Interpretability

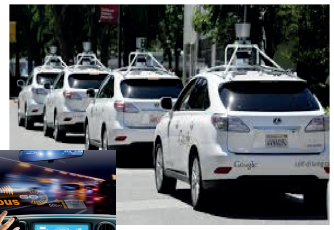
Gitta Kutyniok

(Technische Universität Berlin and University of Tromsø)

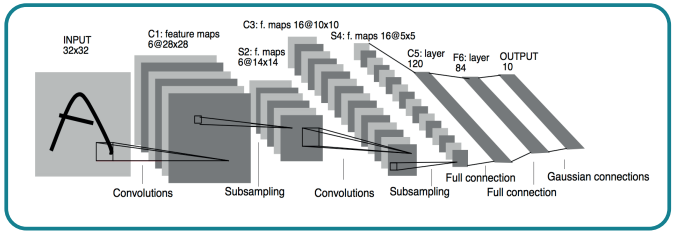
Banach Center – Oberwolfach Graduate Seminar:
Mathematics of Deep Learning
Polish Academy of Sciences, Będlewo, November 17 – 23, 2019

The Dawn of Deep Learning

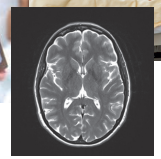
Self-Driving Cars



Surveillance



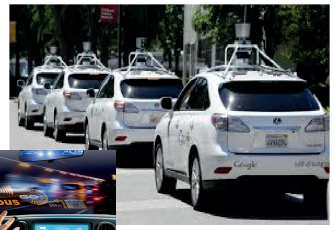
Legal Issues



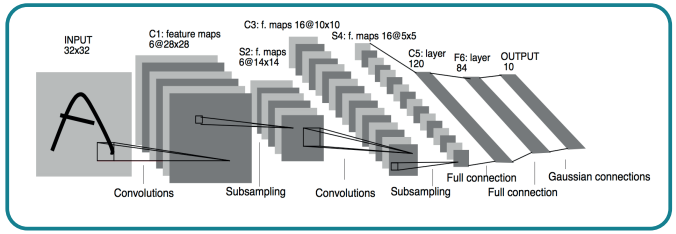
Health Care

The Dawn of Deep Learning

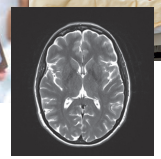
Self-Driving Cars



Surveillance



Legal Issues



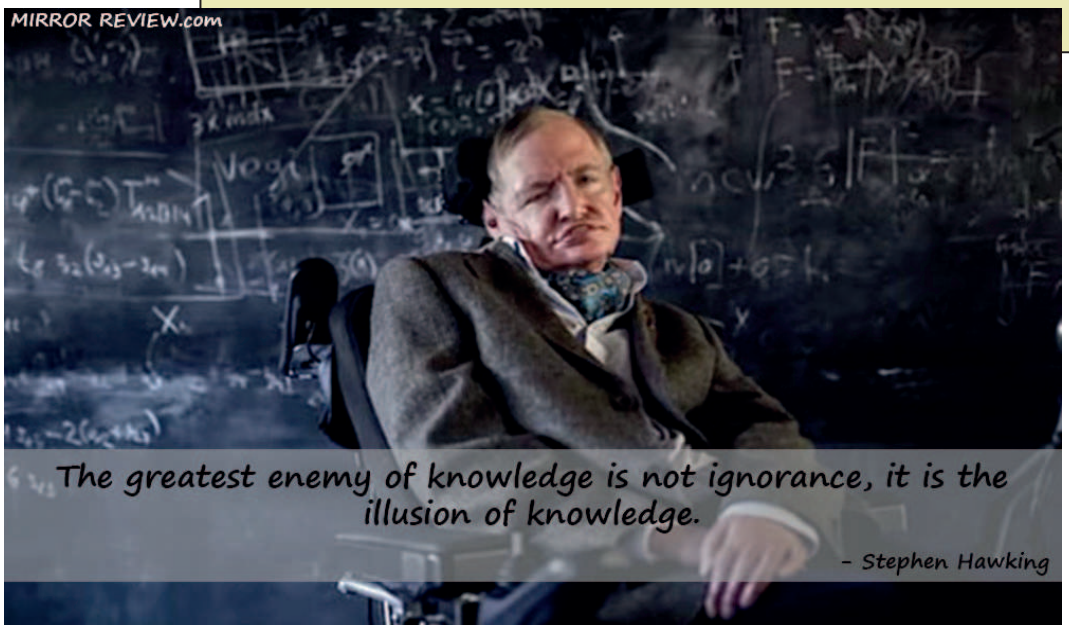
Health Care

Very few theoretical results explaining their success!

Deep Learning = Alchemy?

„Ali Rahimi, a researcher in artificial intelligence (AI) at Google in San Francisco, California, took a swipe at his field last December—and received a 40-second ovation for it. Speaking at an AI conference, Rahimi charged that **machine learning algorithms, in which computers learn through trial and error, have become a form of „alchemy."** Researchers, he said, **do not know why some algorithms work and others don't, nor do they have rigorous criteria for choosing one AI architecture over another**...."

Science, May 2018



Fundamental Questions concerning Deep Neural Networks

- *Expressivity:*
  - How powerful is the network architecture?
  - Can it indeed represent the correct functions?

  $\rightsquigarrow$ *Applied Harmonic Analysis, Approximation Theory, ...*
- *Learning:*
  - Why does the current learning algorithm produce anything reasonable?
  - What are good starting values?

  $\rightsquigarrow$ *Differential Geometry, Optimal Control, Optimization, ...*
- *Generalization:*
  - Why do deep neural networks perform that well on data sets, which do not belong to the input-output pairs from a training set?
  - What impact has the depth of the network?

  $\rightsquigarrow$ *Learning Theory, Optimization, Statistics, ...*

Fundamental Questions concerning Deep Neural Networks

- *Expressivity:*
  - How powerful is the network architecture?
  - Can it indeed represent the correct functions?

  $\rightsquigarrow$ *Applied Harmonic Analysis, Approximation Theory, ...*
- *Learning:*
  - Why does the current learning algorithm produce anything reasonable?
  - What are good starting values?

  $\rightsquigarrow$ *Differential Geometry, Optimal Control, Optimization, ...*
- *Generalization:*
  - Why do deep neural networks perform that well on data sets, which do not belong to the input-output pairs from a training set?
  - What impact has the depth of the network?

  $\rightsquigarrow$ *Learning Theory, Optimization, Statistics, ...*
- *Interpretability:*
  - Why did a trained deep neural network reach a certain decision?
  - Which components of the input do contribute most?

  $\rightsquigarrow$ *Information Theory, Uncertainty Quantification, ...*

Interpretability

Origin of Interpretability

Classification as a Classical Task for Neural Networks:

- Which features are most relevant for the decision?
  - Treat every pixel separately
  - Consider combinations of pixels
  - Incorporate additional knowledge
- How certain is the decision?

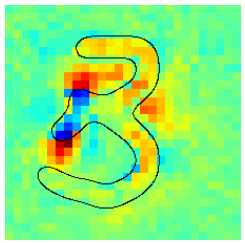
Origin of Interpretability

Classification as a Classical Task for Neural Networks:

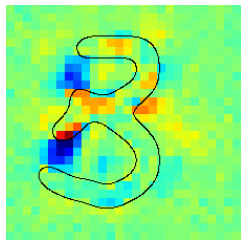
- Which features are most relevant for the decision?
  - Treat every pixel separately
  - Consider combinations of pixels
  - Incorporate additional knowledge
- How certain is the decision?

Illustration of a Relevance Map:

map for digit 3



map for digit 8



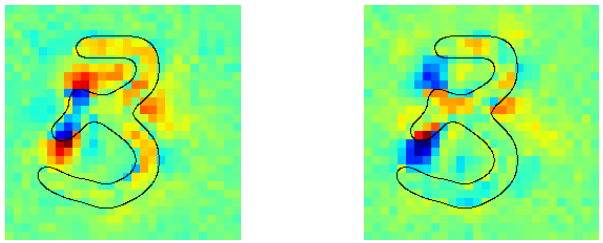
Origin of Interpretability

Classification as a Classical Task for Neural Networks:

- Which features are most relevant for the decision?
  - Treat every pixel separately
  - Consider combinations of pixels
  - Incorporate additional knowledge
- How certain is the decision?

Illustration of a Relevance Map:

map for digit 3 map for digit 8



Does this provide sufficient knowledge about a network decision?

