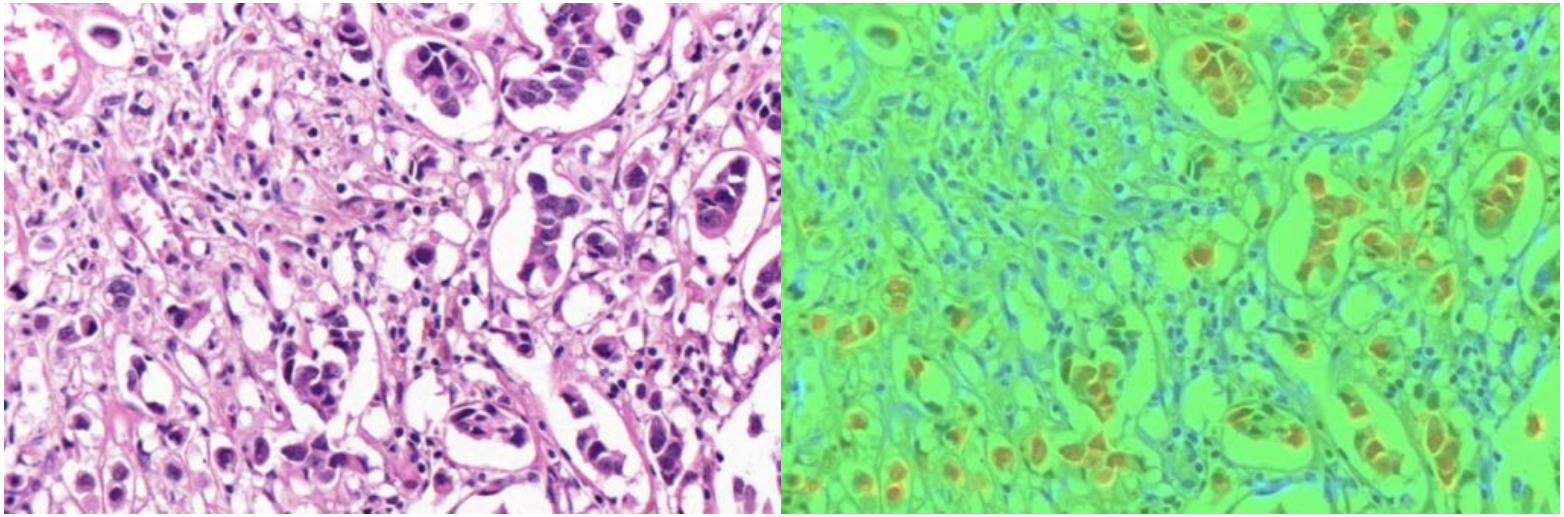


Towards computational fluorescence microscopy: Machine learning-based integrated prediction of morphological and molecular tumor profiles, Binder et al., arxiv 2018

BoW: heatmapping for cancer evidence

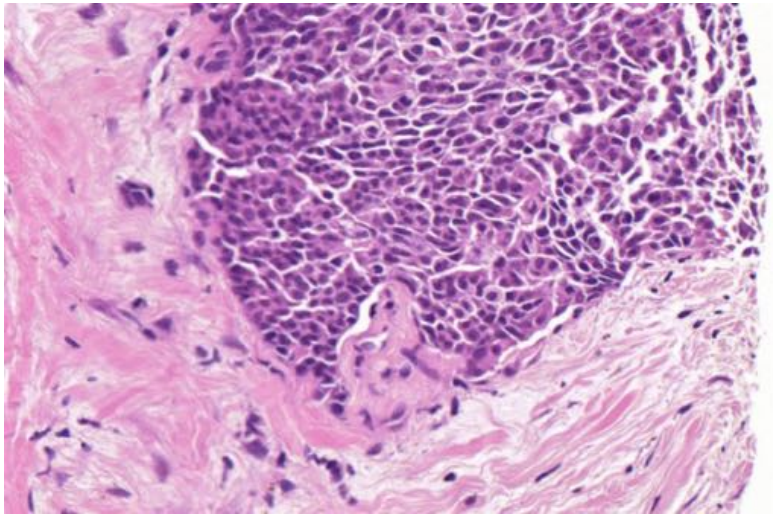
Useful for ?

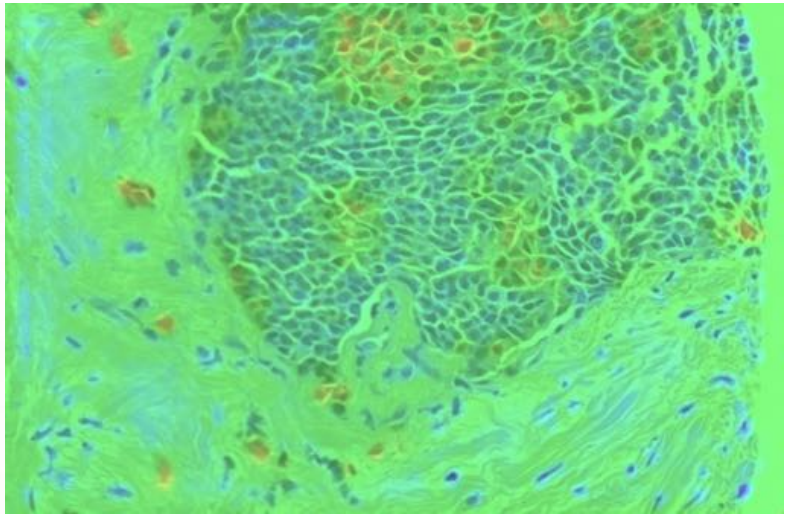
Shows cases where heatmapping works well



BoW: heatmapping for cancer evidence

Useful for ?
Shows cases where heatmapping fails



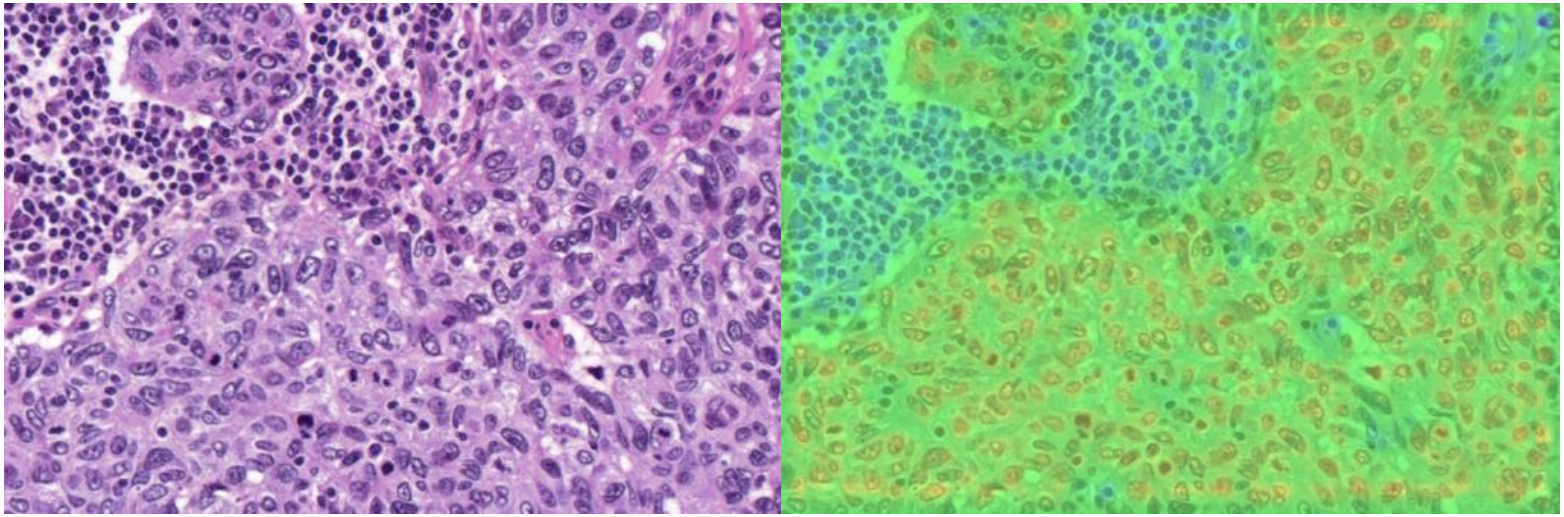


Too similar to dense clusters of TiLs

BoW: heatmapping for cancer evidence

Useful for ?

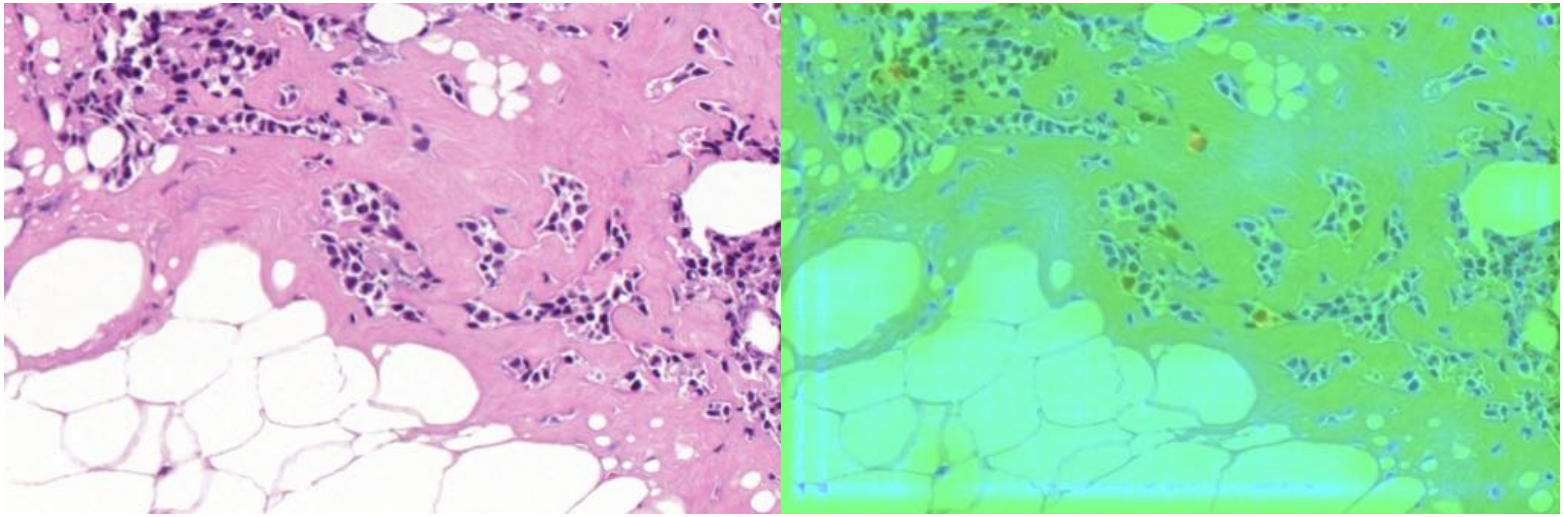
compare to:



Too similar to dense clusters of TiLs

BoW: heatmapping for cancer evidence

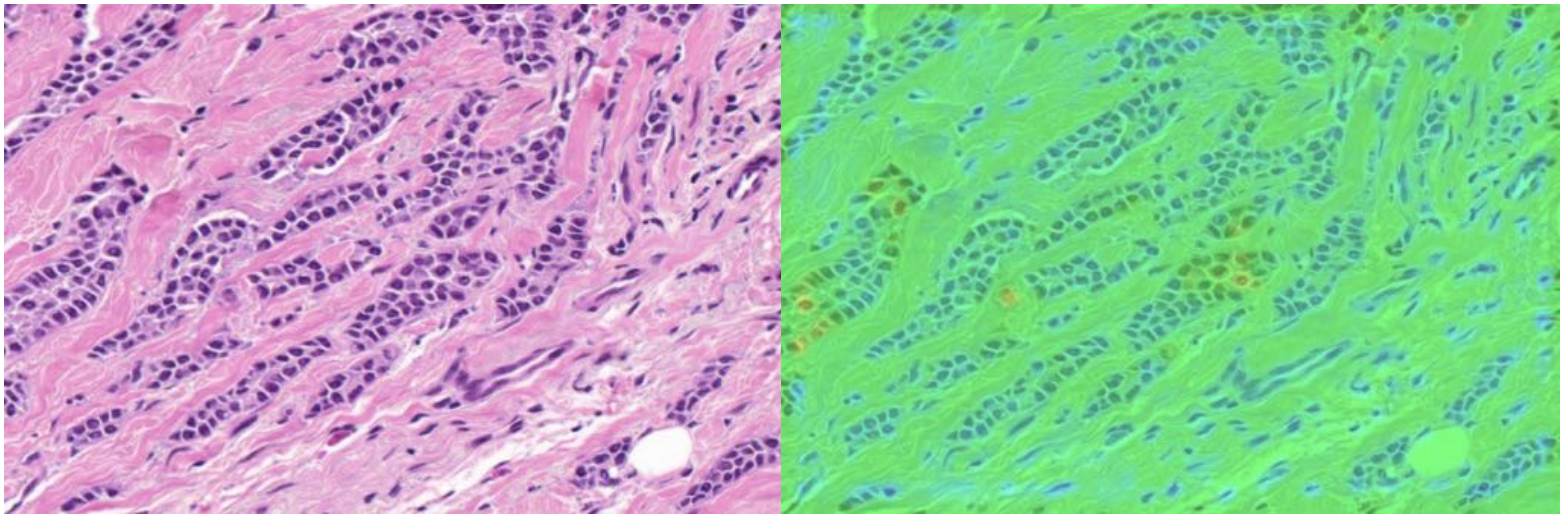
Useful for ?
Shows cases where heatmapping fails



Too similar to lymphocytes, BoW feature does not capture that their distribution is untypical for lymphos

BoW: heatmapping for cancer evidence

Useful for ?
Shows cases where heatmapping fails



Too similar to epithelial cells?? (patch-wise kernel similarity matrix may reveal this)

BoW: heatmapping for cancer evidence

Useful for ?

Finding subtypes that are not recognized well, for example because undersampled in the training+testing set.

potential solutions:

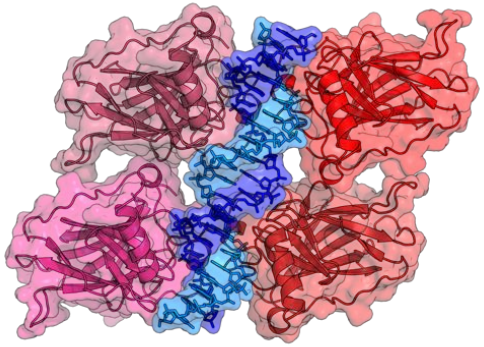
- improved sampling
- feature engineering (BoW)
- data augmentation engineering (deep learning)

BoW: heatmapping for X

Cancer is obvious. How about molecular properties?

