Dataninja Spring School

Explainable AI

Grégoire Montavon, TU Berlin

March 23 2022

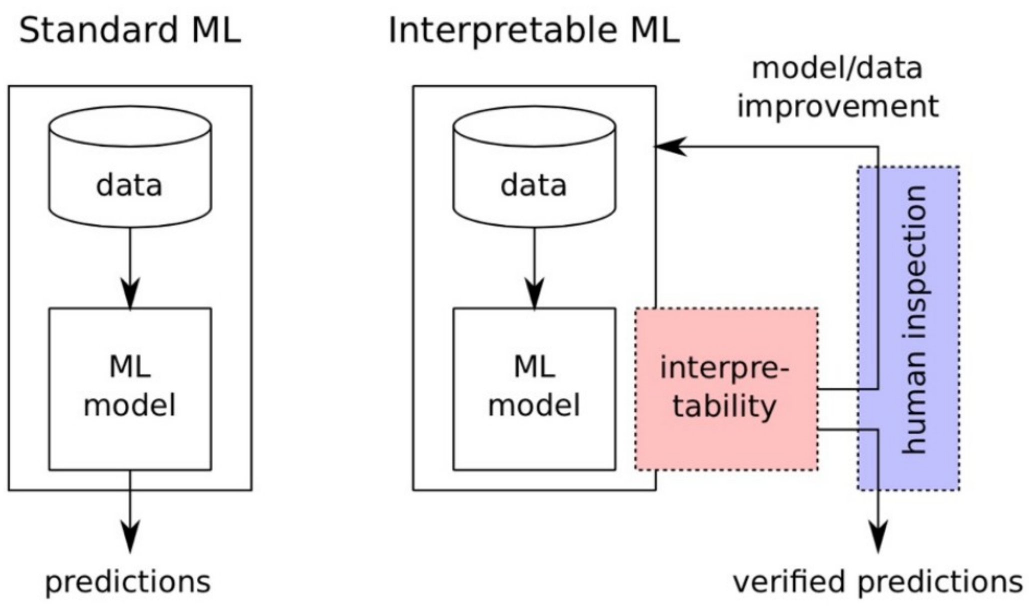
Tutorial Structure

- Part 1: Introduction
- Part 2: Methods of Explainable AI (XAI)
- Part 3: Extensions and Applications

Part 1: Introduction

- Components of XAI (model, explanation, user)
- Practical motivations
- Desiderata of an explanation system
- Types of explanations

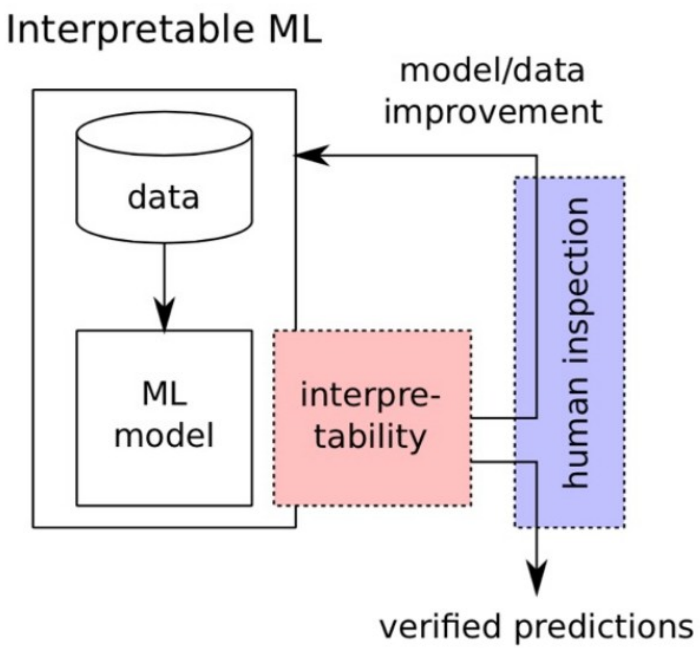
Explainable AI System



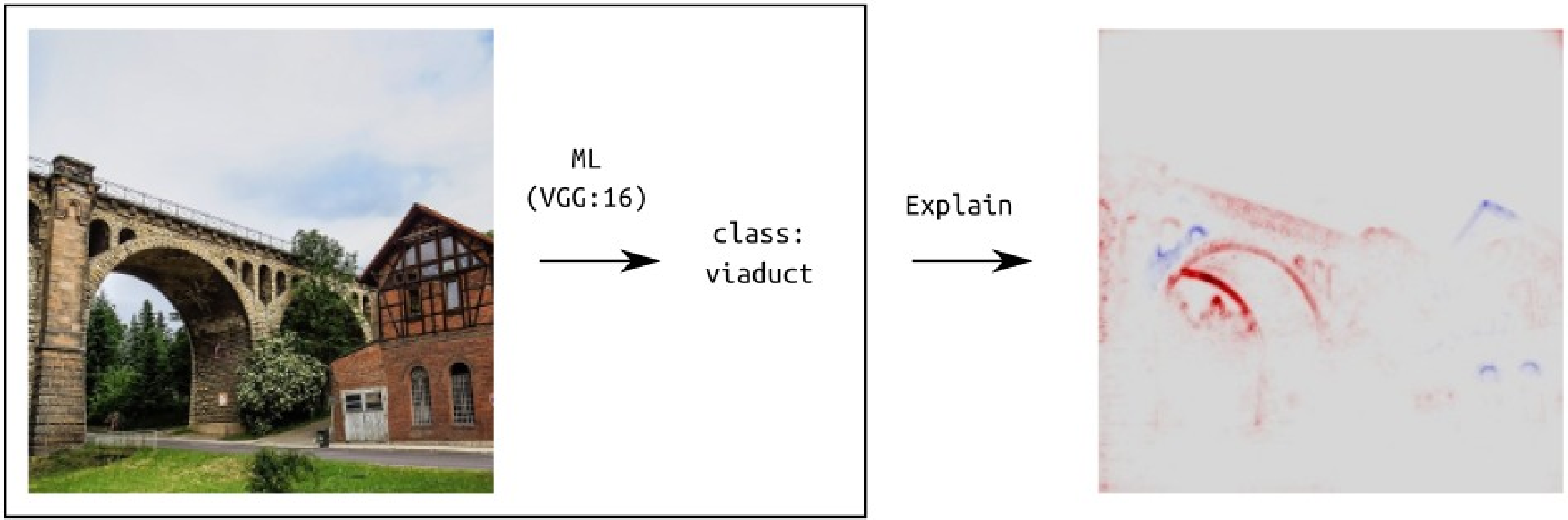
Goals: Expose the decision strategy of the ML model to the user, in order to get insights from the model, confront the explanation with the user's own domain knowledge, and possibly correct model flaws.