History of the Field

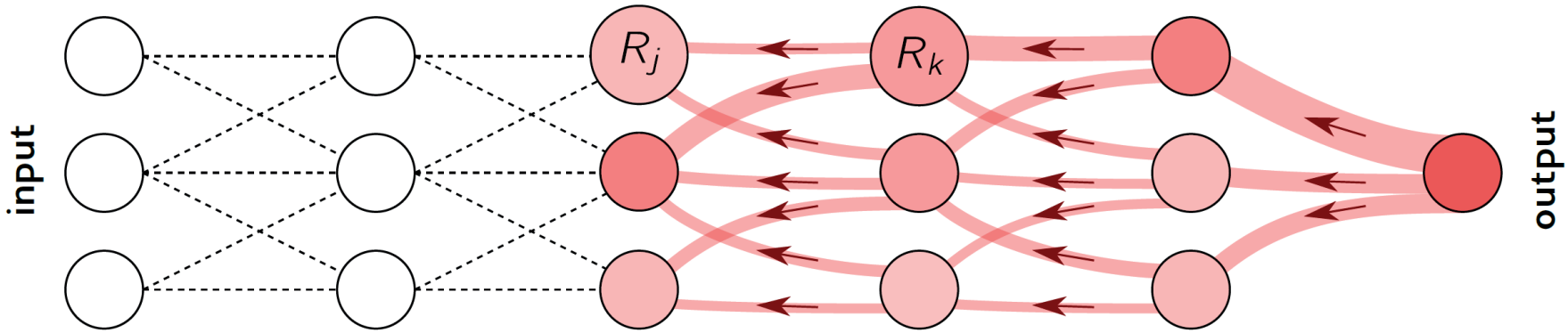
Previous Relevance Mapping Methods:

- Gradient based methods:
  - Sensitivity Analysis (Baehrens, Schroeter, Harmeling, Kawanabe, Hansen, Müller, 2010)
  - SmoothGrad (Smilkov, Thorat, Kim, Viégas, Wattenberg, 2017)
- Backwards propagation based methods:
  - Guided Backprop (Springenberg, Dosovitskiy, Brox, Riedmiller, 2015)
  - Layer-wise Relevance Propagation (Bach, Binder, Montavon, Klauschen, Müller, Samek, 2015)
  - Deep Taylor (Montavon, Samek, Müller, 2018)
- Surrogate model based methods:
  - LIME (Local Interpretable Model-agnostic Explanations) (Ribeiro, Singh, Guestrin, 2016)
- Game theoretic methods:
  - Shapley values (Shapley, 1953), (Kononenko, Štrumbelj, 2010)
  - SHAP (Shapley Additive Explanations) (Lundberg, Lee, 2017)

Layer-Wise Relevance Propagation (LRP)

Idea of LRP

Illustration:



$$R_j = \sum_k \frac{a_j w_{jk}}{\sum_{0,j} a_j w_{jk}} R_k$$

Deep Taylor Decomposition

Relevance Maps

Goal: Consider the realization of a neural network

$$f : \mathbb{R}^d \to \mathbb{R}.$$

Determine the relevance of each x_p of $x = (x_1, \ldots x_d)$ for the output $f(x)$.

Definition:
A collection $R = (R_p)_{p=1}^d$ of functions $R_p : \mathbb{R}^d \to \mathbb{R}$ is called *relevance map*.

- R is *non-negative*, if $R_p(x) \geq 0$ for all p, x.
- R is *conservative* with respect to f, if

$$\sum_p R_p(x) = f(x).$$

- R is *consistent*, if it is both non-negative and conservative.

Remark: Consistency implies that the decision $f(x)$ is distributed among the input pixels and only the contribution towards the decision is counted (not against it).

Sensitivity Analysis

Definition:

Assume f is continuously differentiable. Then

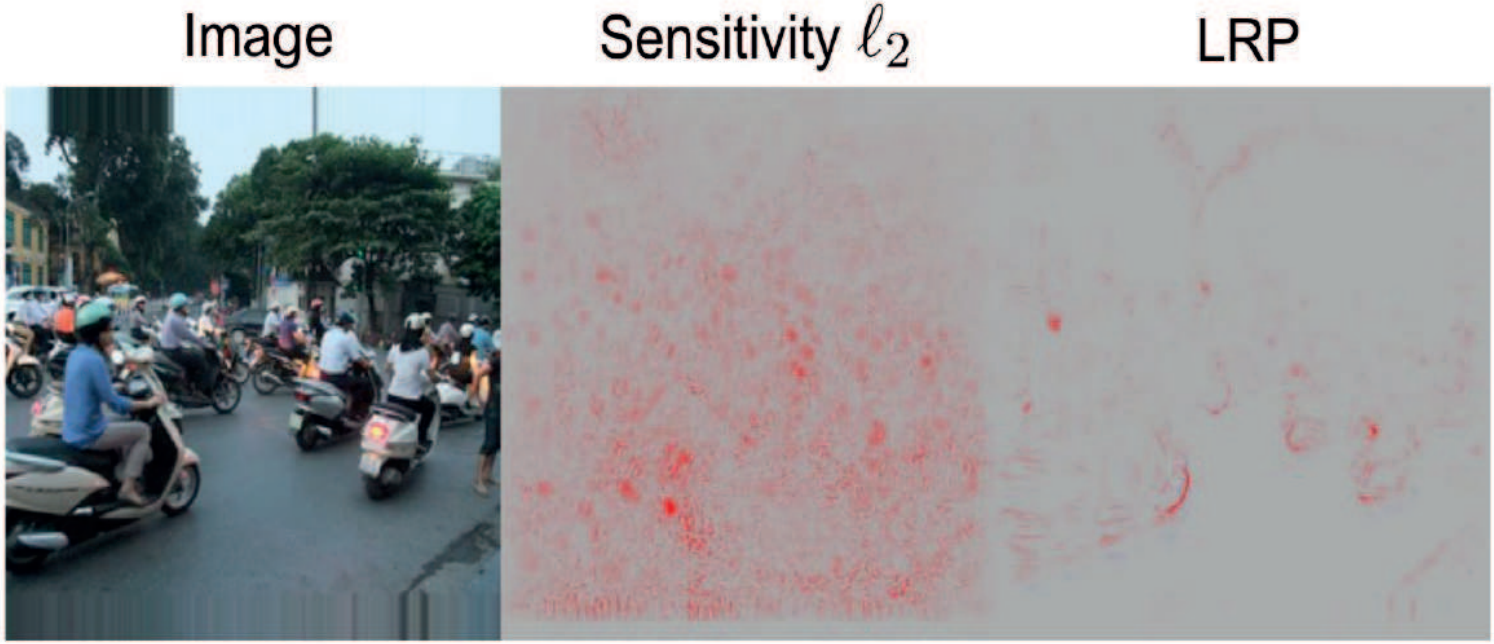
$$R_p(x) := \left(\frac{\partial f(x)}{\partial x_p}\right)^2$$

is a relevance map called *sensitivity analysis*.

Remarks:

- This relevance map is non-negative, but not conservative or consistent.
- Sensitivity analysis only uses ∇f, but not the decision $f(x)$. It answers the question "Changing which pixels makes the image look less/more like a cat?", but not "Which pixels make the image a cat?".

Numerical Experiment for Sensitivity versus LRP



(Source: Samek; 2018)

Taylor Decomposition Relevance Map

Definition (Müller et al.; 2017):

Assume f is continuously differentiable and $\tilde{x}$ a suitably chosen root point of f, for instance $f(\tilde{x}) = 0$. Then

$$R_p(x) := \frac{\partial f(\tilde{x})}{\partial x_p}(x_p - \tilde{x}_p)$$

is the *Taylor decomposition relevance map*.

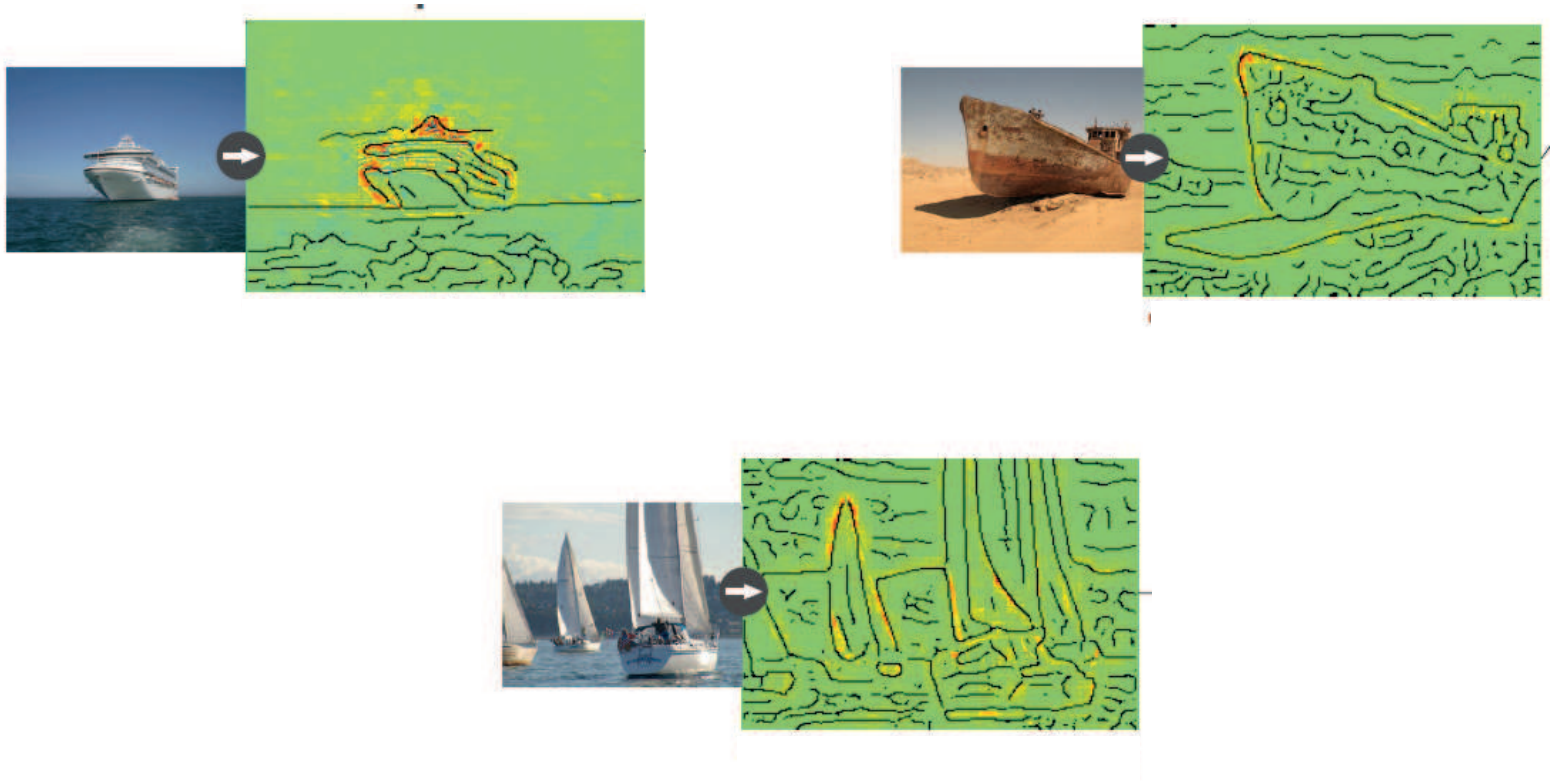
Remarks:

- The idea is to choose a root point $\tilde{x}$ near x which is neutral with respect to f in the sense of $f(\tilde{x}) = 0$.
- Up to second-order terms, this relevance map is conservative:

$$\begin{aligned} f(x) &= f(\tilde{x}) + \nabla f(\tilde{x})^T (x - \tilde{x}) + O(\|x - \tilde{x}\|^2) \\ &= \sum_p R_p(x) + O(\|x - \tilde{x}\|^2). \end{aligned}$$

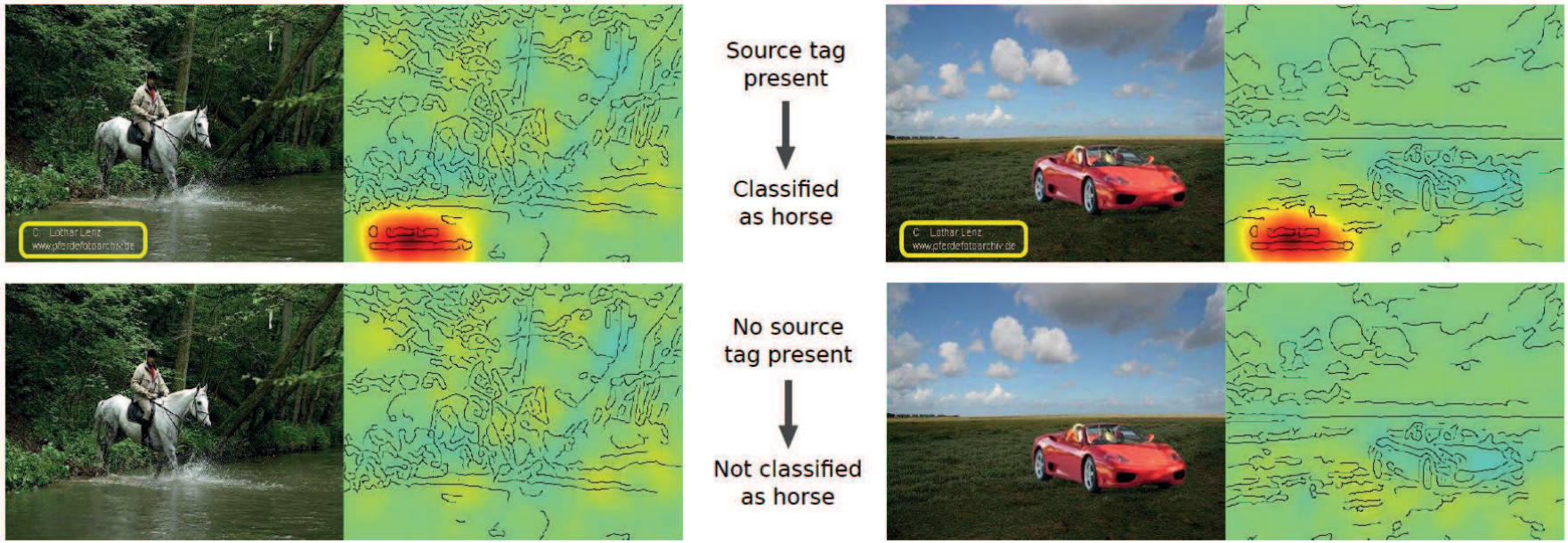
- This relevance map can be efficiently computed by starting at $f(x)$ and going reversely through the network.

Numerical Experiment for Deep Taylor, I



(Source: Laupischkin et al.; 2019)

Numerical Experiment for Deep Taylor, II



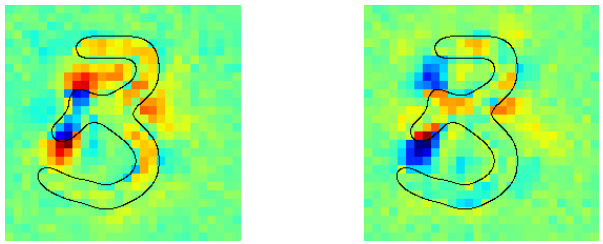
(Source: Laupischkin et al.; 2019)

Towards a More Mathematical Understanding

What is Relevance?

Main Goal: We aim to understand decisions of "black-box" predictors!

map for digit 3 map for digit 8



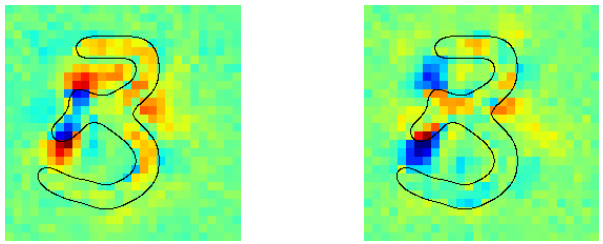
Challenges:

- What exactly is relevance in a mathematical sense?
- What is a good relevance map?
- How to compare different relevance maps?

What is Relevance?

Main Goal: We aim to understand decisions of "black-box" predictors!

map for digit 3 map for digit 8



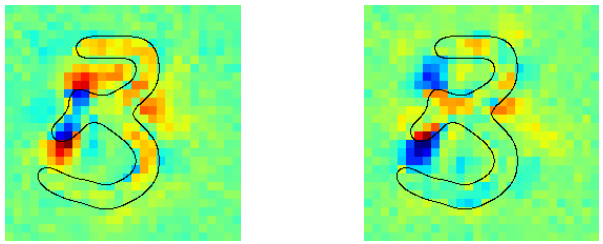
Challenges:

- What exactly is relevance in a mathematical sense?
  $\rightsquigarrow$ *Rigorous definition of relevance by information theory.*
- What is a good relevance map?
- How to compare different relevance maps?

What is Relevance?

Main Goal: We aim to understand decisions of "black-box" predictors!

map for digit 3 map for digit 8



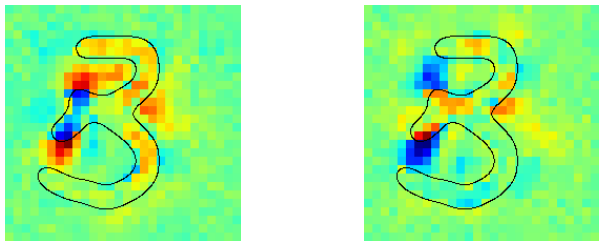
Challenges:

- What exactly is relevance in a mathematical sense?
  $\rightsquigarrow$ *Rigorous definition of relevance by information theory.*
- What is a good relevance map?
  $\rightsquigarrow$ *Formulation of interpretability as optimization problem.*
  $\rightsquigarrow$ *Theoretical analysis of complexity.*
- How to compare different relevance maps?

What is Relevance?

Main Goal: We aim to understand decisions of "black-box" predictors!

map for digit 3 map for digit 8



Challenges:

- What exactly is relevance in a mathematical sense?
  $\rightsquigarrow$ *Rigorous definition of relevance by information theory.*
- What is a good relevance map?
  $\rightsquigarrow$ *Formulation of interpretability as optimization problem.*
  $\rightsquigarrow$ *Theoretical analysis of complexity.*
- How to compare different relevance maps?
  $\rightsquigarrow$ *Canonical framework for comparison.*

The Relevance Mapping Problem

The Relevance Mapping Problem

The Setting: Let

- $\Phi : [0,1]^d \to [0,1]$ be a classification function,
- $x \in [0,1]^d$ be an input signal.

$$\xrightarrow{\Phi} \quad \Phi(x) = 0.97 \qquad \text{“Monkey”}$$

$$\xrightarrow{\Phi} \quad \Phi(x) = 0.07 \qquad \text{“Not a monkey”}$$

The Relevance Mapping Problem

The Task:

- Determine the most relevant components of x for the prediction $\Phi(x)$.
- Choose $S \subseteq \{1, \ldots, d\}$ of components that are considered relevant.
- S should be small (usually not everything is relevant).
- S^c is considered non-relevant.

Original image x

Relevant components S

Non-relevant components S^c

Rate-Distortion Viewpoint

$\Phi(x) = 0.97$

"Monkey"

Original image x

Partial image S

Random completion y

$\Phi(y) = 0.91$

"Monkey"

Obfuscation: Let

- $n \sim \mathcal{V}$ be a random noise vector, and
- y be a random vector defined as $y_S = x_S$ and $y_{S^c} = n_{S^c}$.

Rate-Distortion Viewpoint

Recall: Let

- $\Phi\colon [0,1]^d \to [0,1]$ be a classification function,
- $x \in [0,1]^d$ be an input signal,
- $n \sim \mathcal{V}$ be a random noise vector, and
- y be a random vector defined as $y_S = x_S$ and $y_{S^c} = n_{S^c}$.