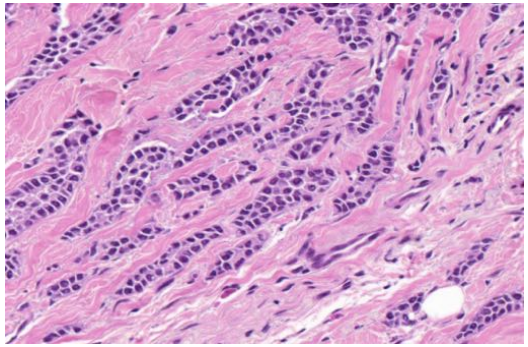
Example: measurement of RNA in a biopsy sample. Can we localize evidence for the expression of the corresponding protein?

Example p53, a tumor suppressor molecule

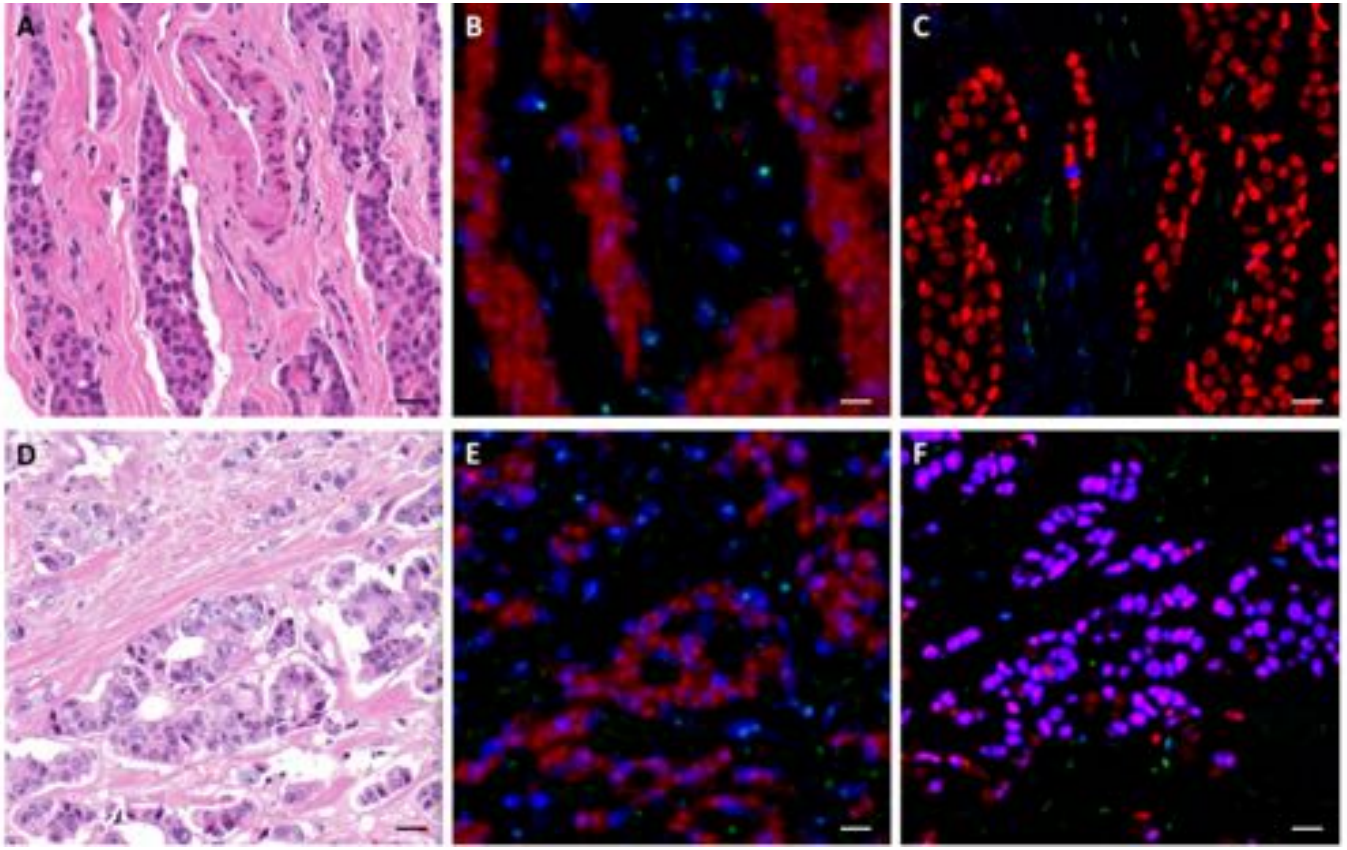


No ad-hoc localization existent.

- forward pass: predict predict concentration from HE stain *as a classification problem*
- backward pass: find evidence localized to pixels



BoW: heatmapping for p53



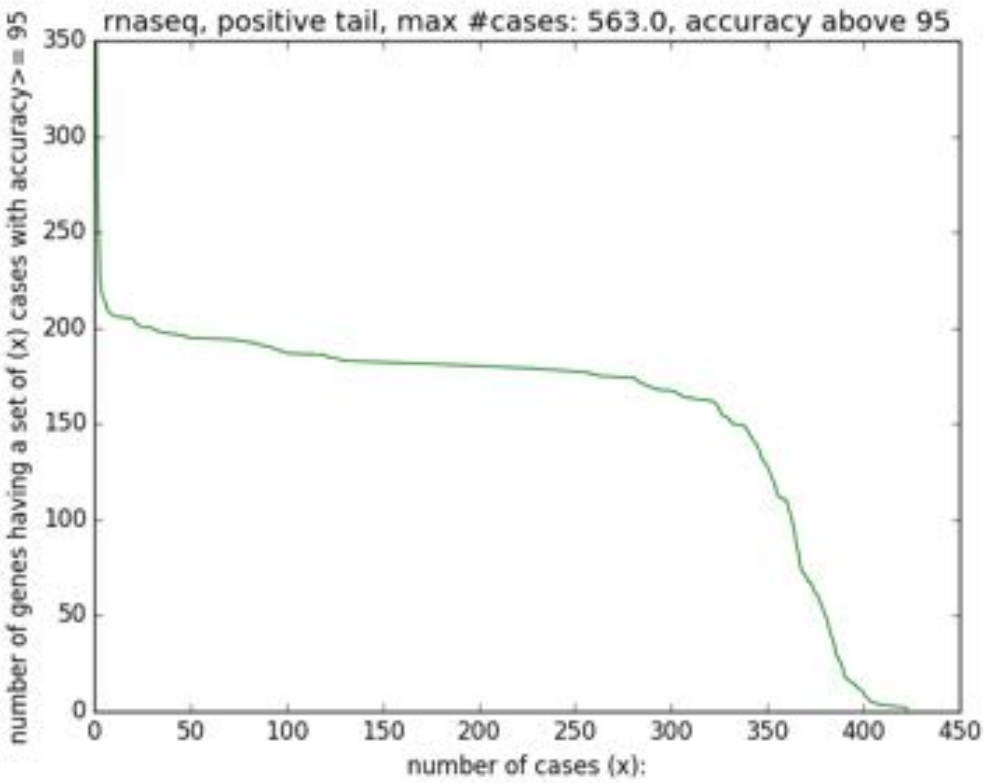
Upper column: low expression case. Lower column: High expression case. Middle row: predicted. Right row: Immunostained groundtruth from a neighboring slice.

Towards computational fluorescence microscopy: Machine learning-based integrated prediction of morphological and molecular tumor profiles, Binder et al., arxiv 2018

BoW: heatmapping for p53

Idea is not limited to heatmapping of evidence for cellular structures.

Accuracy is high on large subsets of patients:



BoW: heatmapping for X - How ?

Forward pass is kernel machine over BoW feature.
Backward pass to obtain scores per pixel:

- Backpropagate from output of SVM $f(x)$ to kernel input dimension $x_{(d)}$ of x
- Backpropagate from kernel input dimensions $x_{(d)}$ to local features l aggregated into the BoW feature
- Backpropagate from local feature l to pixel q

BoW: heatmapping for X - How ?

Backpropagate from output to kernel input dimension. Kernel is given as:

$$f(x) = b + \sum_i a_i y_i k(z_i, x) \tag{1}$$

$$\text{goal: } f(x) \approx \sum_d R_d^{(3)}(x), \text{where} \tag{2}$$

$R_d^{(3)}(x)$ is the contribution of dimension d of the test feature $x = (x_{(1)}, \ldots, x_{(D)})$ to $f(x)$.

BoW: heatmapping for X - How ?

Backpropagate from output to kernel input dimension. Kernel is given as:

$$f(x) = b + \sum_i a_i y_i k(z_i, x) \tag{3}$$

$$\text{goal: } f(x) \approx \sum_d R_d^{(3)}(x) \tag{4}$$

In case of dimension-wise separable kernels, such as the HIK-kernel,

$$k(z, x) = \sum_d \min(z_{(d)}, x_{(d)}) \tag{5}$$

$$k(z, x) = \sum_d k_d(z_{(d)}, x_{(d)}) \tag{6}$$

$$f(x) = b + \sum_i a_i y_i k(z_i, x) \tag{7}$$

$$= b + \sum_d \sum_i a_i y_i k_d(z_{(d)}, x_{(d)}) \tag{8}$$

$$R_d^{(3)}(x) = \frac{b}{D} + \sum_i a_i y_i k_d(z_{(d)}, x_{(d)}) \tag{9}$$

BoW: heatmapping for X - How ?

Backpropagate from output to kernel input dimension. Kernel is given as:

$$f(x) = b + \sum_i a_i y_i k(z_i, x) \tag{10}$$

$$\text{goal: } f(x) \approx \sum_d R_d^{(3)}(x) \tag{11}$$

In case of differentiable kernels, such as the χ^2-kernel,

$$k(z, x) = \exp(-\gamma \sum_{d: z_{(d)} + x_{(d)} > 0} \frac{(z_{(d)} - x_{(d)})^2}{z_{(d)} + x_{(d)}}) \tag{12}$$

Taylor decomposition around a root $f(x_0) = 0$ is a way:

$$f(x) \approx 0 + \sum_d (x_{(d)} - x_{0,(d)}) \sum_i a_i y_i \frac{\partial k(z_i, x_0)}{\partial x_{0,(d)}} \tag{13}$$

BoW: heatmapping for X - How ?

Backpropagate from output to kernel input dimension. Kernel is given as:

$$f(x) = b + \sum_i a_i y_i k(z_i, x) \tag{14}$$

$$\text{goal: } f(x) \approx \sum_d R_d^{(3)}(x) \tag{15}$$

Taylor decomposition around a root $f(x_0) = 0$ is a way:

$$f(x) \approx 0 + \sum_d (x_{(d)} - x_{0,(d)}) \sum_i a_i y_i \frac{\partial k(z_i, x_0)}{\partial x_{0,(d)}} \tag{16}$$

$$R_d^{(3)}(x) = (x_{(d)} - x_{0,(d)}) \sum_i a_i y_i \frac{\partial k(z_i, x_0)}{\partial x_{0,(d)}} \tag{17}$$

BoW: heatmapping for X - How ?

Backpropagate from kernel input dimension to local feature.
The Bow feature is a normalized sum of mappings $m_d(l)$ of local features l onto visual word dimensions:

$$x_d = c \sum_l m_d(l) \tag{18}$$

One example mapping is the hard assignment onto the nearest visual word among the set of all visual words $\{w_{d'}\}$:

$$m_d(l) = 1[d == \text{argmin}_{d'} \|l - w_{d'}\|] \tag{19}$$

Note: not differentiable in l, so cannot use Taylor approximation again.


YOU GOT MATH PROBLEMS?
LET ME PUT ON MY THINKING CAT

BoW: heatmapping for X - How ?

Backpropagate from kernel input dimension to local feature.
The Bow feature is a normalized sum of mappings $m_d(l)$ of local features l onto visual word dimensions:

$$x_d = c \sum_l m_d(l) \tag{20}$$

Dont care how $m_d(l)$ looks like, apply special case of LRP-ϵ-rule. that would look like:

$$R^{(2)}(l) = \sum_d R_d^{(3)} \frac{m_d(l)}{\sum_{l'} m_d(l')} \tag{21}$$

Have to take care for those dimensions d without any weights: $\{d \mid \sum_l m_d(l) = 0\}$

BoW: heatmapping for X - How ?

Backpropagate from kernel input dimension to local feature.
The Bow feature is a normalized sum of mappings $m_d(l)$ of local features l onto visual word dimensions:

$$x_d = c \sum_l m_d(l) \tag{22}$$

Apply special case of LRP-ϵ-rule.
Have to take care for those dimensions d without any weights: $\{d \mid \sum_l m_d(l) = 0\}$:

$$Z(x) = \{d \mid \sum_l m_d(l) = 0\} \tag{23}$$

$$R^{(2)}(l) = \sum_{d \notin Z(x)} R_d^{(3)} \frac{m_d(l)}{\sum_{l'} m_d(l')} + \sum_{d \in Z(x)} R_d^{(3)} \frac{1}{\sum_{l'} 1} \tag{24}$$

BoW: heatmapping for X - How ?

Backpropagate from local feature to pixel

Simple idea: distribute relevance $R^{(2)}(l)$ of a local feature l equally over the support pixels q, used to compute the same local feature.

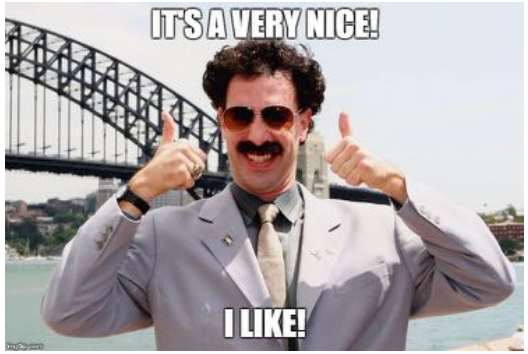
$$LF(q) = \{l \mid q \in supp(l)\} : \text{ local features which touch } q \quad (25)$$

$$R^{(1)}(q) = \sum_{l \in LF(q)} \frac{R^{(2)}(l)}{|supp(l)|} \quad (26)$$

Deep Learning: Heatmapping for looking for biases.