Components of an XAI System

- **The ML model**
  - Generalizes user knowledge (in the form of human labeling) to new data points. Compared to the human, a ML model is faster, less costly, and sometimes more accurate.
- **The explanation**
  - Transformation of the prediction strategy implemented by the ML model into something informative and intelligible for the human.
- **The user**
  - User possesses expert knowledge, that is sometimes not integrated in the model (due to small dataset, or flawed training).



Example of an Explanation



Pixels that are relevant for the model to classify the input image into a particular class (here viaduct) are highlighted in red.

Note: Explanation reveals the decision strategy of the model, not necessarily the actual object in the image.

Why Explainable AI? Practical Motivations

- **Trustworthy AI**
  - XAI is used to further validate the learned ML model (in order to verify that it implements the correct decision strategy and generalizes well).
- **Generating Scientific Insights**
  - XAI is used in combination with ML to identify the relation between different variables in some complex system of scientific interest (e.g. a molecular system or a biological cell).
- **Compliant AI**
  - Explanations of AI decision (and valid explanation) is required (e.g. by law) to deploy an AI system and let the AI system take decisions.
- **Actionable AI**
  - XAI is used in combination with ML to characterize the input-output behavior of a complex system so that the latter can be actioned in a meaningful manner.

Motivations: XAI for Trust

Pascal VOC 2007 dataset: Fisher Vector Classifier vs. DeepNet pretrained on ImageNet

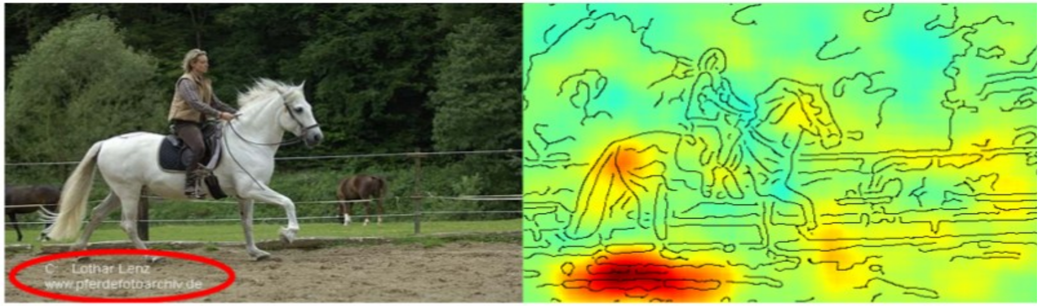
	aeroplane	bicycle	bird	boat	bottle	bus	car
Fisher	79.08%	66.44%	45.90%	70.88%	27.64%	69.67%	80.96%
DeepNet	88.08%	79.69%	80.77%	77.20%	35.48%	72.71%	86.30%
	cat	**chair**	**cow**	**diningtable**	**dog**	**horse**	**motorbike**
Fisher	59.92%	51.92%	47.60%	58.06%	42.28%	80.45%	69.34%
DeepNet	81.10%	51.04%	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%

Motivations: XAI for Trust

Pascal VOC 2007 dataset: Fisher Vector Classifier vs. DeepNet pretrained on ImageNet

	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%

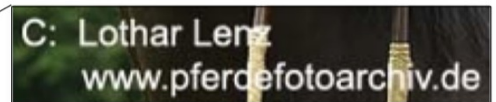
Fisher classifier



Lapuschkin et al. 2016. Analyzing Classifiers: Fisher Vectors and Deep Neural Networks

Motivations: XAI for Trust

‘horse’ images in PASCAL VOC 2007



Motivations: XAI for Trust

Because the classifier relies on a non-informative feature (the copyright tag), it can be easily fooled.

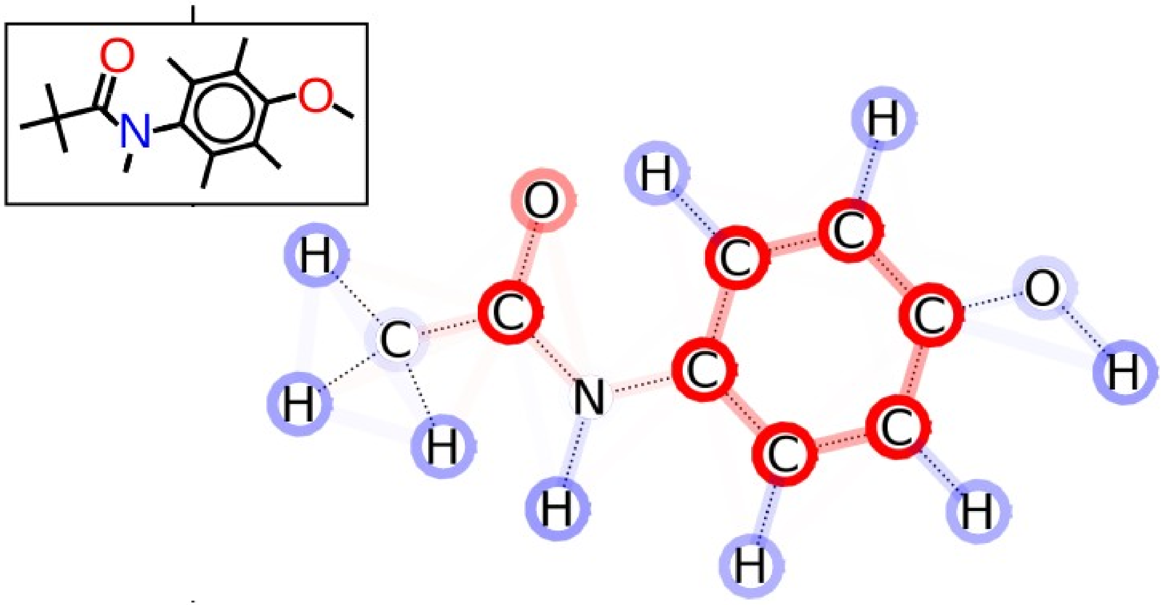
Examples:

→ NOT HORSE

→ HORSE

Clever Hans models are unlikely to perform well on **future data**.

Motivations: XAI for Scientific Insights



Example: What atoms or regions of the molecule contribute most strongly to the atomization energy of a molecule.

Motivations: XAI for Compliant AI

- Art 13. GDPR (excerpt)

  ... In addition to the information referred to in paragraph 1, the controller shall, at the time when personal data are obtained, provide the data subject with the following further information necessary to ensure fair and transparent processing:

  - [...]
  - the existence of automated decision-making, including profiling, referred to in Article 22(1) and (4) and, at least in those cases, meaningful information about the logic involved, as well as the significance and the envisaged consequences of such processing for the data subject.
  - [...]

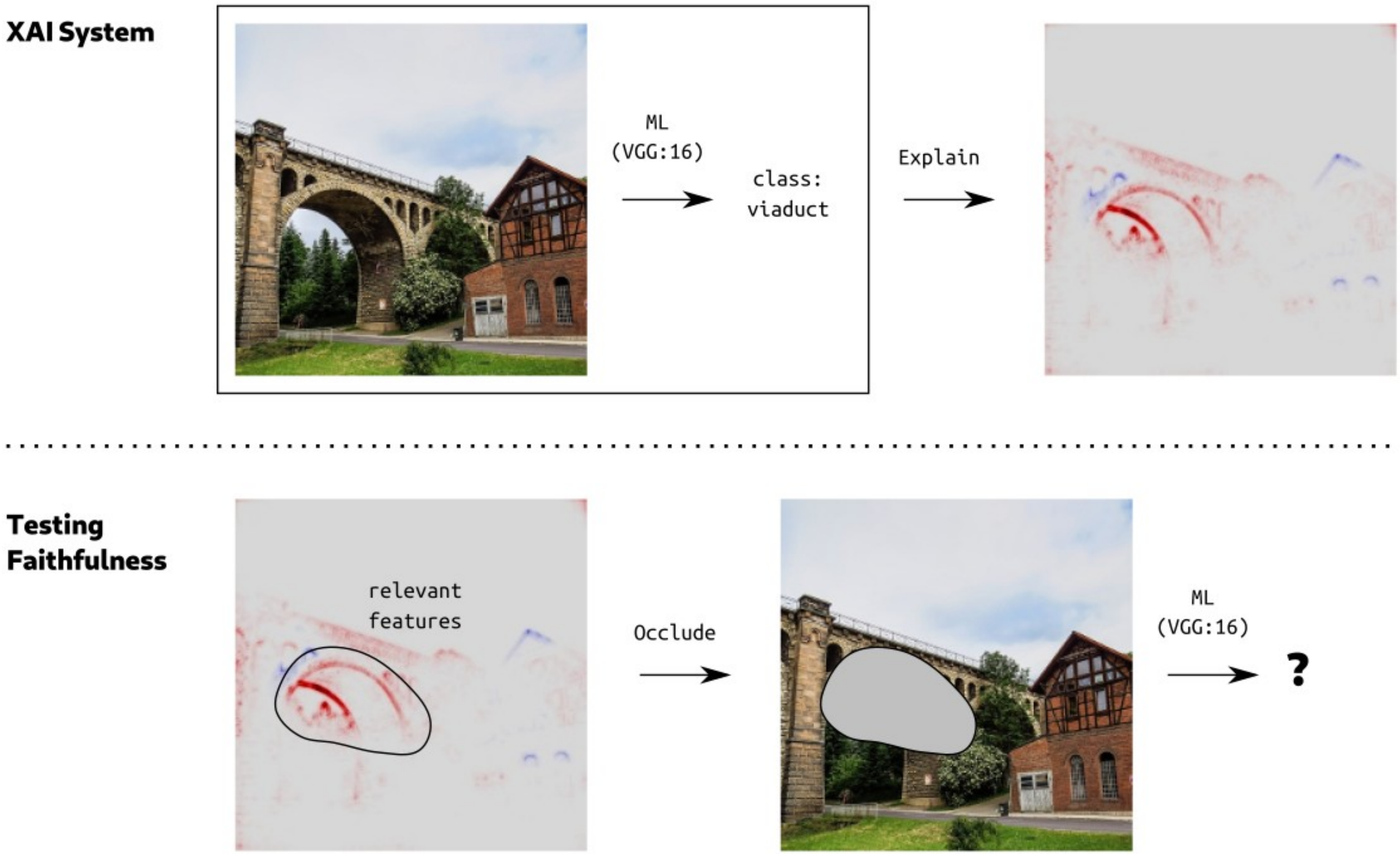
- **Question:** Are XAI outputs 'compatible' with what is required by law?