Expected Distortion:

$$D(S) = D(\Phi, x, S) = \mathbb{E}\left[\frac{1}{2}\left(\Phi(x) - \Phi(y)\right)^2\right]$$

Rate-Distortion Function:

$$R(\epsilon) = \min_{S \subseteq \{1,\dots,d\}} \{|S| \,:\, D(S) \leq \epsilon\}$$

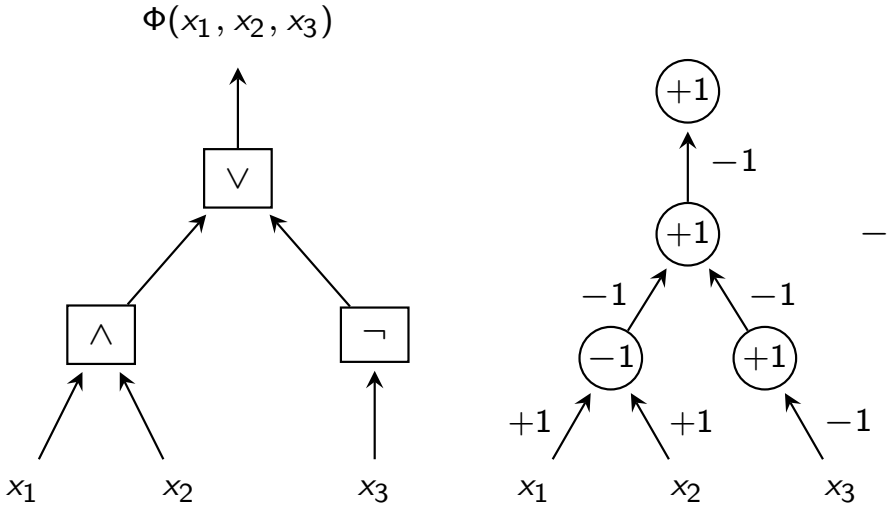
$\rightsquigarrow$ *Use this viewpoint for the definition of a relevance map!*

Finding a minimizer of $R(\epsilon)$

or even approximating it is very hard!

Hardness Results

Boolean Functions as ReLU Neural Networks:

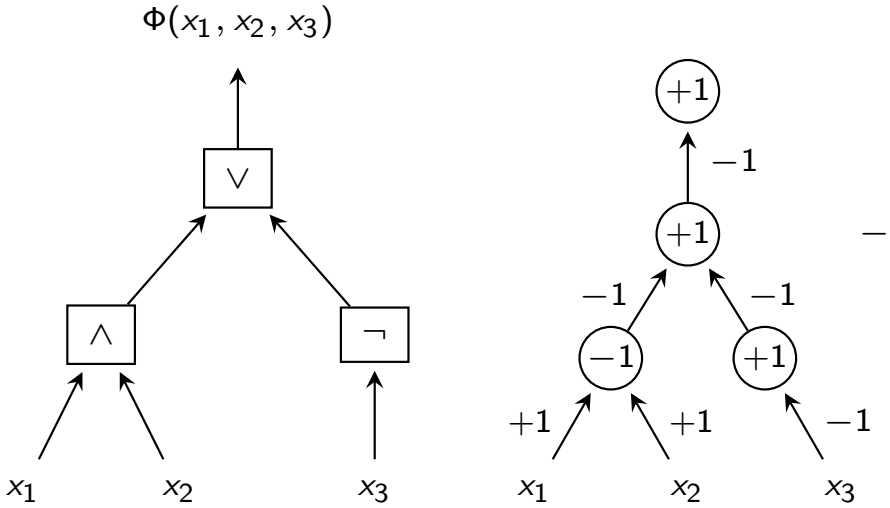


ReLU activation function $\varrho(x) = \max\{0, x\}$

$$-\varrho\left(\begin{bmatrix}-1 & -1\end{bmatrix}\varrho\left(\begin{bmatrix}1 & 1 & 0\\ 0 & 0 & -1\end{bmatrix}\begin{bmatrix}x_1\\ x_2\\ x_3\end{bmatrix}+\begin{bmatrix}-1\\ 1\end{bmatrix}\right)+1\right)+1$$

Hardness Results

Boolean Functions as ReLU Neural Networks:



ReLU activation function $\varrho(x) = \max\{0, x\}$

$$-\varrho\left(\begin{bmatrix}-1 & -1\end{bmatrix}\varrho\left(\begin{bmatrix}1 & 1 & 0\\ 0 & 0 & -1\end{bmatrix}\begin{bmatrix}x_1\\ x_2\\ x_3\end{bmatrix}+\begin{bmatrix}-1\\ 1\end{bmatrix}\right)+1\right)+1$$

The Binary Setting: Let

- $\Phi\colon \{0,1\}^d \to \{0,1\}$ be classifier functions,
- $x \in \{0,1\}^d$ be signals, and
- $\mathcal{V} = \mathcal{U}(\{0,1\}^d)$ be a uniform distribution.

Hardness Results

We consider the binary case.

Theorem (Wäldchen, Macdonald, Hauch, K, 2019):
Given Φ, x, $k \in \{1, \dots, d\}$, and $\epsilon < \frac{1}{4}$. Deciding whether $R(\epsilon) \leq k$ is $\mathrm{NP}^{\mathrm{PP}}$-complete.

Finding a minimizer of $R(\epsilon)$ is hard!

Theorem (Wäldchen, Macdonald, Hauch, K, 2019):
Given Φ, x, and $\alpha \in (0, 1)$. Approximating $R(\epsilon)$ to within a factor of $d^{1-\alpha}$ is NP-hard.

Even the approximation problem of it is hard!

Preprint: "The Computational Complexity of Understanding Network Decisions", https://arxiv.org/abs/1905.09163

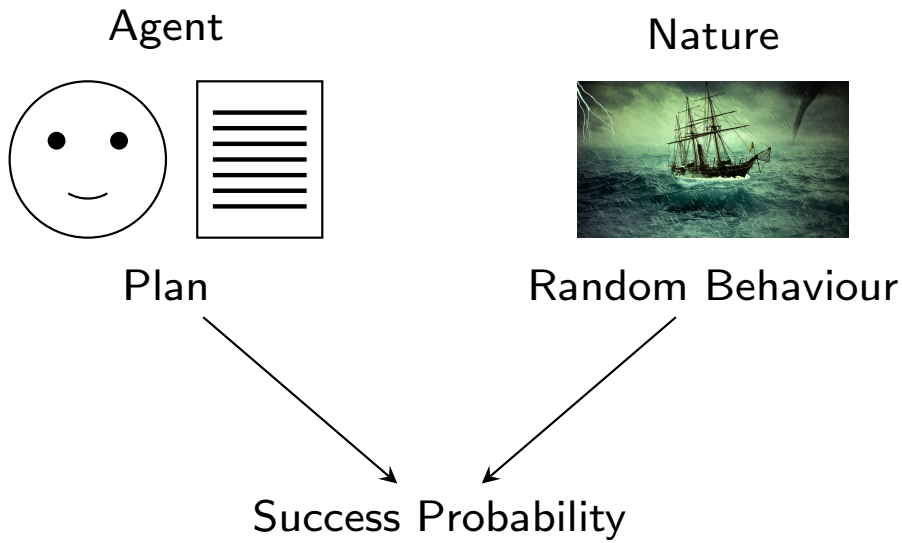
What is NP^{PP}?

The Complexity Class NP^{PP}:

Many important problems in artificial intelligence belong to this class.

Some Examples:

- Planning under uncertainties
- Finding maximum a-posteriori configurations in graphical models
- Maximizing utility functions in Bayesian networks



Our Method:

Rate-Distortion Explanation (RDE)

Problem Relaxation

	Discrete problem	Continuous problem
Relevant set	$S \subseteq \{1, \dots, d\}$	
Obfuscation	$y_S = x_S,\ y_{S^c} = n_{S^c}$	
Distortion	$D(S)$	
Rate/Size	$\lvert S \rvert$	

Problem Relaxation

	Discrete problem	Continuous problem
Relevant set	$S \subseteq \{1, \dots, d\}$	$s \in [0,1]^d$
Obfuscation	$y_S = x_S,\ y_{S^c} = n_{S^c}$	$y = s \odot x + (1-s) \odot n$
Distortion	$D(S)$	$D(s)$
Rate/Size	$\lvert S \rvert$	$\lVert s \rVert_1$

Problem Relaxation

	Discrete problem	Continuous problem
Relevant set	$S \subseteq \{1, \dots, d\}$	$s \in [0,1]^d$
Obfuscation	$y_S = x_S,\ y_{S^c} = n_{S^c}$	$y = s \odot x + (1-s) \odot n$
Distortion	$D(S)$	$D(s)$
Rate/Size	$\lvert S \rvert$	$\lVert s \rVert_1$

Resulting Minimization Problem:

$$\text{minimize} \quad D(s) + \lambda \|s\|_1 \quad \text{subject to} \quad s \in [0,1]^d$$

Observations

Distortion:

$$D(s) = \mathbb{E}\left[\frac{1}{2}\left(\Phi(x) - \Phi(y)\right)^2\right]$$
$$= \frac{1}{2}\left(\Phi(x) - \mathbb{E}\left[\Phi(y)\right]\right)^2 + \frac{1}{2}\operatorname{cov}\left[\Phi(y)\right]$$

Obfuscation:

$$\mathbb{E}\left[y\right] = s \odot x + (1 - s) \odot \mathbb{E}\left[n\right]$$
$$\operatorname{cov}\left[y\right] = \operatorname{diag}(1 - s)\operatorname{cov}\left[n\right]\operatorname{diag}(1 - s)$$

Observations

$$\mathbb{E}[y], \operatorname{cov}[y] \xrightarrow{\Phi} \mathbb{E}[\Phi(y)], \operatorname{cov}[\Phi(y)]$$

Observations

$$\mathbb{E}[y], \operatorname{cov}[y] \xrightarrow{\Phi} \mathbb{E}[\Phi(y)], \operatorname{cov}[\Phi(y)]$$

Generic Approach:

- Estimate using sample mean and sample covariance
- Possible for any classifier function Φ
- Might require large number of samples

Observations

$$\mathbb{E}[y], \operatorname{cov}[y] \xrightarrow{\Phi} \mathbb{E}[\Phi(y)], \operatorname{cov}[\Phi(y)]$$

Generic Approach:

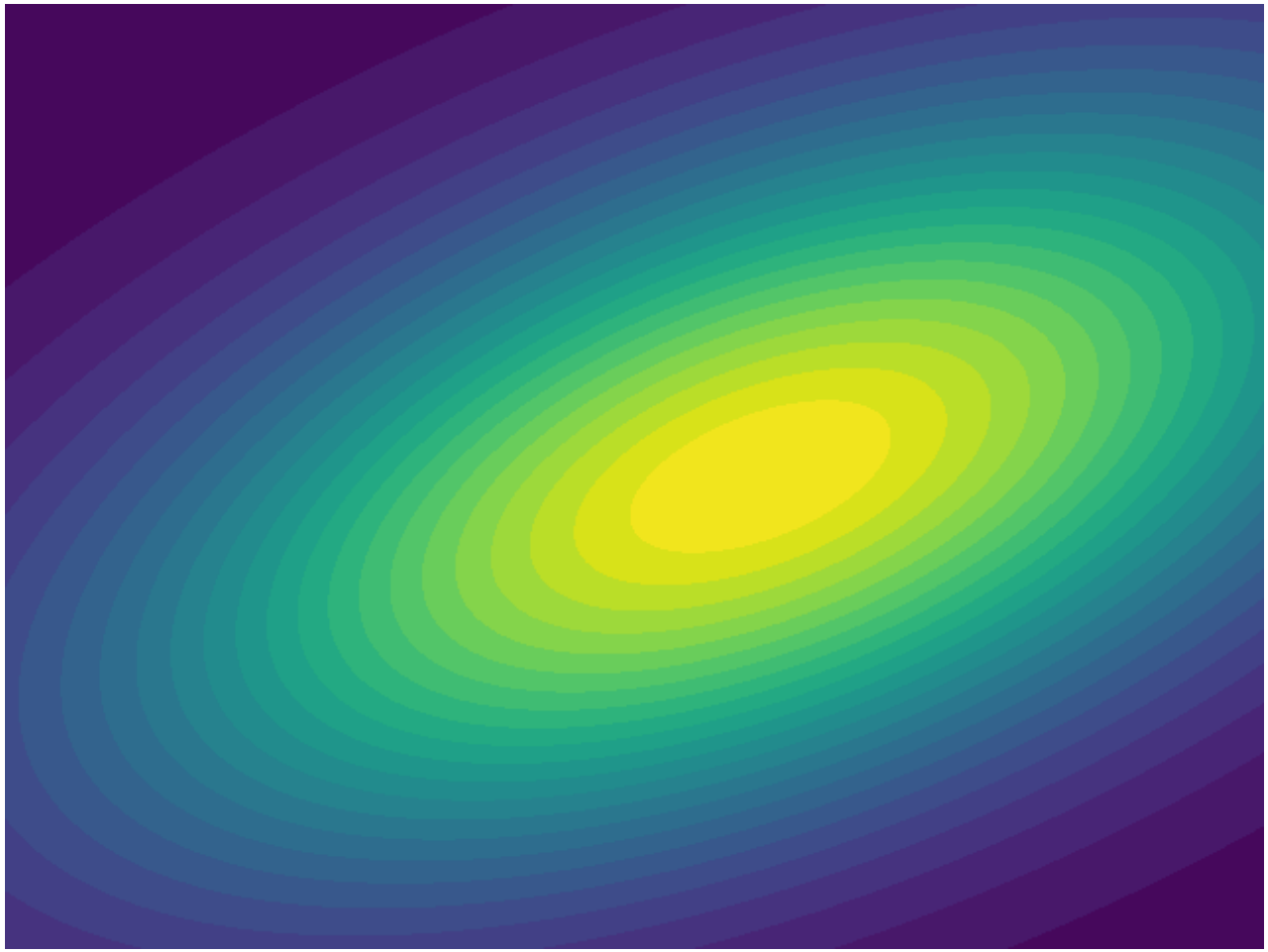
- Estimate using sample mean and sample covariance
- Possible for any classifier function Φ
- Might require large number of samples

Neural Network Approach:

- Use compositional structure of Φ
- Propagate distribution through the layers
- Project to simple family of distributions at each layer