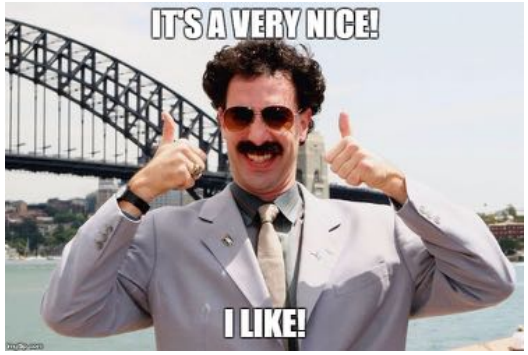
Changes in the work flow due to deep learning

- Feature engineering is dead. Learn your features from data.
- No need for tuning of features by hand.



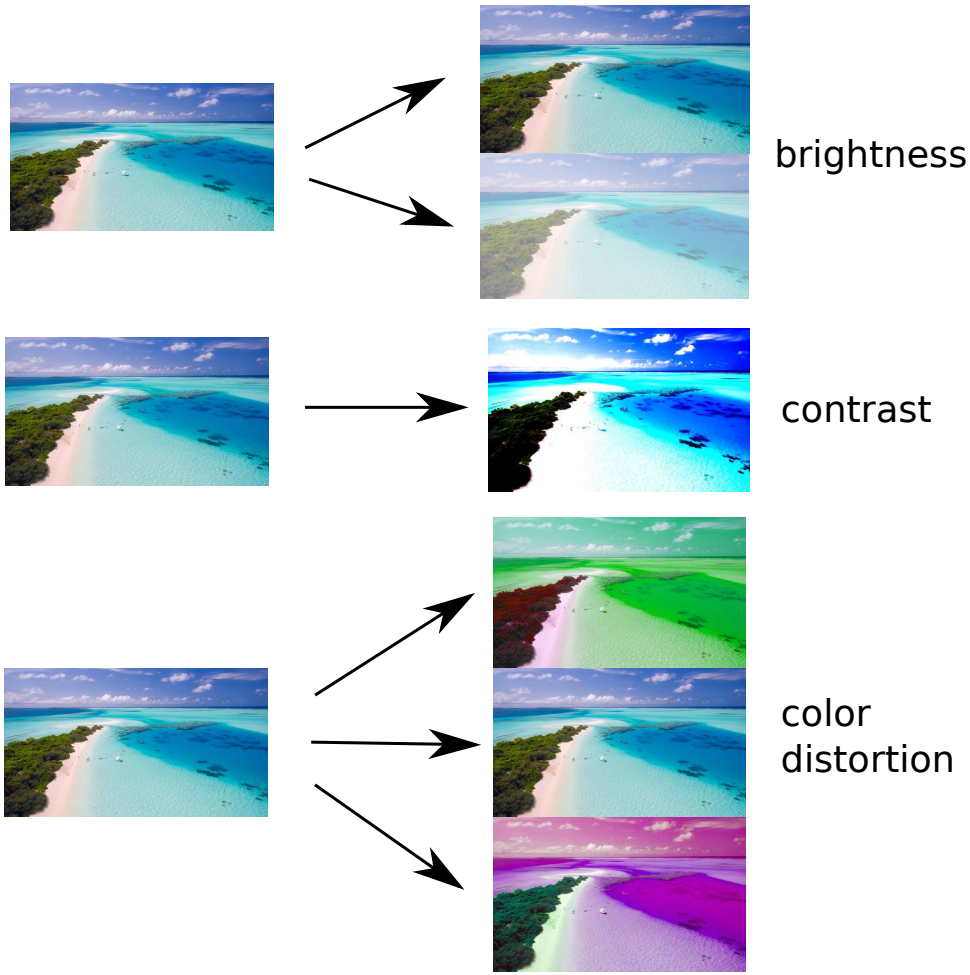
Changes in the work flow due to deep learning

- Feature engineering is dead. Learn your features from data.
- No need for tuning of features by hand.



- Long live data augmentation engineering
- Long live hyperparameter search over grids of 37 different parameters.

Data Augmentation Engineering



This are 6 parameters here already for hyperparameter search here

Data Augmentation Engineering

Many hyperparameters +

- batch size
- minibatch structure
- initial learning rate
- learning rate decay
- optimizer (SGD, momentum, ADAM)

Hyperparameter search in 20-dim space

Data Augmentation Engineering

How to sample elements in minibatches ?

Histopathology in research phase: the inevitable problem of biases

Given a prediction target - example: find evidence for cancer cells.

- If a subclass is undersampled, poor performance on it cannot be detected, because it is not represented in the test set.
- extreme high variability of prediction target and of background. What are relevant subclasses?
- A relevant subclass from positive or negative labeled structures possibly undersampled, and we dont know it!

Histopathology in research phase: the inevitable problem of biases

- If a subclass is undersampled, poor performance on it cannot be detected, because it is not represented in the test set.

Leads to a design problem:

- how to sample positive regions for annotation? (how much of certain structures need to be sampled?)
- how to sample negative regions for annotation?

Hypothesis: Heatmapping over large test slides may reveal undersampled structures in a qualitative manner and help in the iterative solution of the design problem.

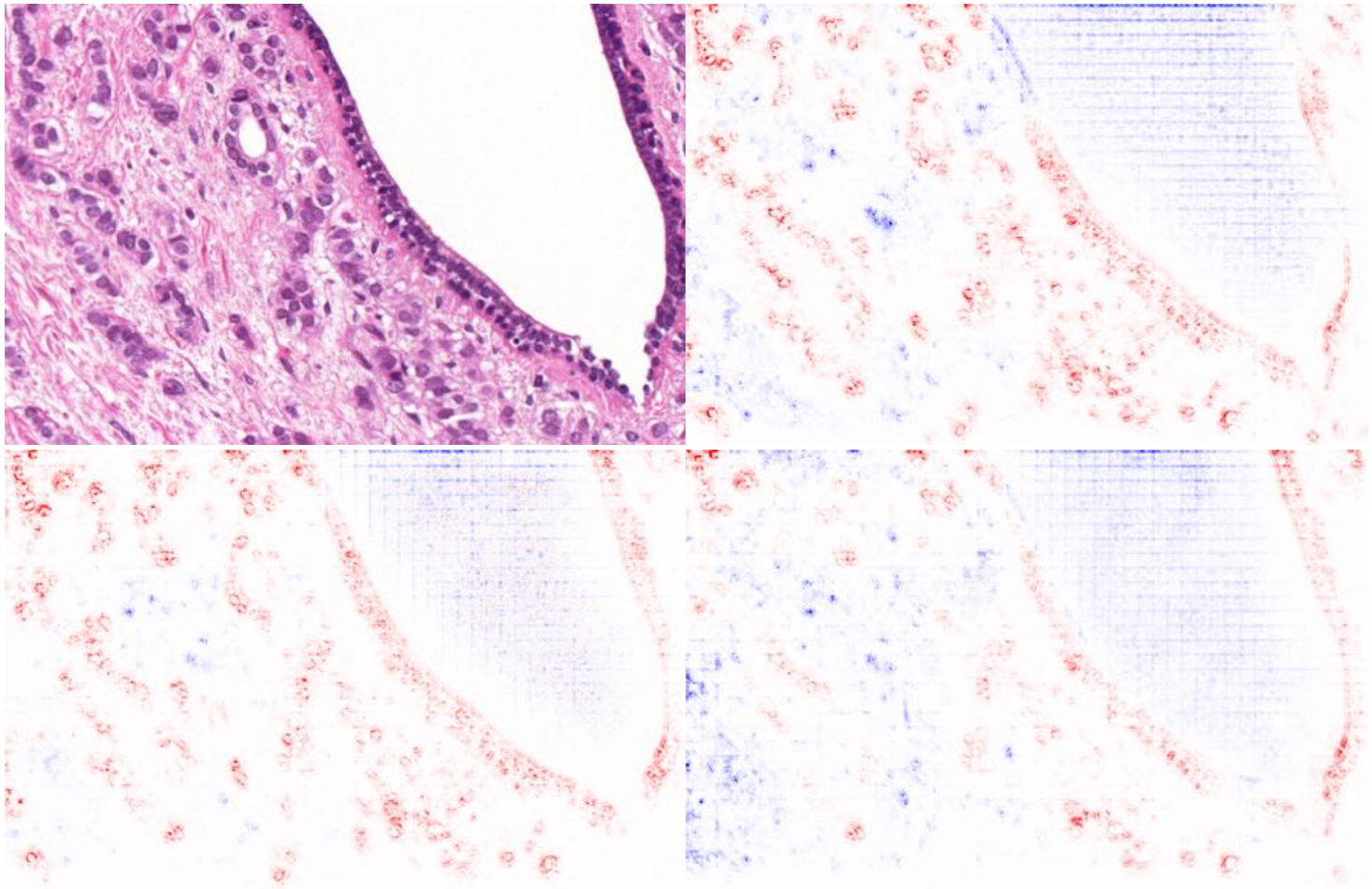
setup:

- HE stain, breast cancer
- positive annotations: positions of cancer nuclei
- negative annotations: ???