Desiderata of an Explanation System

1. **Fidelity:** The explanation should reflect the quantity being explained and not something else.
2. **Understandability:** The explanation must be easily understandable by its receiver.
3. **Sufficiency:** The explanation should provide sufficient information on how the model came up with its prediction.
4. **Low Overhead:** The explanation should not cause the prediction model to become less accurate or less efficient.
5. **Runtime Efficiency:** Explanations should be computable in reasonable time.

(cf. Swartout & Moore 1993 [13])

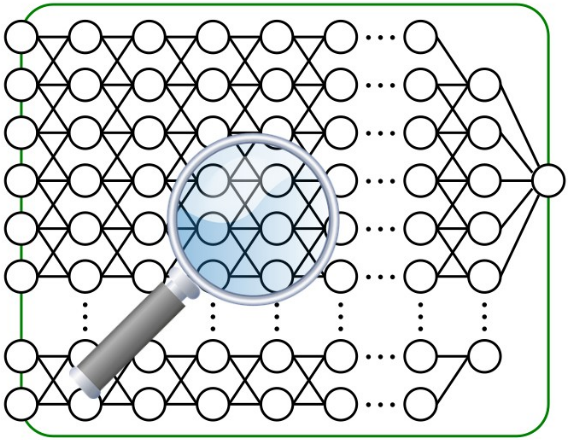
Desiderata: Fidelity (Faithfulness)

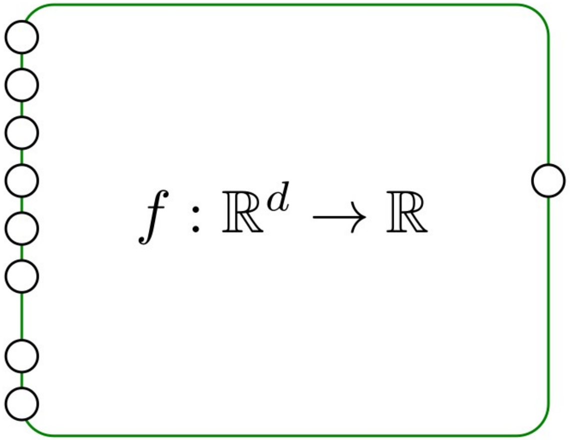


Types of Explanation

- **Mechanistic vs. Functional**
  - What do we want to explain about the model: how it is designed, or how it behaves?
- **Feature Set or Feature Scoring**
  - Are we interested in extracting a set of relevant features, or finding the exact contribution of each feature?
- **Local vs. Global**
  - Are we interested in explaining a particular prediction (e.g. for a given image), or the behavior of the model on a whole dataset / input domain?

Types of Explanations: Mechanistic vs. Functional

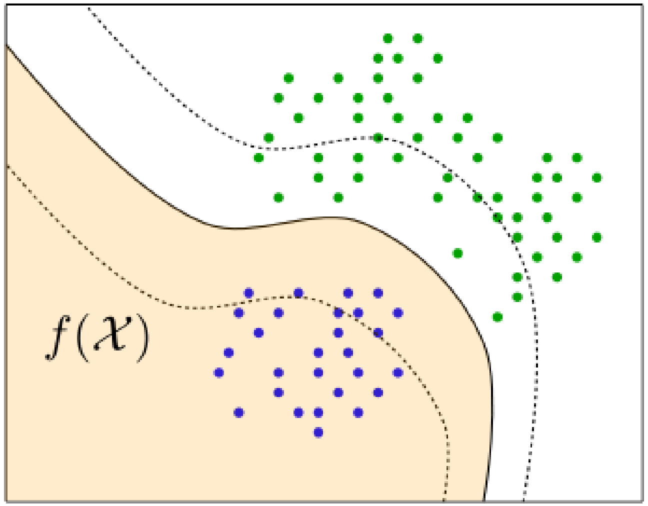




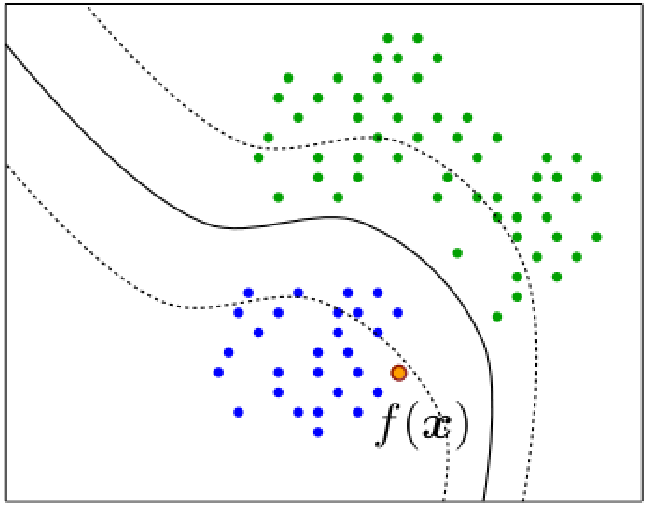
Mechanistic: Understanding what mechanism the network uses to solve a problem or implement a function.