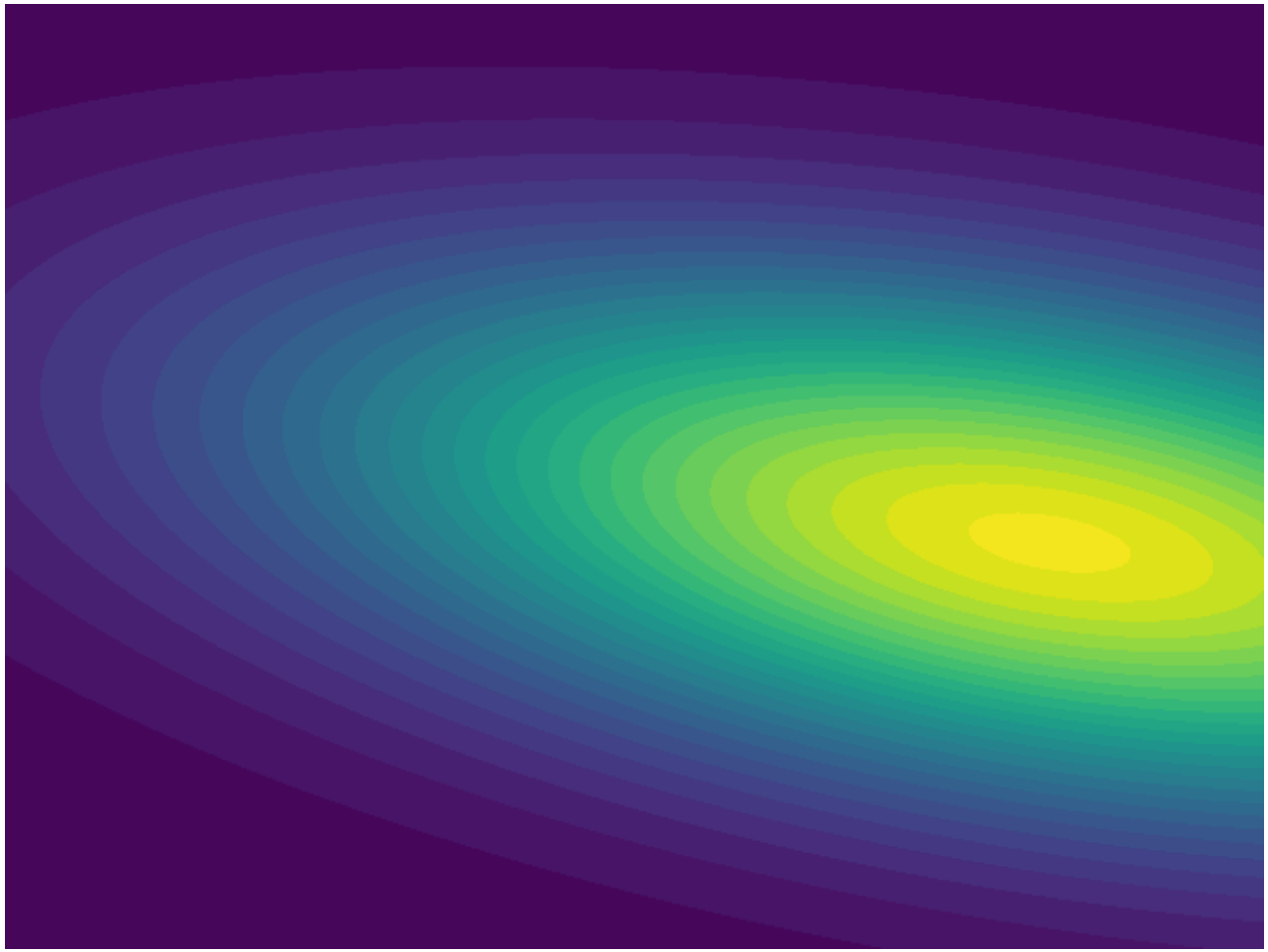
Assumed Density Filtering

Input distribution: $\mathcal{N}(\mu_{\text{in}}, \sigma_{\text{in}})$



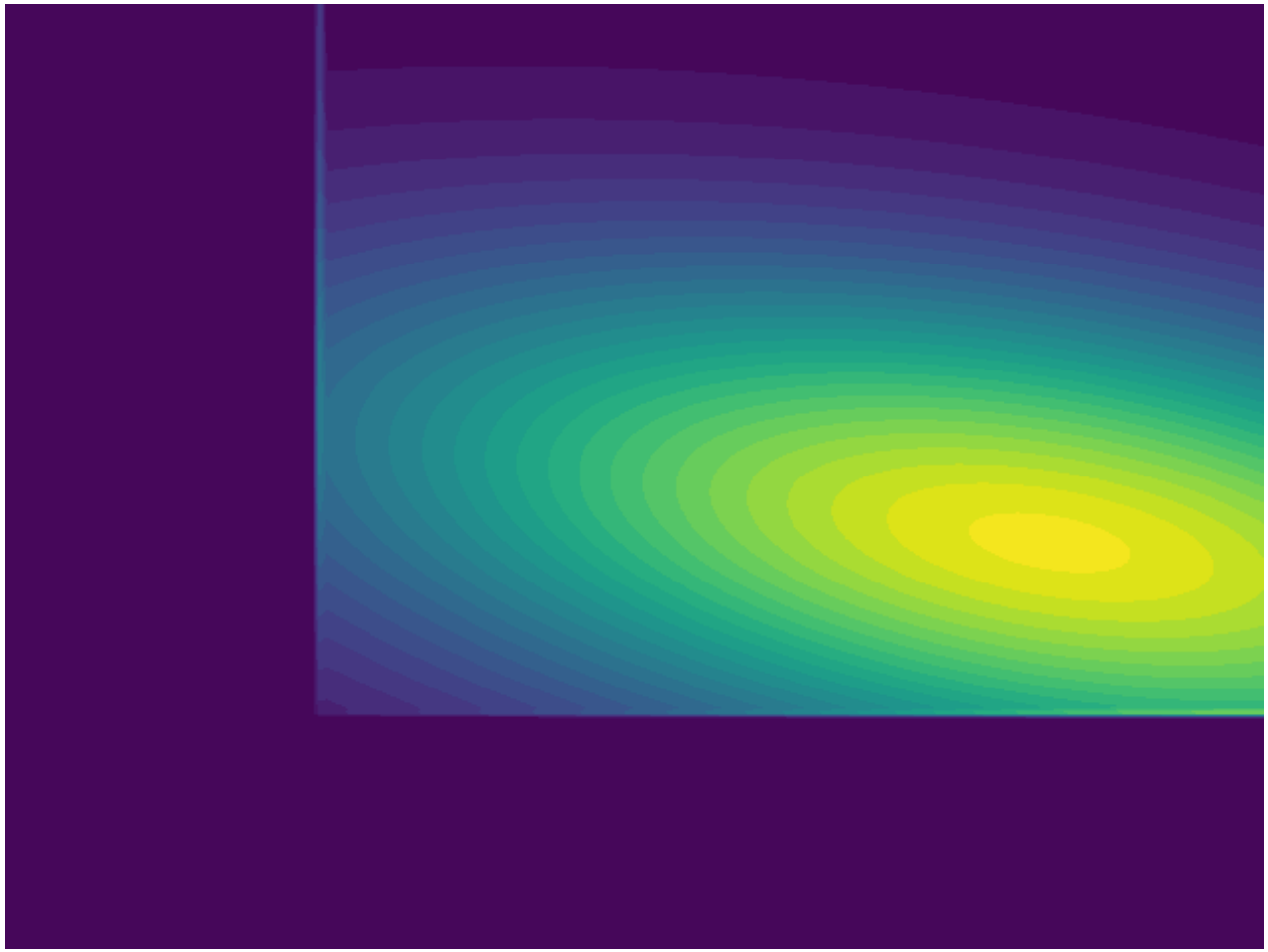
Assumed Density Filtering

Affine transform: $\mathcal{N}(W\mu_{\text{in}} + b, W\sigma_{\text{in}}W^\top)$



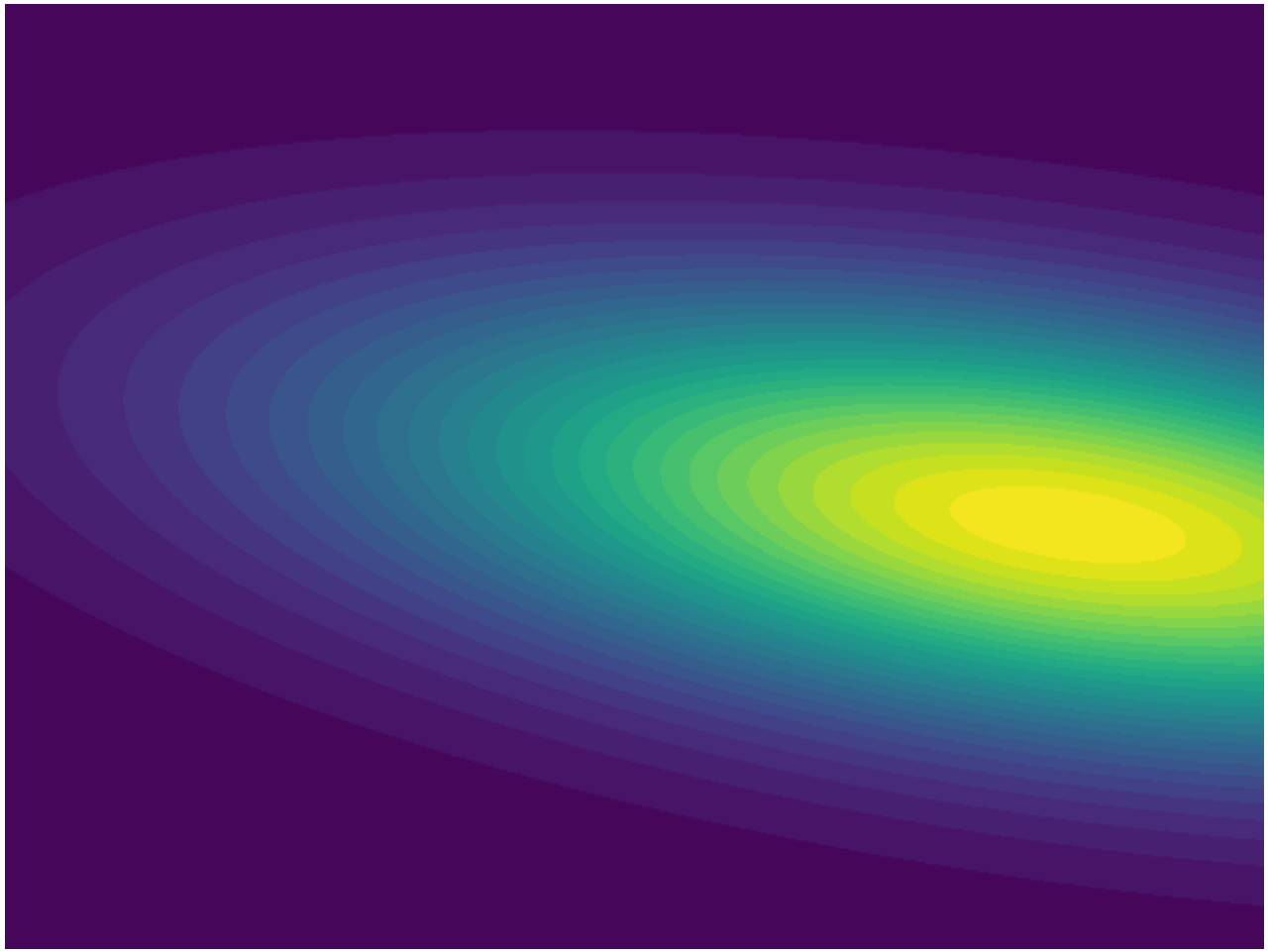
Assumed Density Filtering

ReLU activation: Not Gaussian anymore



Assumed Density Filtering

Moment matching output distribution: $\mathcal{N}(\mu_{\text{out}}, \sigma_{\text{out}})$

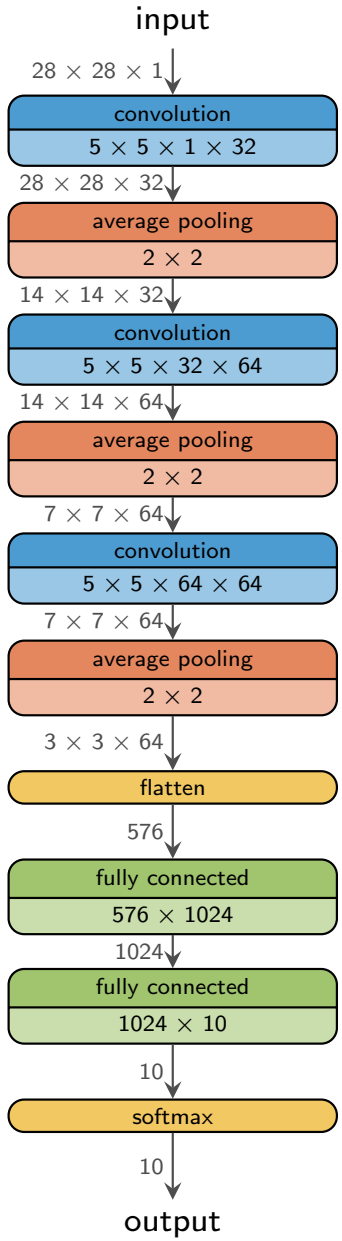


Numerical Experiments

MNIST Experiment

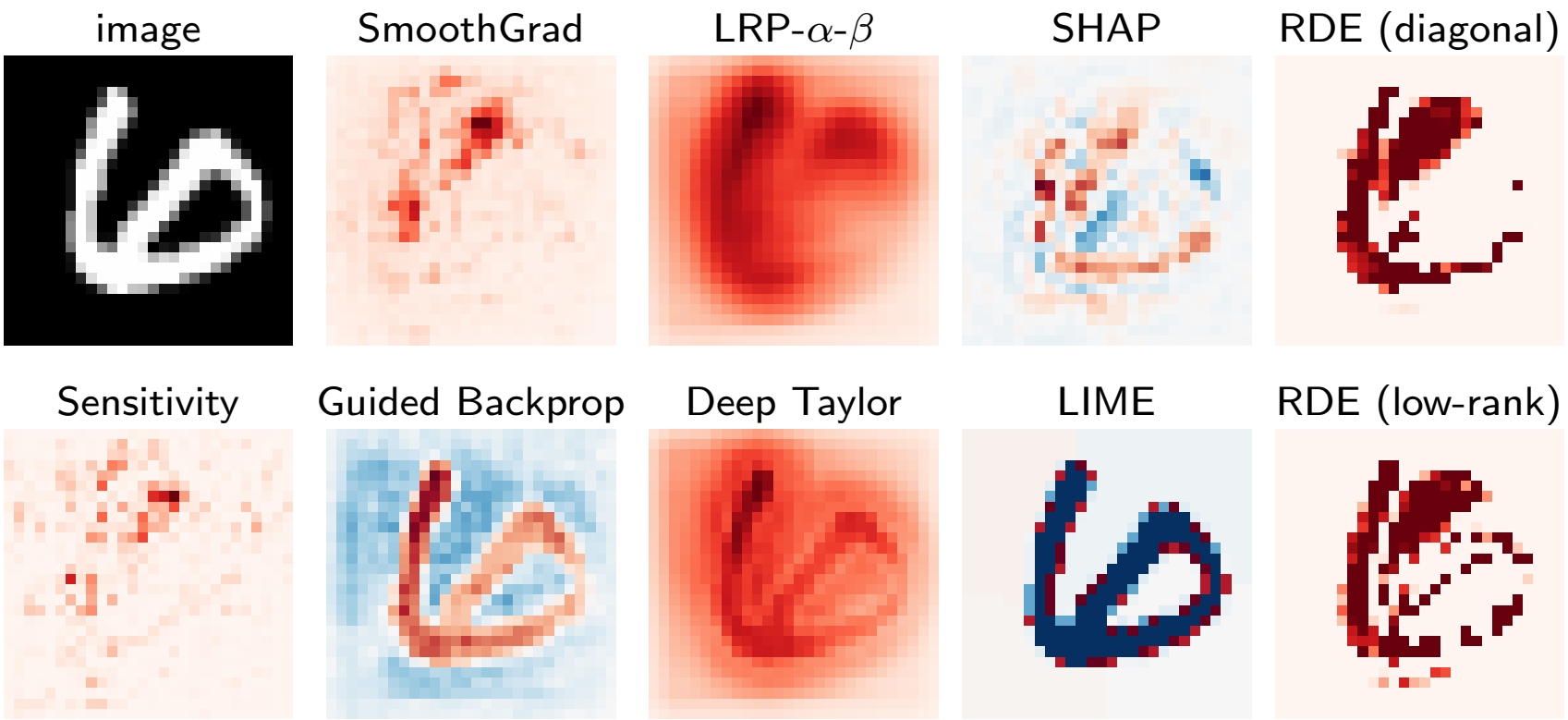
Data Set	
Image size	$28 \times 28 \times 1$
Number of classes	10
Training samples	50000