Batch composition

orig

1:1



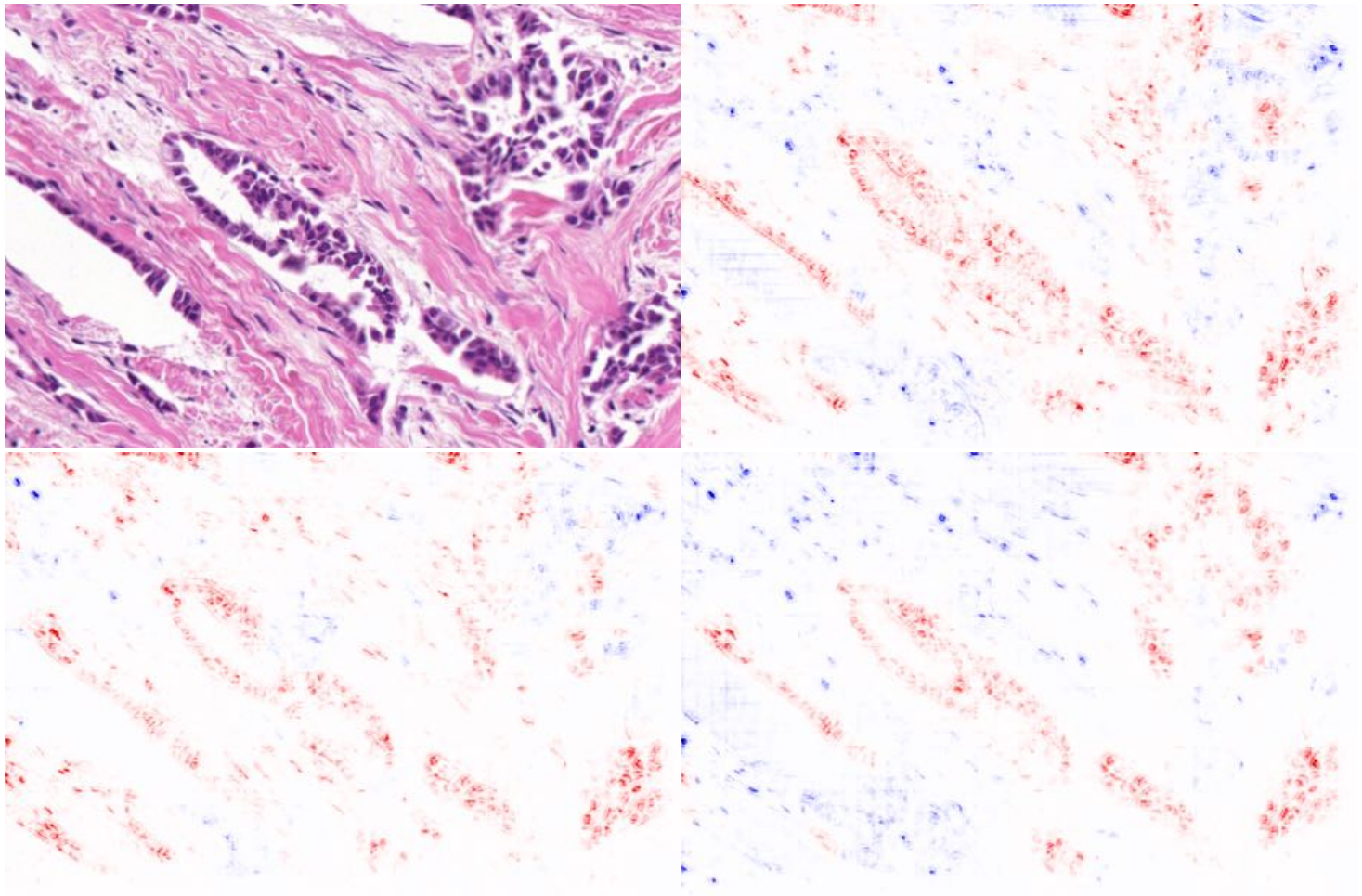
1.5:1

2:1

Batch composition

orig

1:1

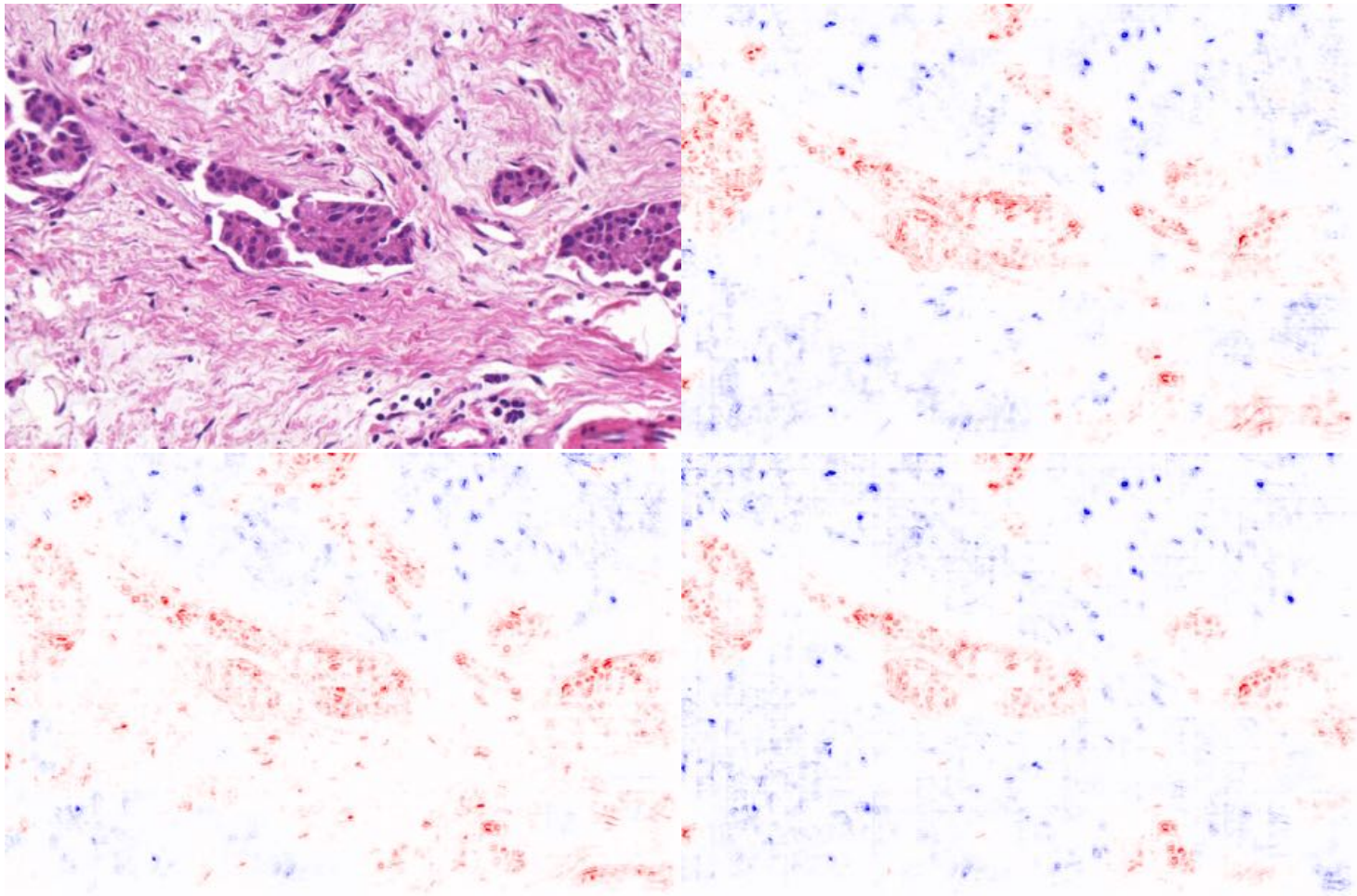


1.5:1

2:1

Batch composition

orig



1:1

1.5:1

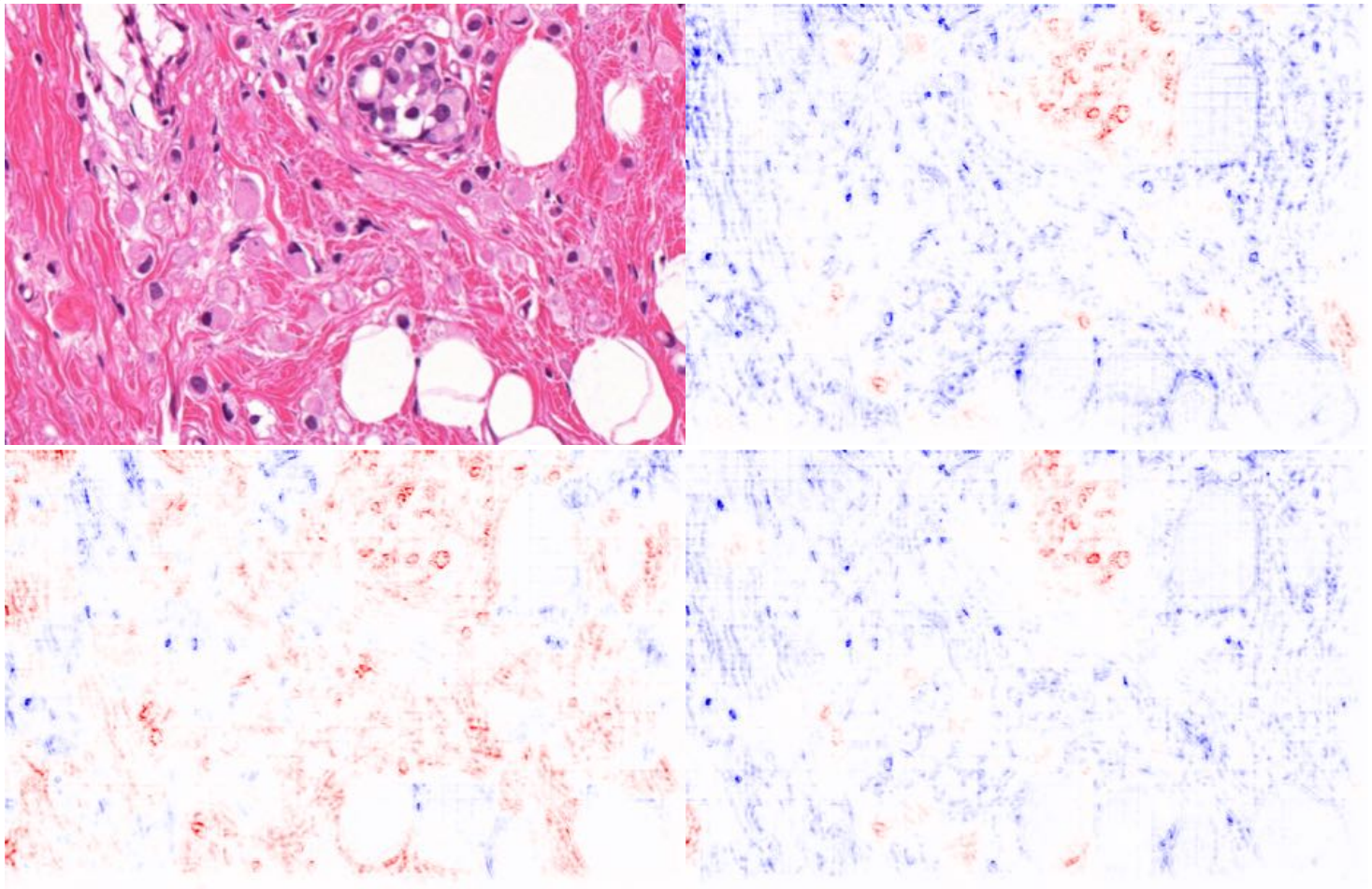
2:1

65.6 %

Batch composition

orig

1:1



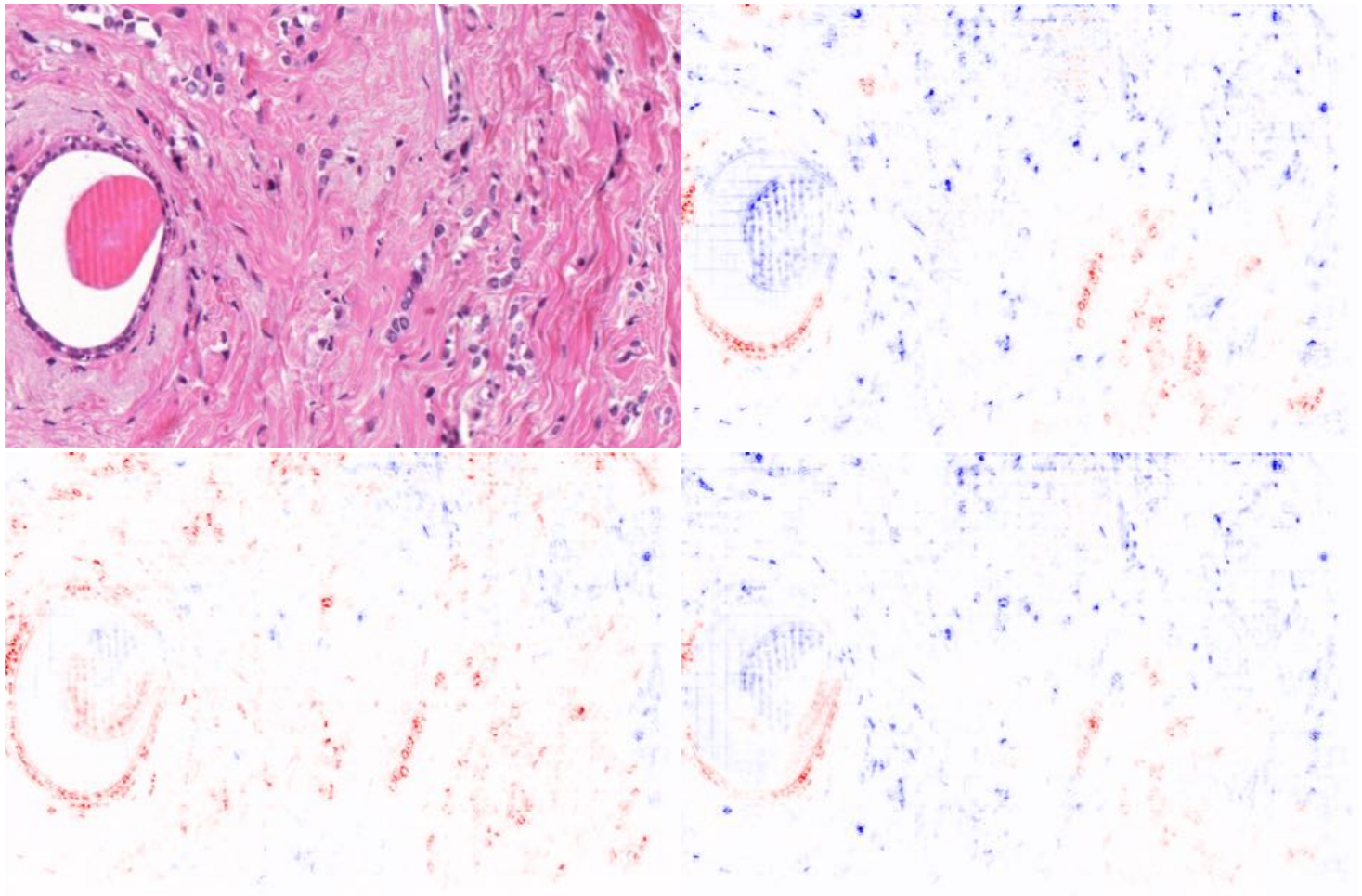
1.5:1

2:1

Batch composition

orig

1:1



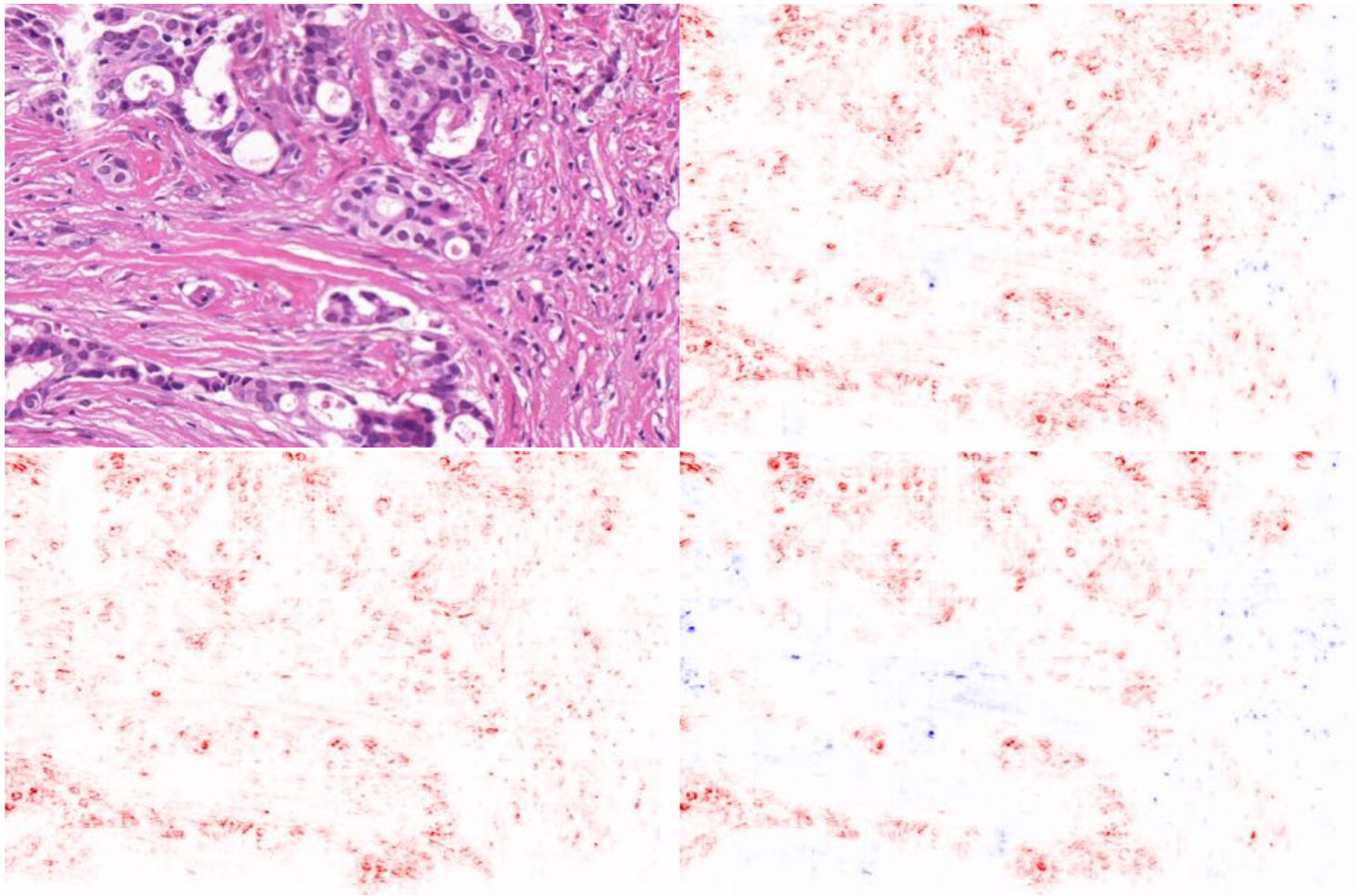
1.5:1