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

Types of Explanations: Local vs. Global

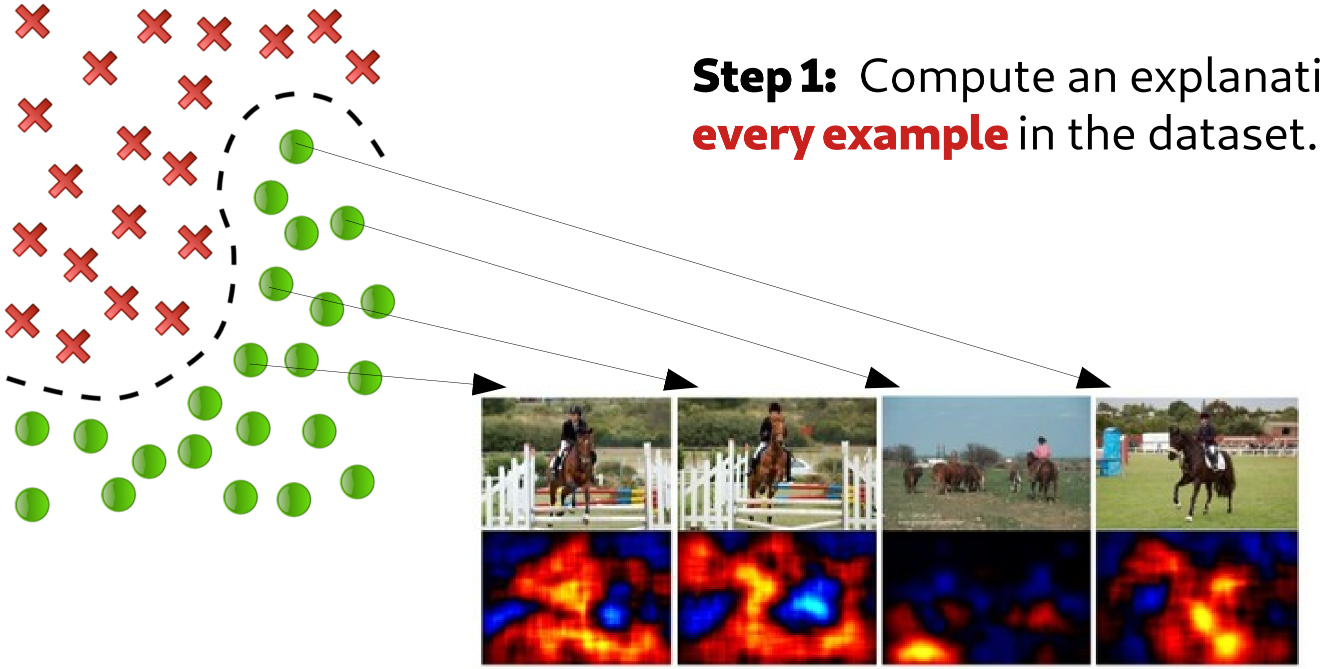


Global: What features are relevant in order to produce a positive response f(x) in general.



Local: What features are relevant for a given data point.

Types of Explanations: From Local to Global

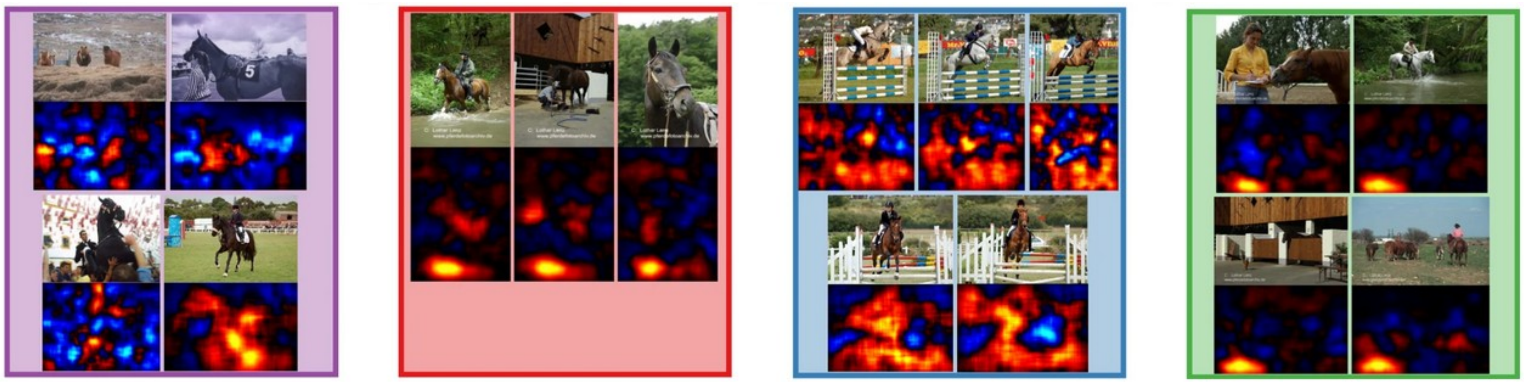


Step 1: Compute an explanation for **every example** in the dataset.

Lapuschkin et al. (2019)
Unmasking Clever Hans Predictors and Assessing What Machines Really Learn

Types of Explanations: From Local to Global

Step 2: Organize explanations into **clusters**.



Clever Hans effects are now obtained **systematically**.

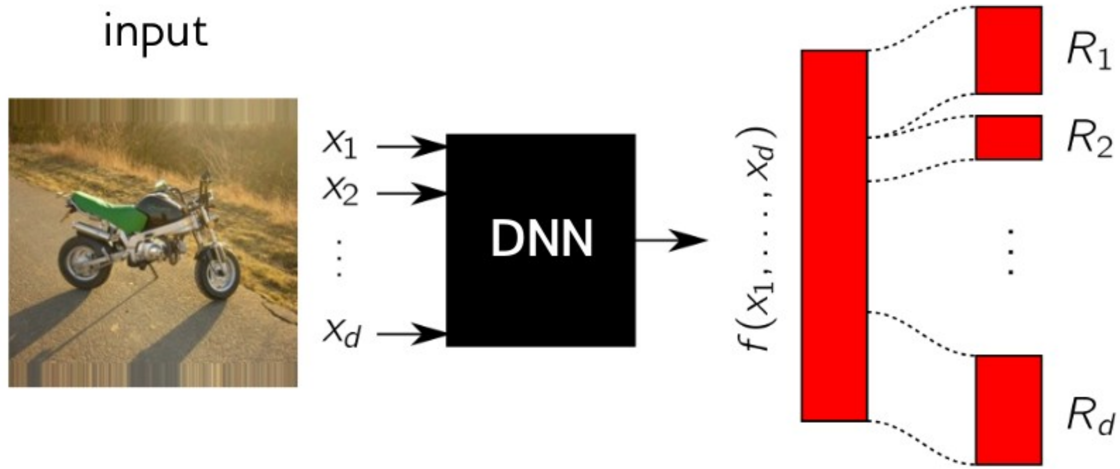
Lapuschkin et al. (2019)
Unmasking Clever Hans Predictors and Assessing What Machines Really Learn