Test accuracy: 99.1%



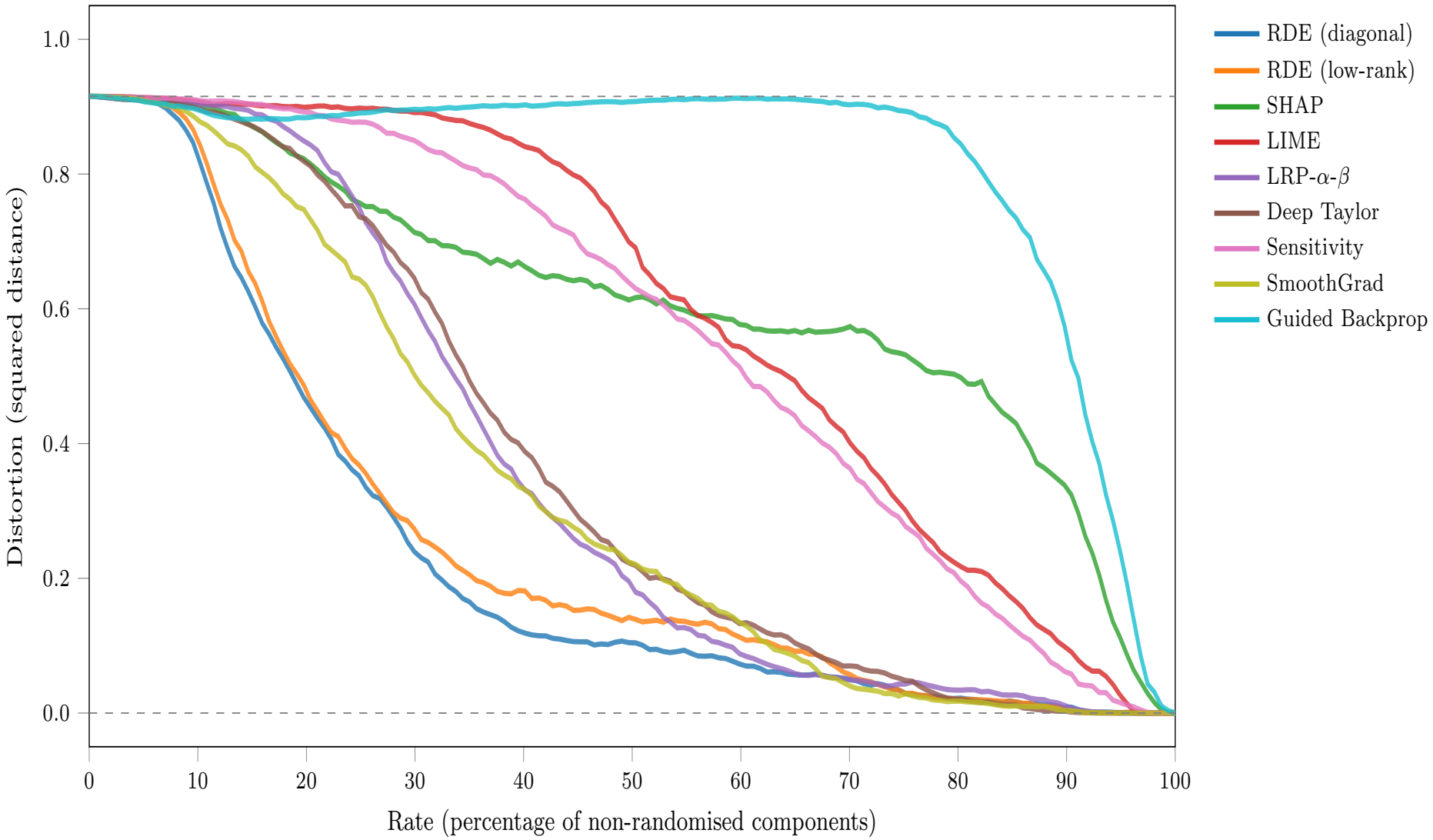
MNIST dataset of handwritten digits (LeCun, Cortes, 1998)

MNIST Experiment



SmoothGrad (Smilkov, Thorat, Kim, Viégas, Wattenberg, 2017), Layer-wise Relevance Propagation (Bach, Binder, Montavon, Klauschen, Müller, Samek, 2015), SHAP (Lundberg, Lee, 2017), Sensitivity Analysis (Simonyan, Vedaldi, Zisserman, 2013), Guided Backprop (Springenberg, Dosovitskiy, Brox, Riedmiller, 2015), Deep Taylor Decompositions (Montavon, Samek, Müller, 2018), LIME (Ribeiro, Singh, Guestrin, 2016)

MNIST Experiment



SmoothGrad (Smilkov, Thorat, Kim, Viégas, Wattenberg, 2017), Layer-wise Relevance Propagation (Bach, Binder, Montavon, Klauschen, Müller, Samek, 2015), SHAP (Lundberg, Lee, 2017), Sensitivity Analysis (Simonyan, Vedaldi, Zisserman, 2013), Guided Backprop (Springenberg, Dosovitskiy, Brox, Riedmiller, 2015), Deep Taylor Decompositions (Montavon, Samek, Müller, 2018), LIME (Ribeiro, Singh, Guestrin, 2016)

STL-10 Experiment

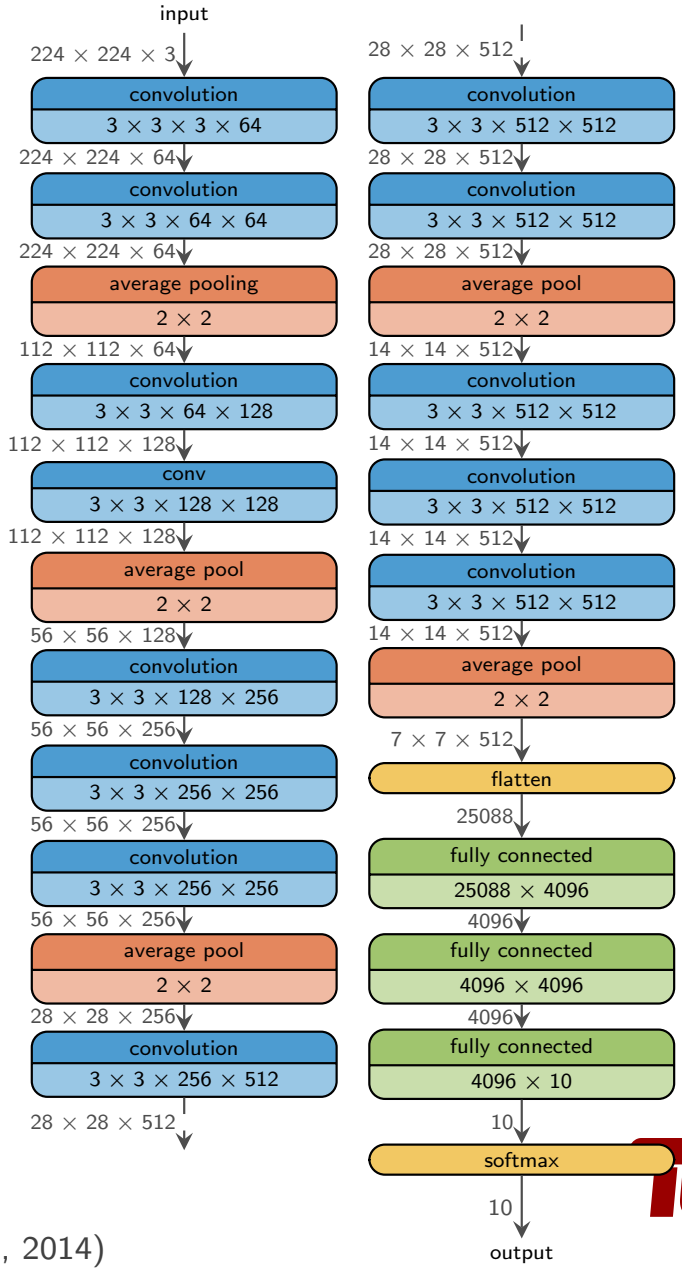



Data Set

Image size	$96 \times 96 \times 3$
	$(224 \times 224 \times 3)$
Number of classes	10
Training samples	4000