2:1

Batch composition

orig

1:1



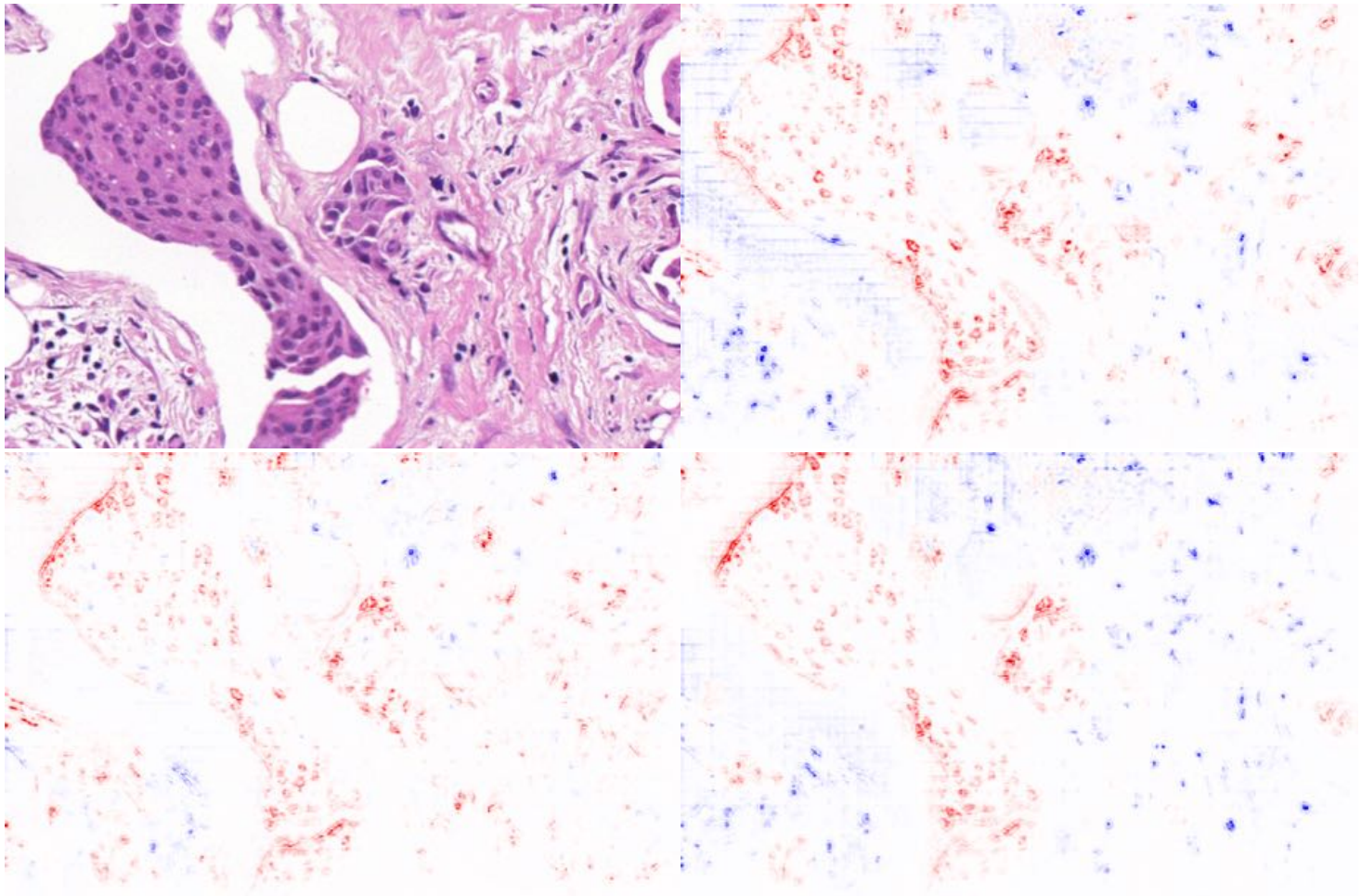
1.5:1

2:1

Batch composition

orig

1:1



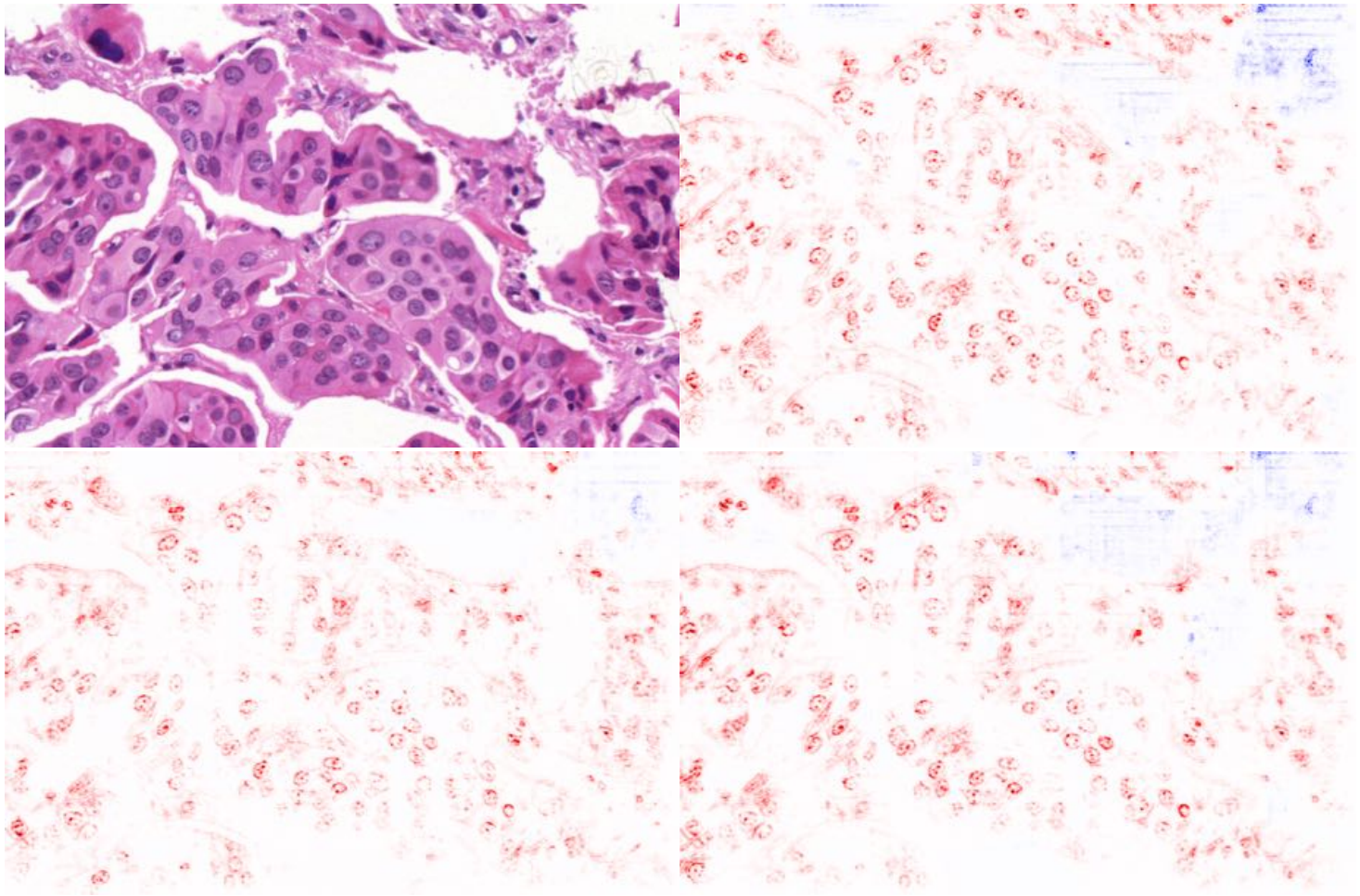
1.5:1

2:1

Batch composition

orig

1:1



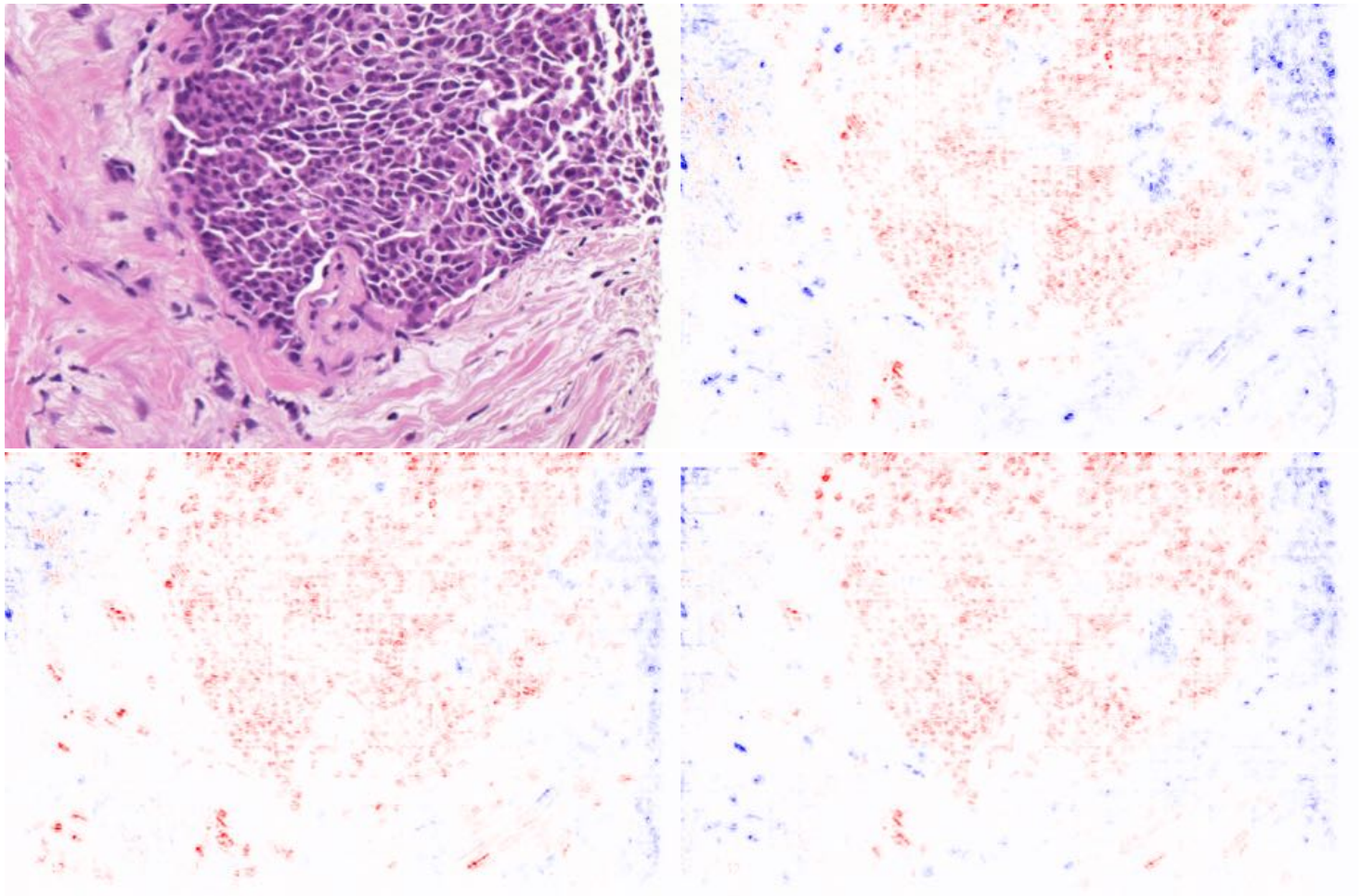
1.5:1

2:1

Batch composition

orig

1:1



1.5:1

2:1

Batch composition

observation: different bias learned, inconsistent to ratio

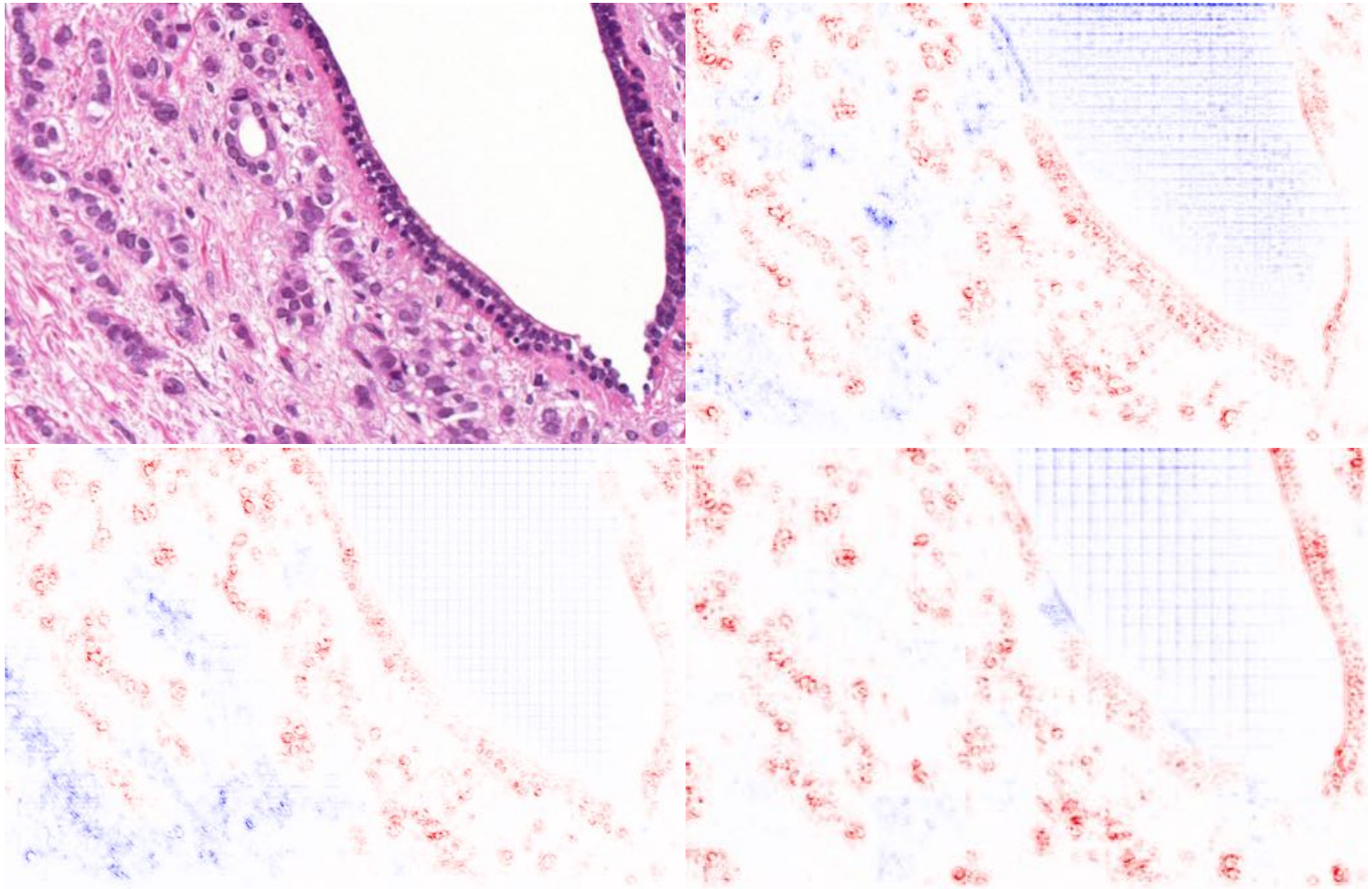
setup:

- HE stain, breast cancer
- positive annotations: positions of cancer nuclei
- negative annotations: (overlapping) windows without cancer nuclei
- preprocessing: shrink image to 80%,patchsize 120, grid stride 20
- Densenet 121, batchsize 8
- LRP-ϵ for FC layers, LRP-$\beta = 0$ for all others
- innvestigate toolbox with neuron selection index for cancer

Impact of Scaling

orig

80%



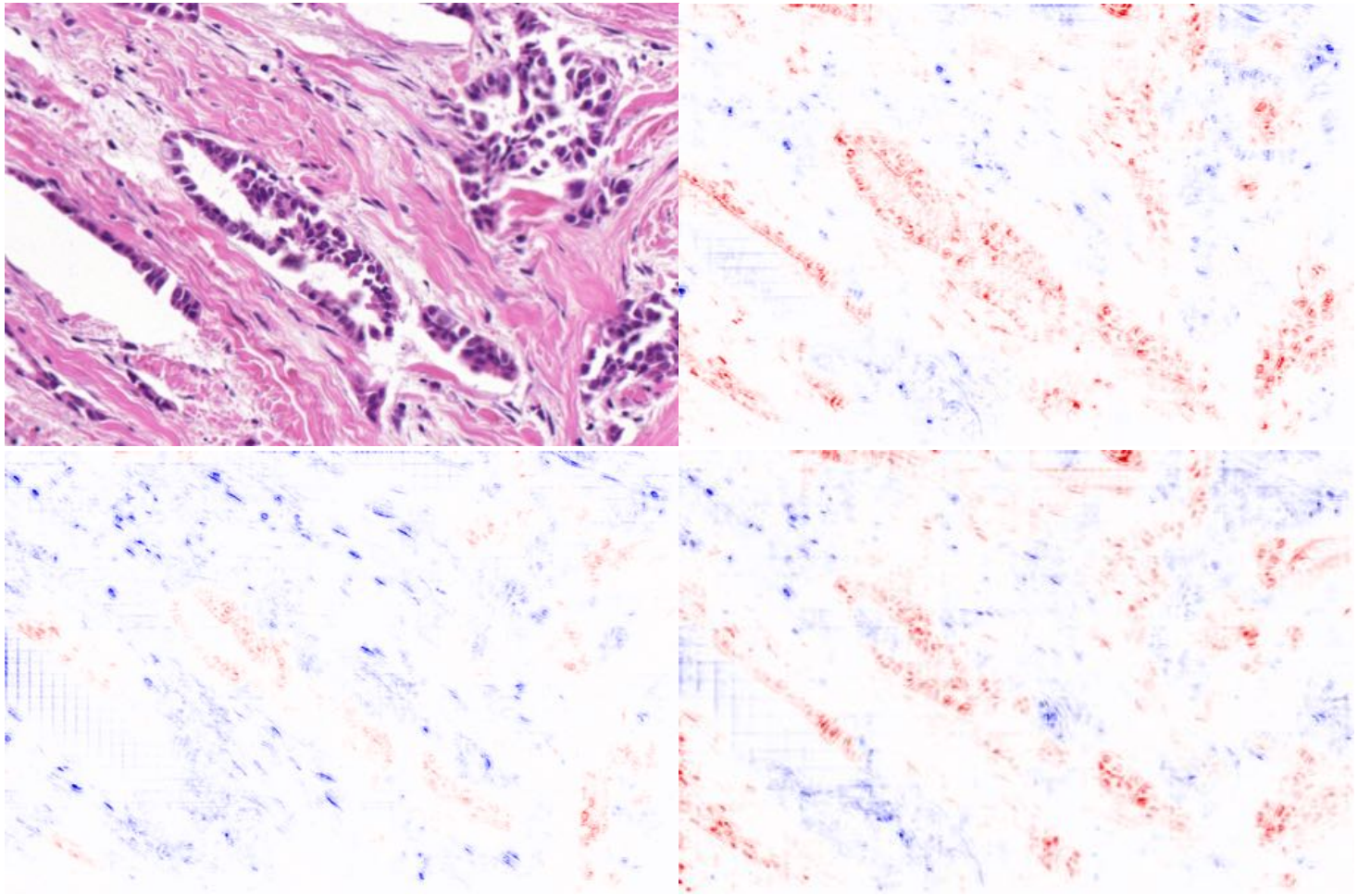
100%

66%

Impact of Scaling

orig

80%



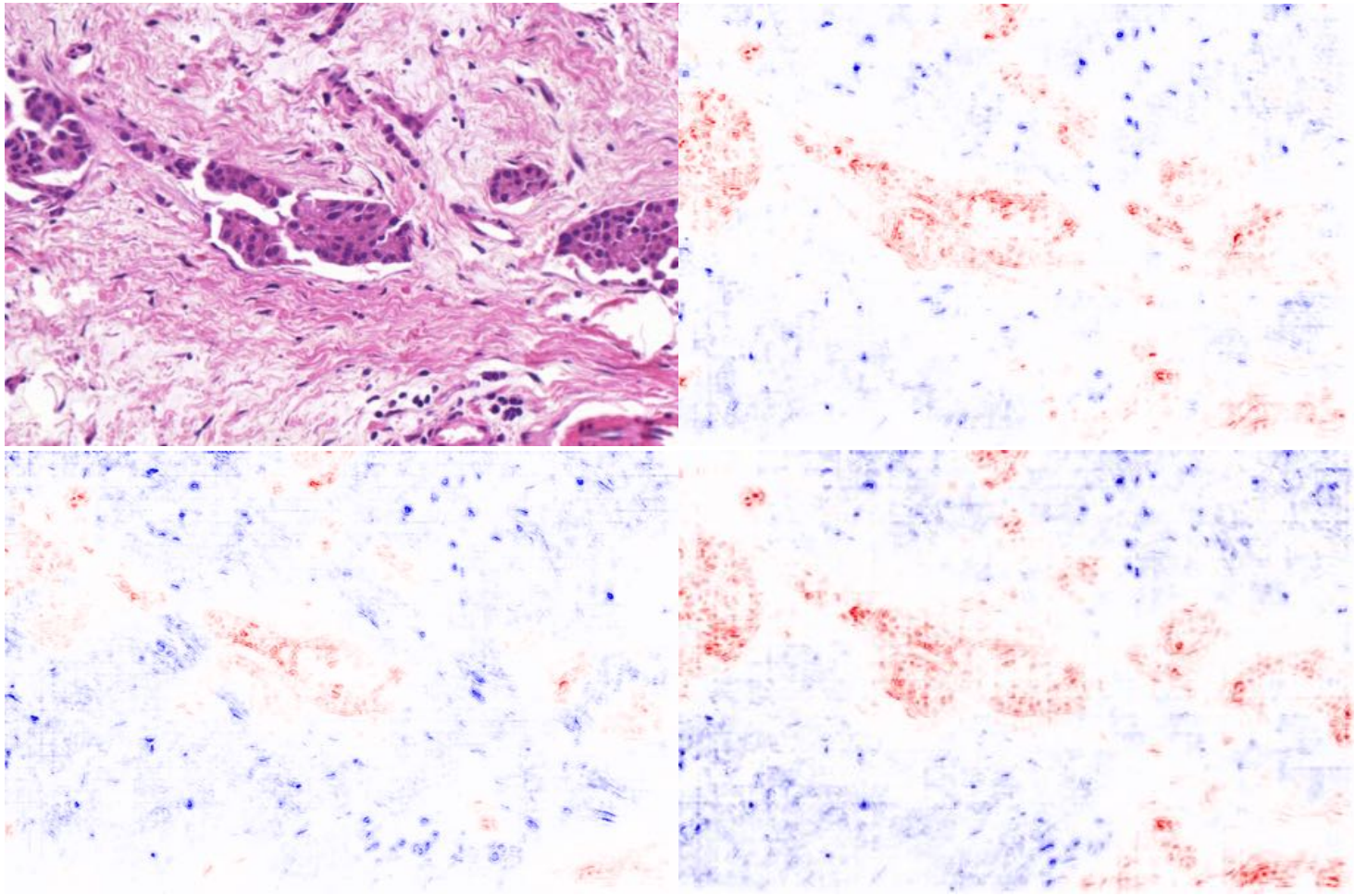
100%

66%

Impact of Scaling

orig

80%



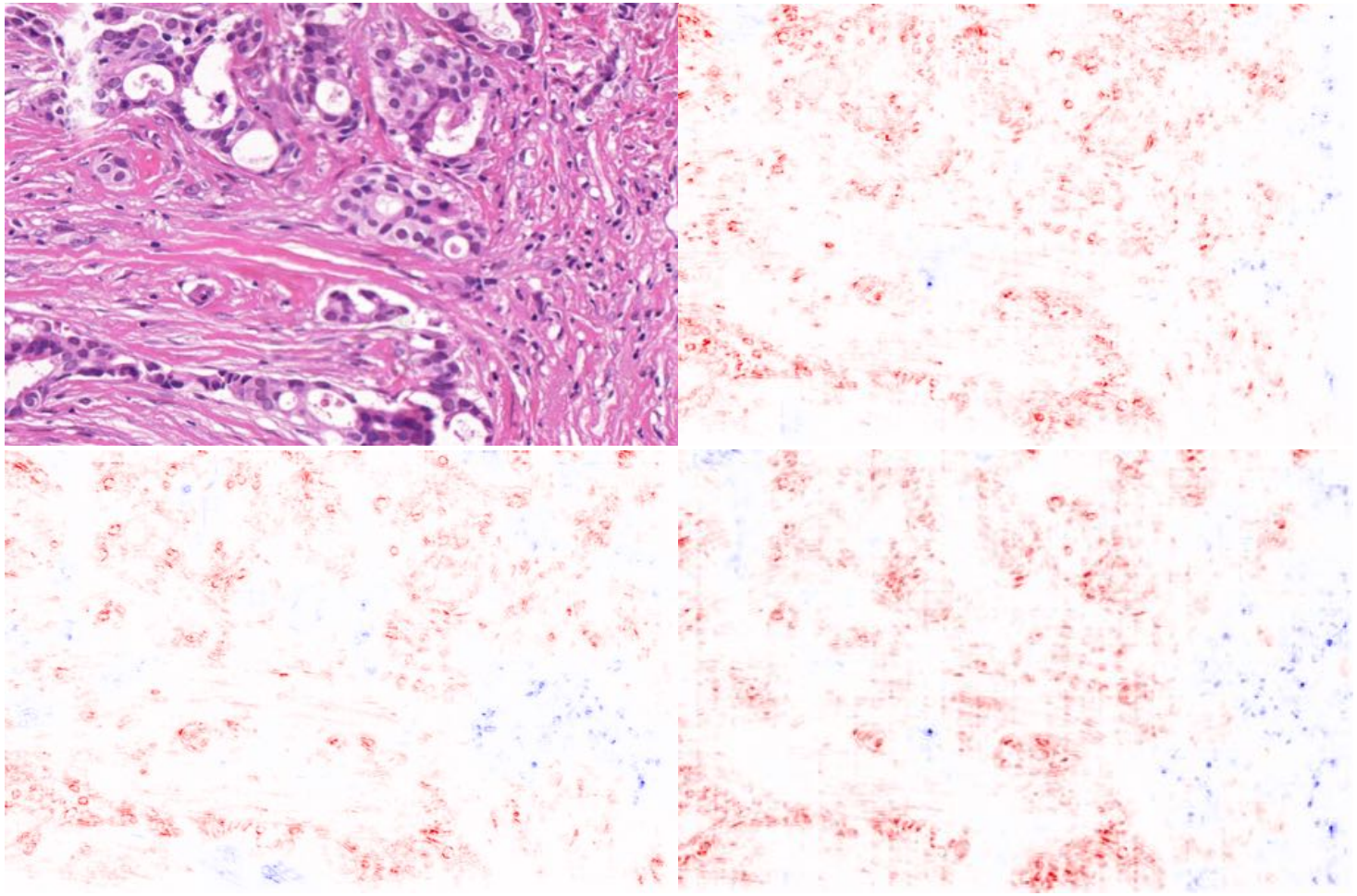
100%

66%

Impact of Scaling

orig

80%



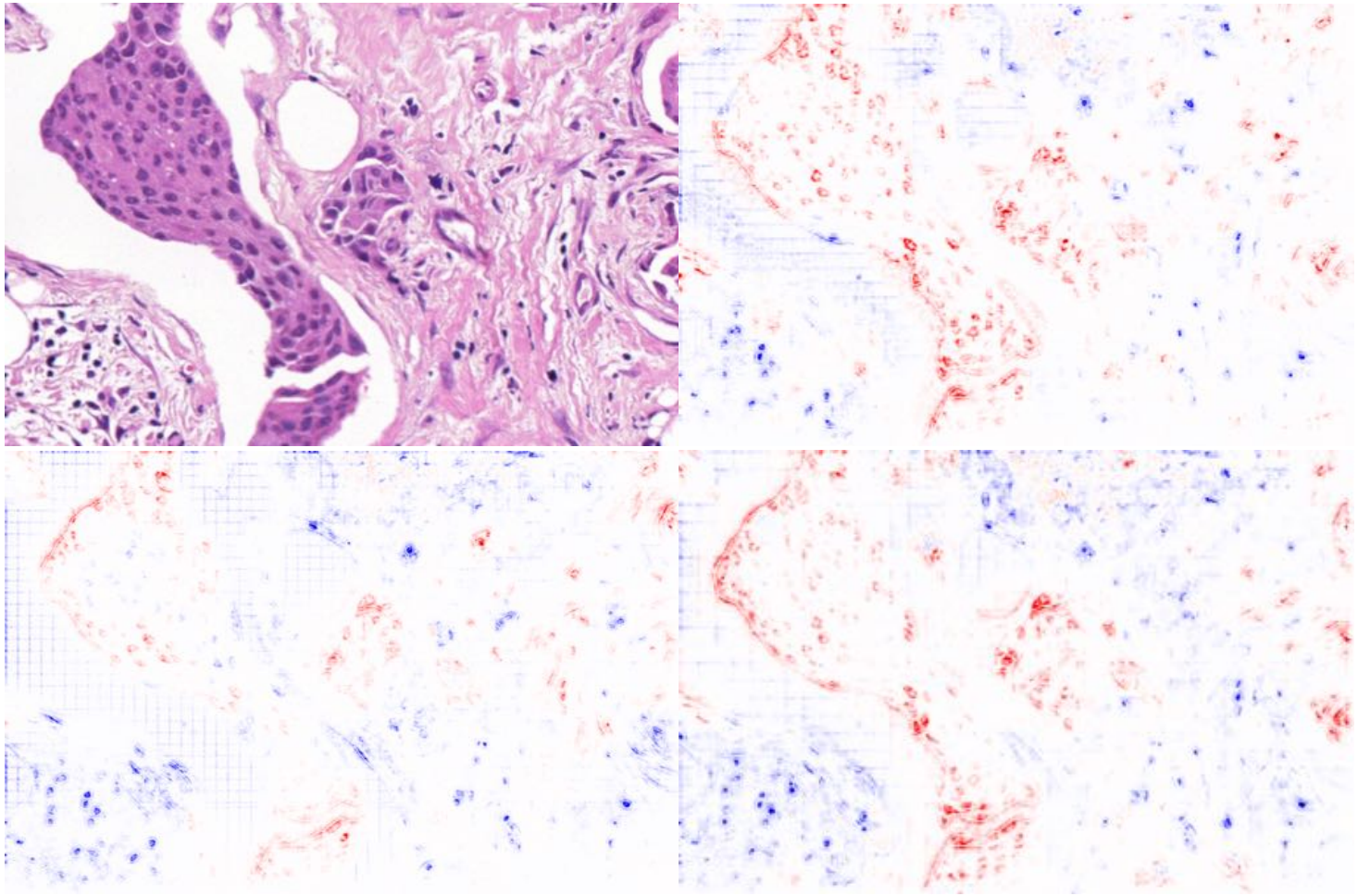
100%

66%

Impact of Scaling

orig

80%



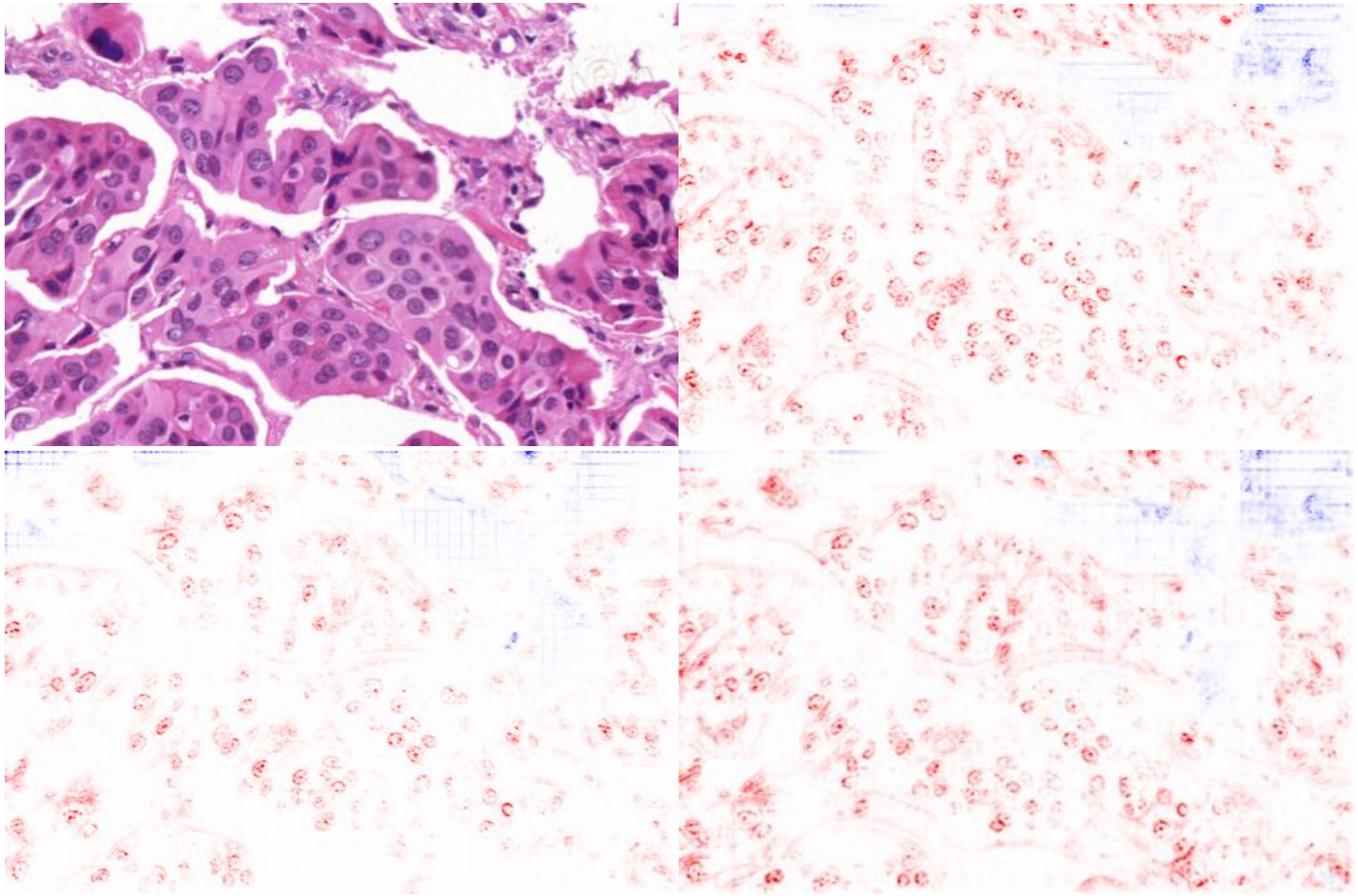
100%

66%

Impact of Scaling

orig

80%



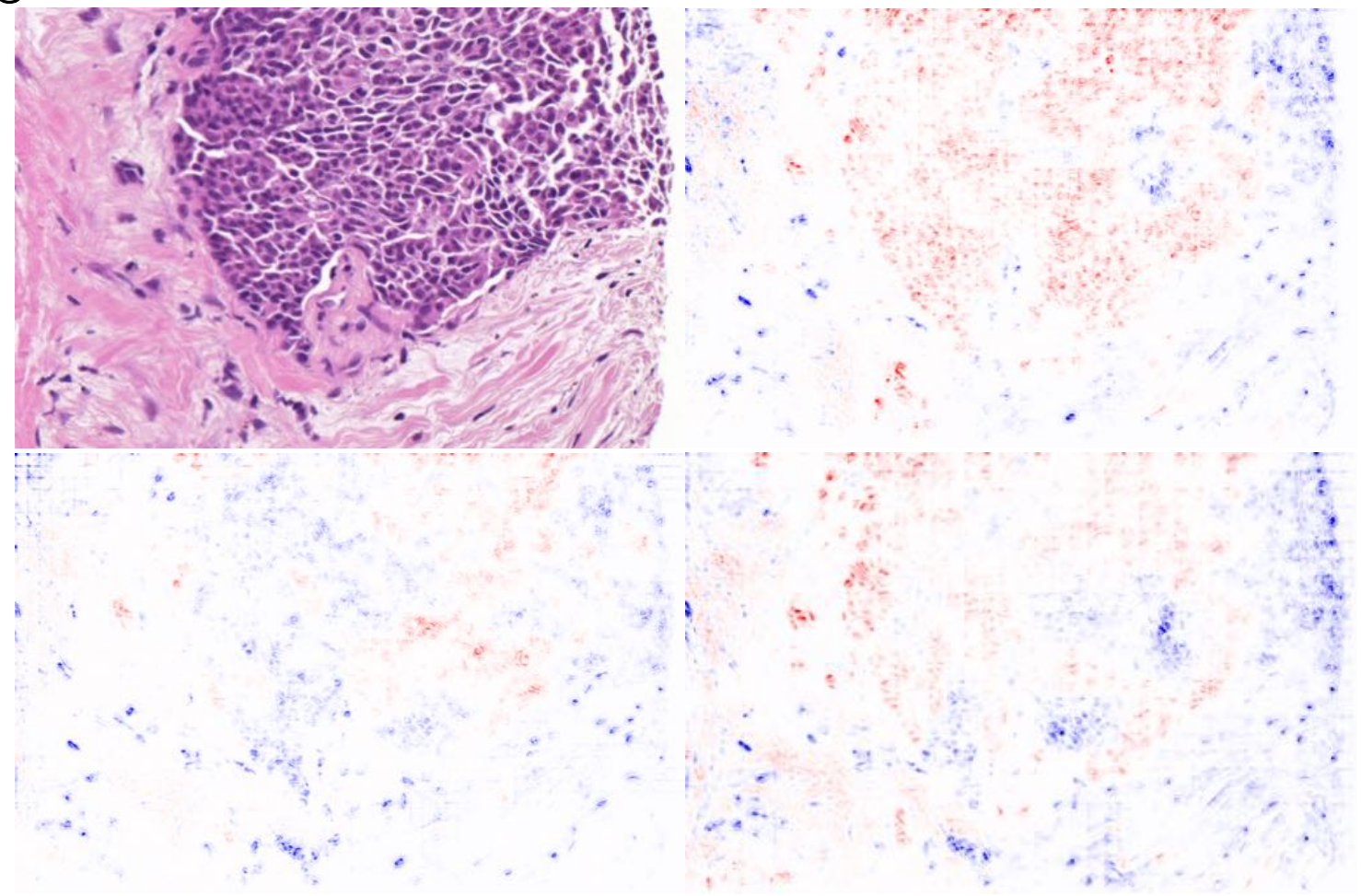
100%

66%

Impact of Scaling

orig

80%



100%

66%

Impact of Scaling

observation: 100% scaling: nuclei are too large for the fixed kernel sizes, difficult to recognize cancer, too faint heatmaps