Part 2: Methods

- The problem of attribution
- XAI methods for attribution
  - Shapley Value
  - Gradient x Input (GI)
  - Layer-wise Relevance Propagation (LRP)
- Theoretical properties

The Problem of Attribution

Attribution: Determining the contribution of each input features to the score predicted at the output of the model, e.g. what percentage of the function output is explained by a particular input feature.



Decomposition property: $f(x_1, \ldots, x_d) = \sum_{i=1}^{d} R_i$

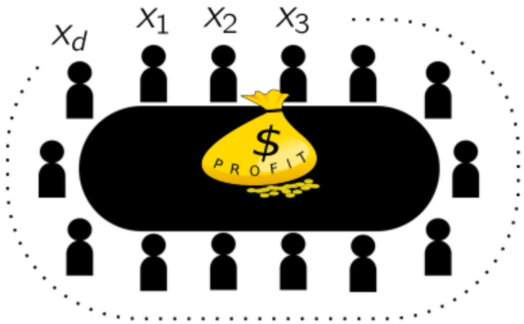
Categories of Attribution Methods

- Perturbation-Based
  - Shapley Value
  - Occlusion
- Gradient-Based
  - Gradient x Input (GI)
  - SmoothGrad
  - Integrated Gradients
- Propagation-Based
  - Guided Backprop
  - Layer-wise Relevance Propagation (LRP)
- Additive Surrogates Models

Shapley Values

- Framework originally proposed in the context of game theory (Shapley 1951) for assigning payoffs in a cooperative game, and recently applied to ML models.

- Each input variable is viewed as a player, and the function output as the profit realized by the cooperating players.



The Shapley values $\phi_1, \ldots, \phi_d$ measuring the contribution of each feature are:

$$\phi_i = \sum_{\mathcal{S}:\, i \notin \mathcal{S}} \frac{|\mathcal{S}|!(d-|\mathcal{S}|-1)!}{d!} \big[f(\boldsymbol{x}_{\mathcal{S}\cup\{i\}}) - f(\boldsymbol{x}_{\mathcal{S}}) \big]$$

where $(\boldsymbol{x}_{\mathcal{S}})_{\mathcal{S}}$ are all possible subsets of features contained in the input $\boldsymbol{x}$.

Shapley Values

Recall: $$\phi_i = \sum_{\mathcal{S}:\, i \notin \mathcal{S}} \underbrace{\frac{|\mathcal{S}|!(d-|\mathcal{S}|-1)!}{d!}}_{\alpha_{\mathcal{S}}} \underbrace{\left[f(\boldsymbol{x}_{\mathcal{S}\cup\{i\}}) - f(\boldsymbol{x}_{\mathcal{S}})\right]}_{\Delta_{\mathcal{S}}}$$

Worked-through example: Consider the function $f(\boldsymbol{x}) = x_1 \cdot (x_2 + x_3)$. Calculate the contribution of each feature to the prediction $f(\mathbf{1}) = 1 \cdot (1+1) = 2$.

Gradient x Input

A feature is contributing to the prediction if (1) the model is sensitive to it and (2) the feature is activated. The Gradient $\times$ Input method [1]:

$$\phi_i = [\nabla f(\boldsymbol{x})]_i \cdot x_i$$

implements this idea and it can be computed quickly in one forward/backward pass.

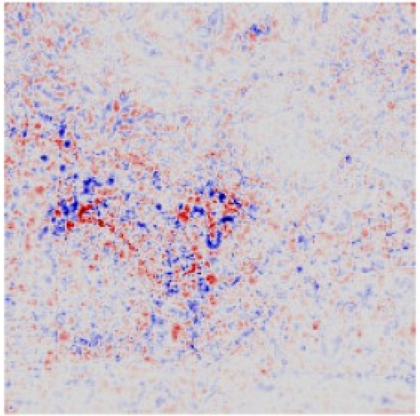
Proposition: *When f is a deep ReLU network (without bias), i.e. when*

$$f(\boldsymbol{x}) = \rho(W_L \rho(\dots \rho(W_1 \boldsymbol{x})))$$

then, Gradient $\times$ Input satisfies $\sum_i \phi_i = f(\boldsymbol{x})$.