Test accuracy: 93.5%

(VGG-16 convolutions pretrained on Imagenet)

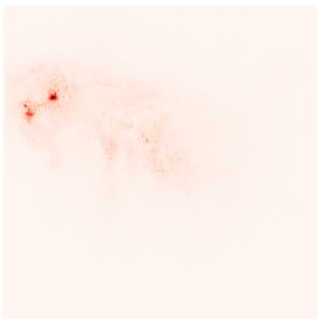


STL-10 dataset (Coates, Lee, Ng, 2011), VGG-16 network (Simonyan, Zisserman, 2014)

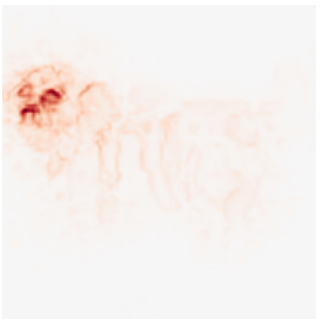
STL-10 Experiment

image

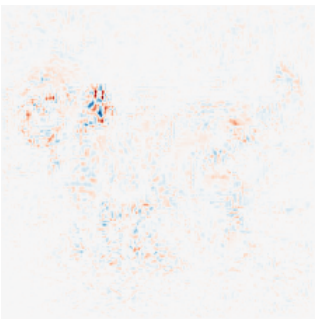
SmoothGrad



LRP-α-β

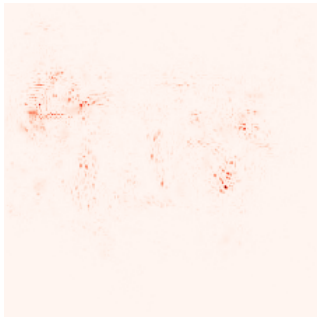


SHAP

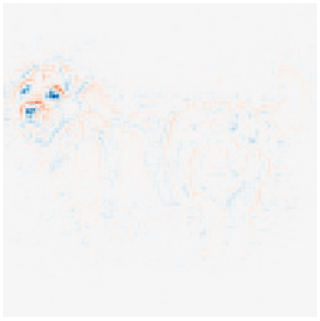


RDE (diagonal)

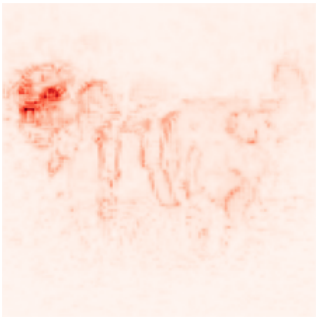
Sensitivity



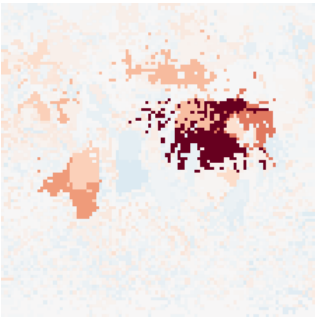
Guided Backprop



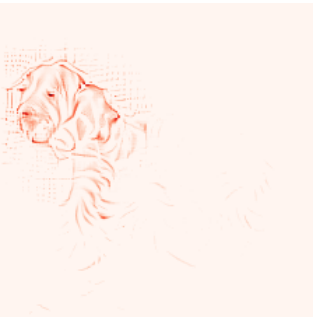
Deep Taylor



LIME

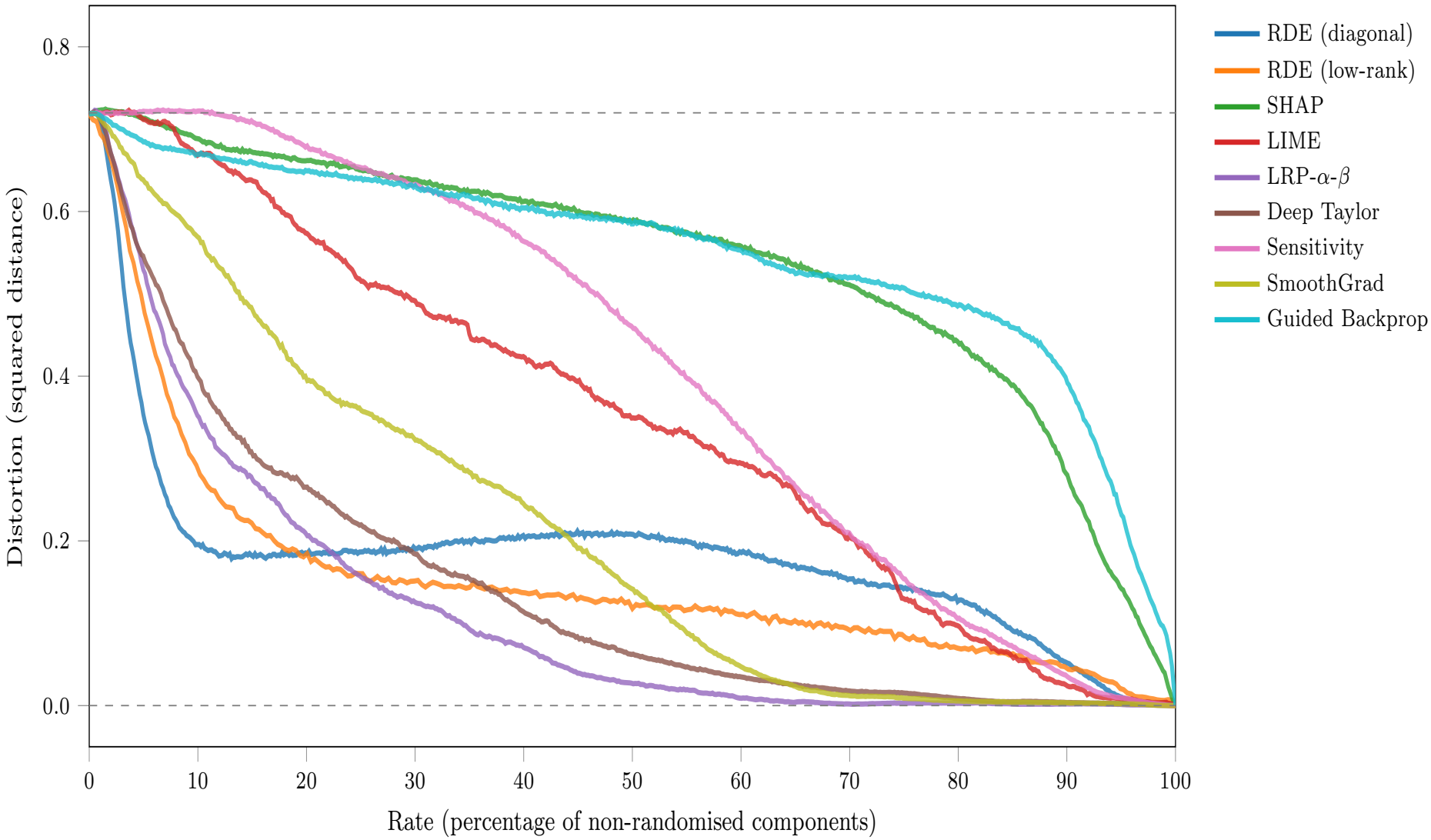


RDE (low-rank)



SmoothGrad (Smilkov, Thorat, Kim, Viégas, Wattenberg, 2017), Layer-wise Relevance Propagation (Bach, Binder, Montavon, Klauschen, Müller, Samek, 2015), SHAP (Lundberg, Lee, 2017), Sensitivity Analysis (Simonyan, Vedaldi, Zisserman, 2013), Guided Backprop (Springenberg, Dosovitskiy, Brox, Riedmiller, 2015), Deep Taylor Decompositions (Montavon, Samek, Müller, 2018), LIME (Ribeiro, Singh, Guestrin, 2016)

STL-10 Experiment



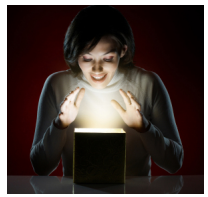
SmoothGrad (Smilkov, Thorat, Kim, Viégas, Wattenberg, 2017), Layer-wise Relevance Propagation (Bach, Binder, Montavon, Klauschen, Müller, Samek, 2015), SHAP (Lundberg, Lee, 2017), Sensitivity Analysis (Simonyan, Vedaldi, Zisserman, 2013), Guided Backprop (Springenberg, Dosovitskiy, Brox, Riedmiller, 2015), Deep Taylor Decompositions (Montavon, Samek, Müller, 2018), LIME (Ribeiro, Singh, Guestrin, 2016)

Conclusions

What to take Home...?

Deep Learning:

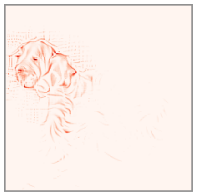
- Impressive performance *in combination with classical mathematical methods* (Inverse Problems, PDEs, ...).
- A theoretical foundation of neural networks is largely missing: *Expressivity, Learning, Generalization, and Interpretability*.



Interpretability:

- Opening the *black box* of a trained neural network.
- Determining which input features are *most relevant* for a decision.

Rate-Distortion Explanation (RDE):



- Determining the *minimal rate* is *hard*.
- Even the *approximation problem* of it is *hard*.
- *RDE considers a relaxed version* within the *rate-distortion framework*.
- It *outperforms current methods* for smaller rates.

THANK YOU!

References available at:

www.math.tu-berlin.de/∼kutyniok

Related Book:

- P. Grohs and G. Kutyniok
  Theory of Deep Learning
  Cambridge University Press (in preparation)