Histopathology in application phase: heatmapping for acceptance

A classifier that simply tells a clinician: "its grade 3"

Histopathology in application phase: heatmapping for acceptance

A classifier that simply tells a clinician: "its grade 3"

Problem: in case of doubt clinician cannot validate the prediction. Classifier mistaken or clinician overlooked something?

Heatmapping allows to point the clinician to relevant regions.

Histopathology in application phase: heatmapping for acceptance

A classifier that simply tells a clinician: "its grade 3"

Problem: in case of doubt clinician cannot validate the prediction. Classifier mistaken or clinician overlooked something?

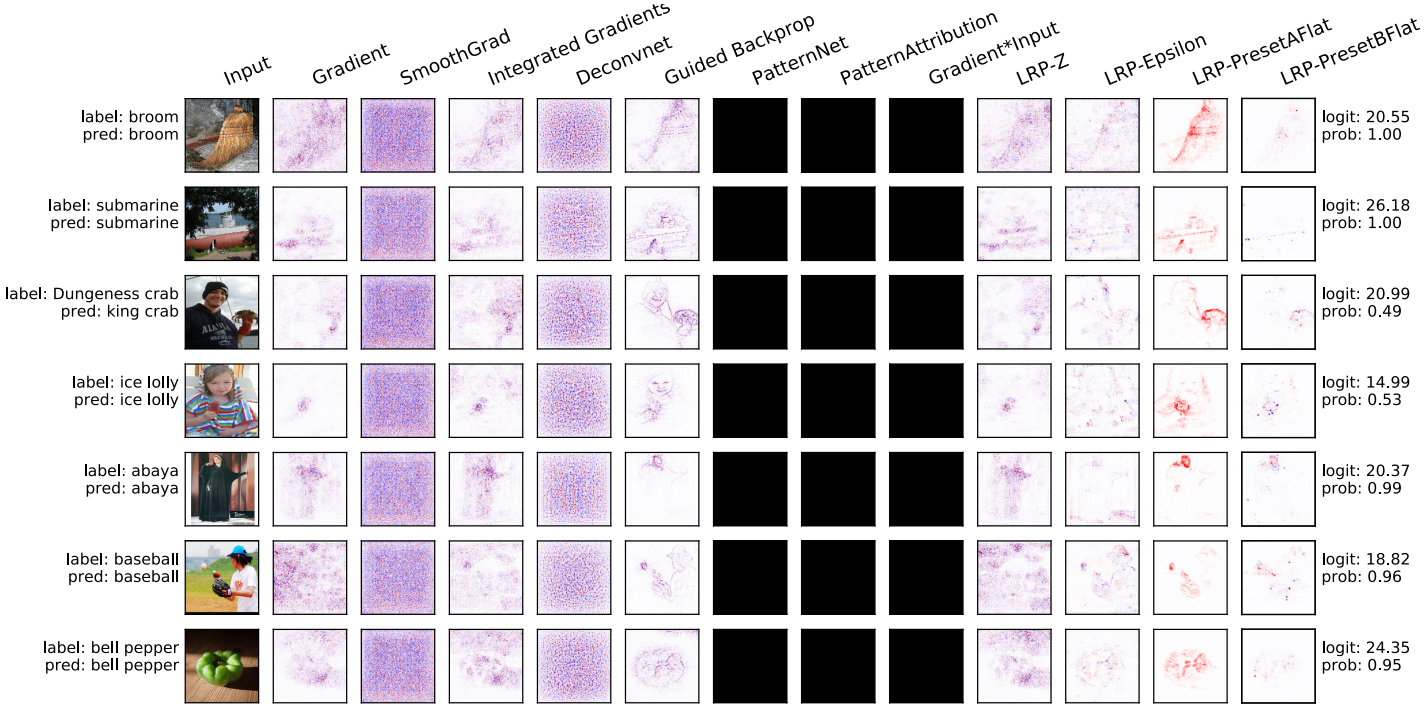
Heatmapping allows to point the clinician to relevant regions.
Heatmapping allows to identify nonsensical predictions on outlier samples.

Applications III: How well does LRP scale?

DenseNet-121

Made with Keras:

`https://github.com/albermax/innvestigate`



Thank you!