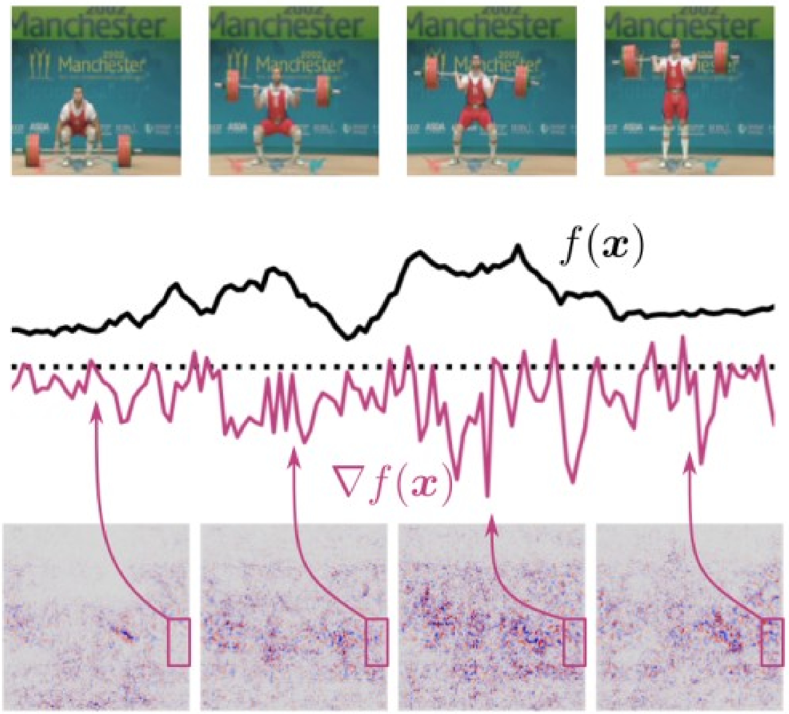
Gradient x Input in Practice

Example: Gradient $\times$ Input explanation of the VGG-16 neural network output neuron 'viaduct' for a given input image:



Observation: There is an exceedingly large amount of positive (red) and negative (blue) scores. Explanations also appear noisy and are hard to interpret.

Problem: Gradients are 'Shattered'



- We look at the DNN output (and its gradient) along some trajectory in the input space, e.g. an athlete lifting a barebell.
- The function is relatively stable, but the gradient strongly oscillates and appears noisy (cf. [3]).

Shattered Gradients: A Construction

Consider the function:

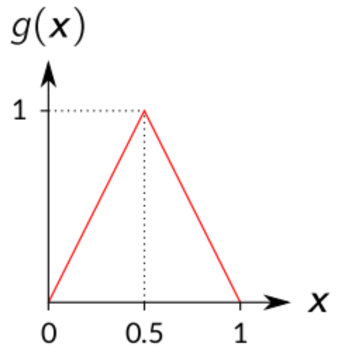
$$g(x) = 2 \cdot \mathrm{ReLU}(x) - 4 \cdot \mathrm{ReLU}(x - 0.5)$$

defined on the interval $[0, 1]$.

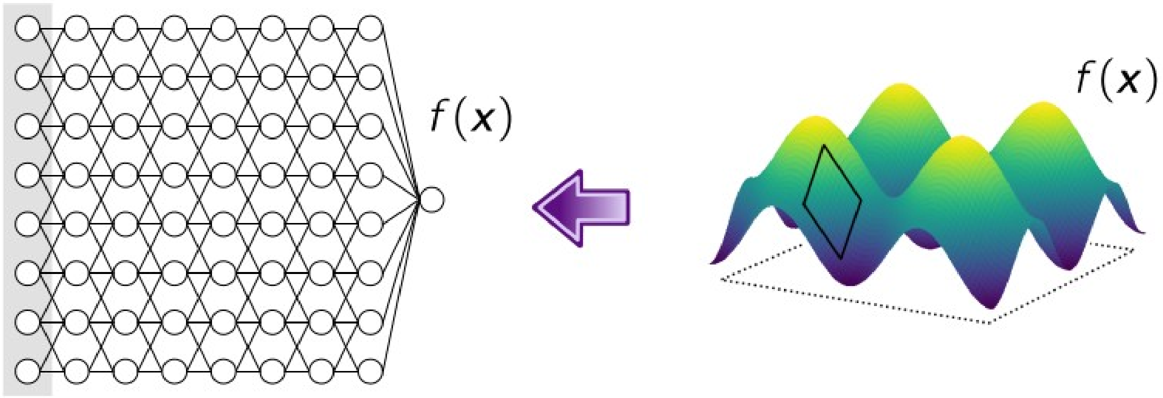
We apply the function recursively to form a deep neural network.

function	output	max slope	# linear pieces
$g(x)$	$[0, 1]$	2	2
$g \circ g(x)$	$[0, 1]$	4	4
$g \circ g \circ g(x)$	$[0, 1]$	8	8
$g \circ g \circ g \circ g(x)$	$[0, 1]$	16	16

Potentially exponential growth of gradient and linear pieces (cf. [11]).



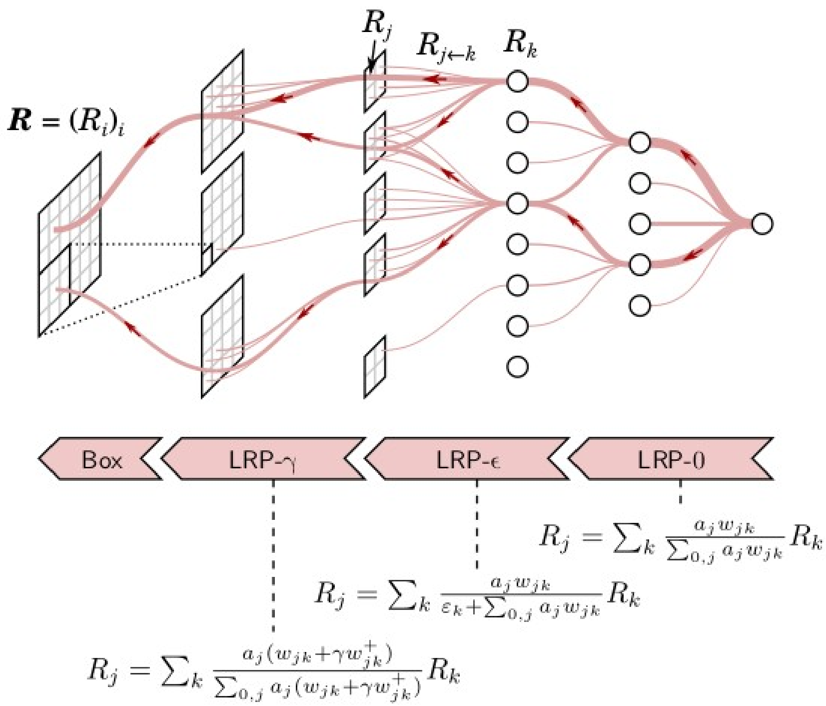
From Function-Based to Propagation-Based



Questions:

- Can using the structure of the network *explicitly* (e.g. by running a special propagation pass) help to produce a better explanation?
- Can this approach reduce explanation noise *without* having to evaluate the function multiple times?