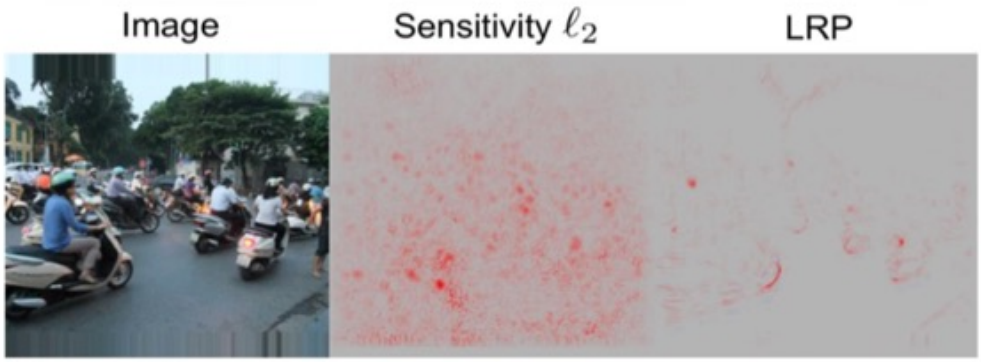
Links (LRP for LSTM for example):
http://www.heatmapping.org/
Tutorial: http://www.heatmapping.org/tutorial/
for Keras: https://github.com/albermax/innvestigate
LRPToolbox:
https://github.com/sebastian-lapuschkin/lrp_toolbox
Experimental MXnet integration:
https://github.com/sebastian-lapuschkin/lrp_toolbox/tree/python-wip/python
Demos: https://lrpserver.hhi.fraunhofer.de/

Tutorial on Interpretable Machine Learning

Wrap-up

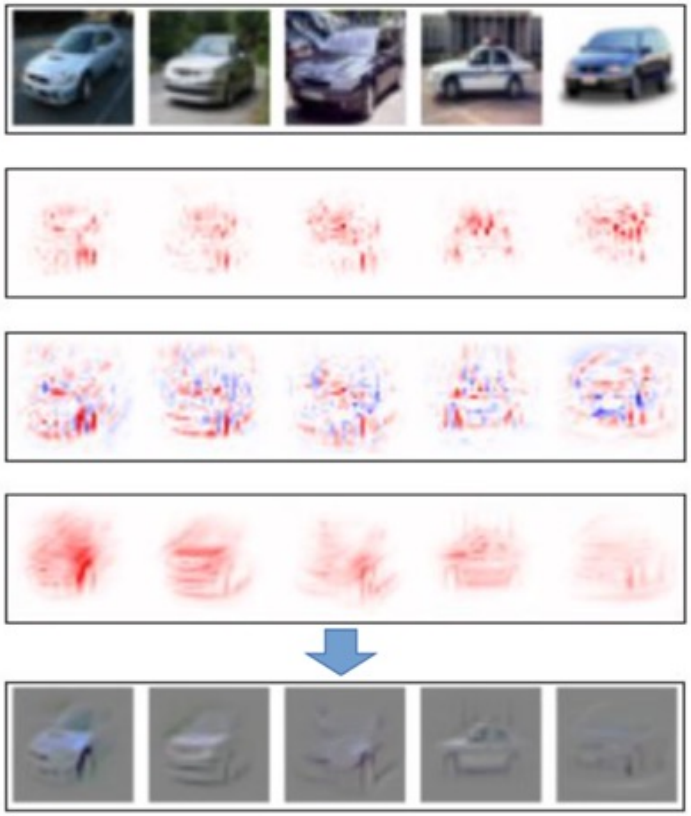
Take Home Messages

Sensitivity analysis is not the question that you would like to ask!



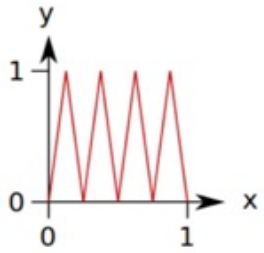
Take Home Messages

What works for simple models doesn't work for deep models.



gradient-based methods

vulnerable to shattered gradients



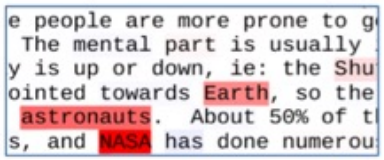
Our LRP method is robust to this.

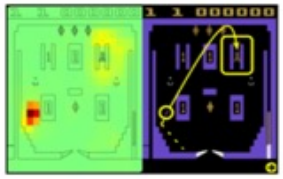
Take Home Messages

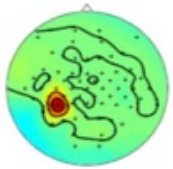
LRP works 4 all: deep models, LSTMs, kernel methods …

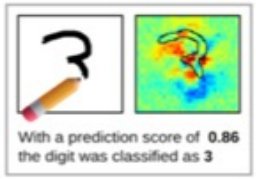
LRP Explanation Framework

(software, tutorials, demos, insights, applications)









Take Home Messages

$$\text{LRP} \neq \text{Gradient} \times \text{Input}$$

… except for special cases. LRP was developed among others because gradient-based methods aren't satisfying.

High flexibility: Different LRP variants, free parameters

Good news: No need to reimplement LRP, check our software at www.heatmapping.org.

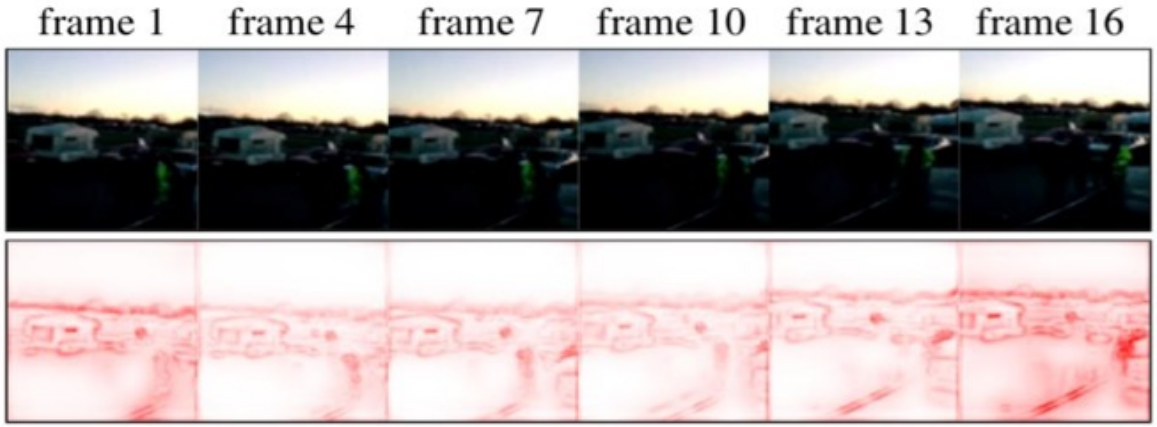
Take Home Messages

Explanations can be evaluated:
Pixel flipping (model agnostic)
And beyond LRP and DTD

[Samek et al. IEEE TNNLS 2017]

Take Home Messages

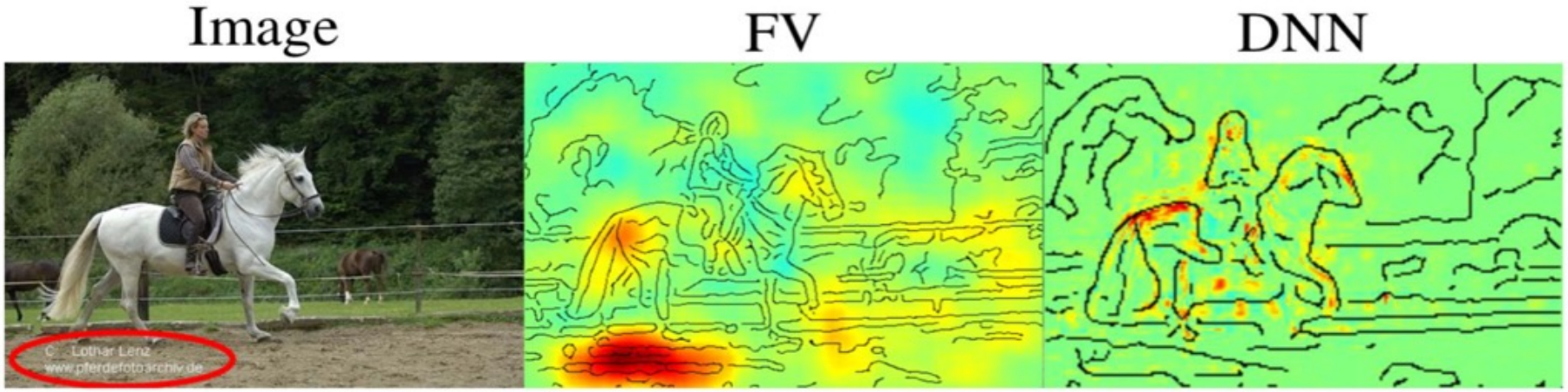
Explanation helps to improve models



Explaining ML, Now What?

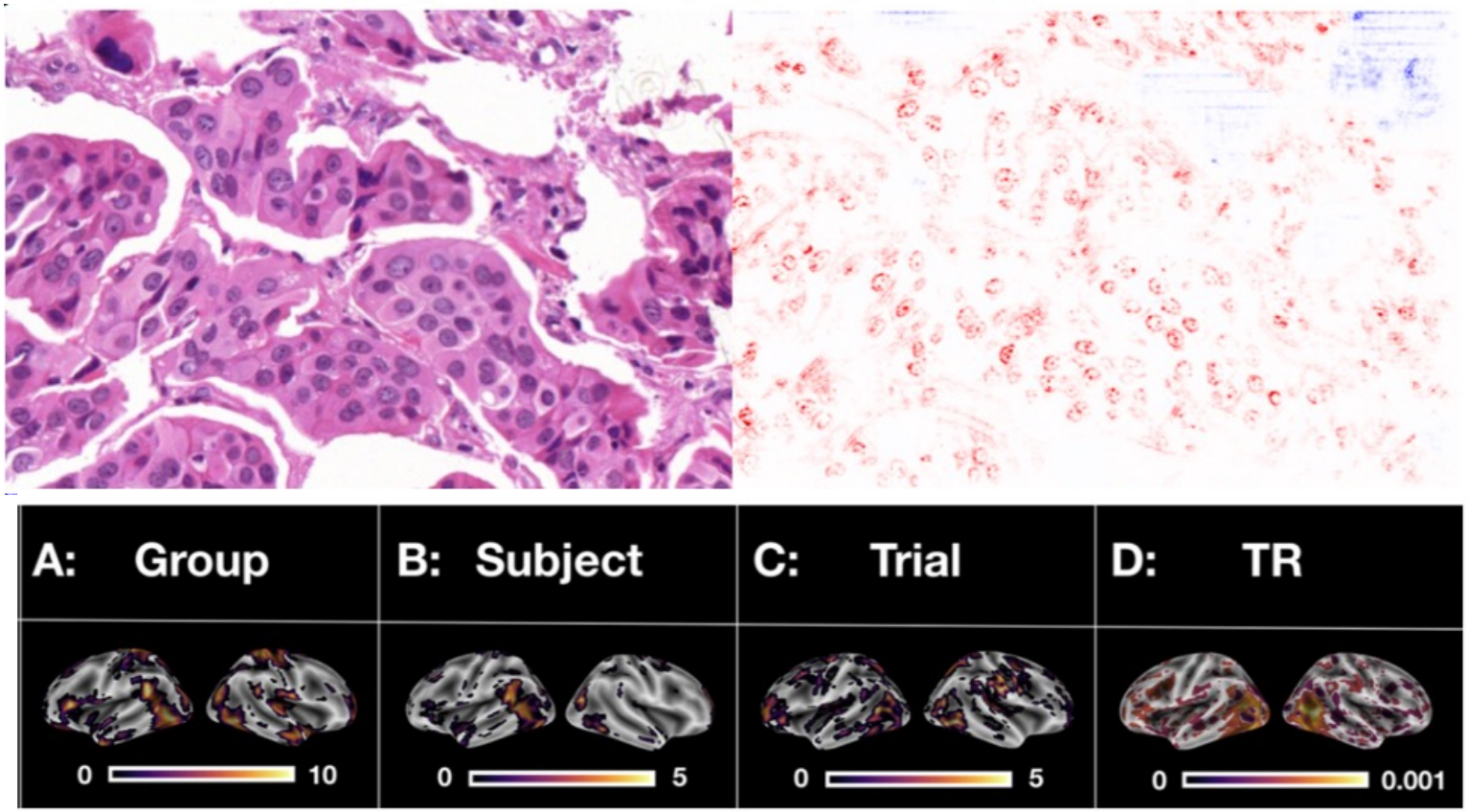
Take Home Messages

Explanation helps to find flaws in models



Take Home Messages

Getting new Insights in the Sciences

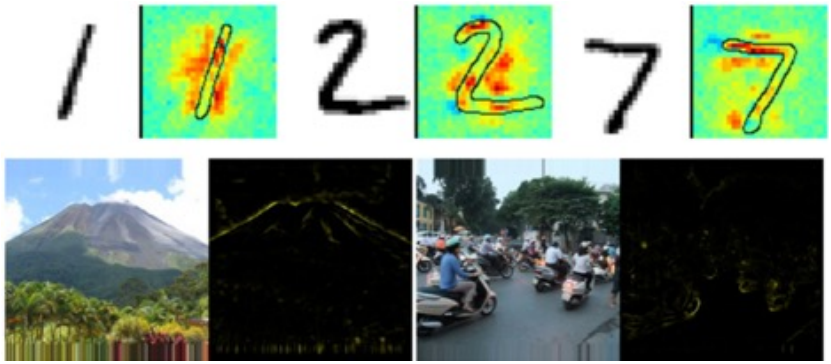


More information

Visit:

`http://www.heatmapping.org`

- Tutorials
- Software
- Online Demos



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`

References

Tutorial / Overview Papers

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

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, KR Müller, G Montavon. Towards Explaining Anomalies: A Deep Taylor Decomposition of One-Class Models. *arXiv:1805.06230*, 2018.

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.

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.

L Arras, A Osman, G Montavon, KR Müller, W Samek. Evaluating and Comparing Recurrent Neural Network Explanation Methods in NLP. *arXiv*, 2018.

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 Sciences

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*, 2018.

KT Schütt, F. Arbabzadah, S Chmiela, KR Müller, A Tkatchenko. Quantum-chemical insights from deep tensor neural networks. Nature communications, 8, 13890, 2017.

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.

F Horst, S Lapuschkin, W Samek, KR Müller, WI Schöllhorn. What is Unique in Individual Gait Patterns? Understanding and Interpreting Deep Learning in Gait Analysis. *arXiv:1808.04308*, 2018

References

Software

M Alber, S Lapuschkin, P Seegerer, M Hägele, KT Schütt, G Montavon, W Samek, KR Müller, S Dähne, PJ Kindermans. iNNvestigate neural networks!. *arXiv:1808.04260*, 2018.

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.