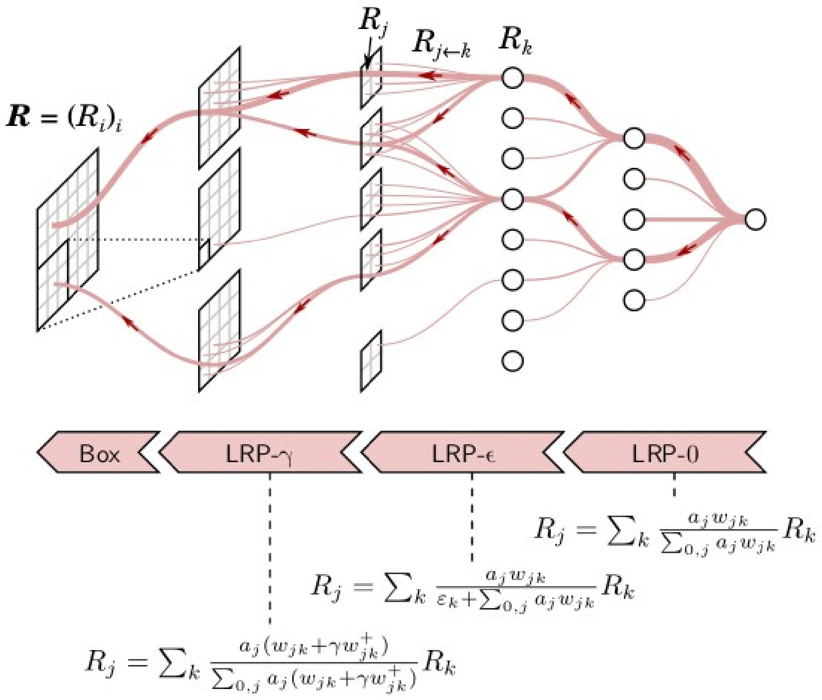
Layer-wise Relevance Propagation (LRP)



Ideas:

- Use the structure of the neural network to robustly compute relevance scores for the input features.
- Propagate the output of the network backwards by means of propagation rules.
- Propagation rules can be tuned for explanation quality. E.g. sensitive in top-layers, robust in lower layers.

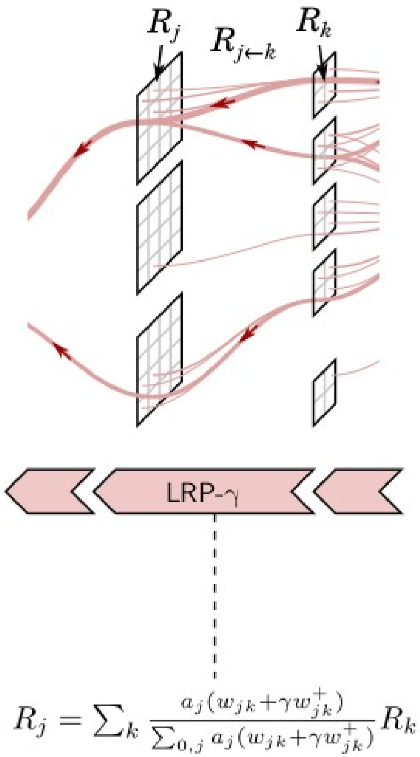
Layer-wise Relevance Propagation (LRP)



Some notation:

- j and k: neurons from successive layers
- w_{jk}: weight connecting neuron j to neuron k
- w_{0k}: bias for neuron k.
- $\sum_{0,j}$ sum over all input neurons j of neuron k and the bias.
- ReLU neuron: $a_k = \max\left(0, \sum_{0,j} a_j w_{jk}\right)$.

Dissecting an LRP Propagation Rule

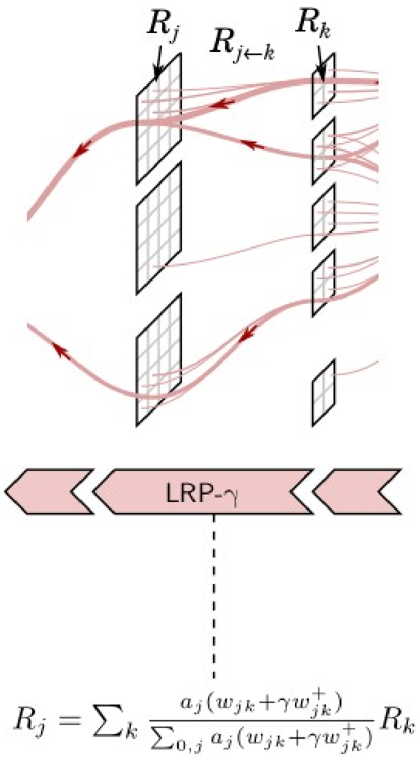


Example: LRP-γ [9]

$$R_j = \sum_k \frac{a_j(w_{jk} + \gamma w_{jk}^+)}{\sum_{0,j} a_j(w_{jk} + \gamma w_{jk}^+)} R_k$$

- $a_j(w_{jk} + \gamma w_{jk}^+)$: Contribution of neuron a_j to the activation a_k.
- R_k 'Relevance' of neuron k available for redistribution.
- $\sum_{0,j} a_j(w_{jk} + \gamma w_{jk}^+)$ Normalization term that implements conservation.
- $\sum_k$: Pool all 'relevance' received by neuron j from the layer above.

Dissecting an LRP Propagation Rule



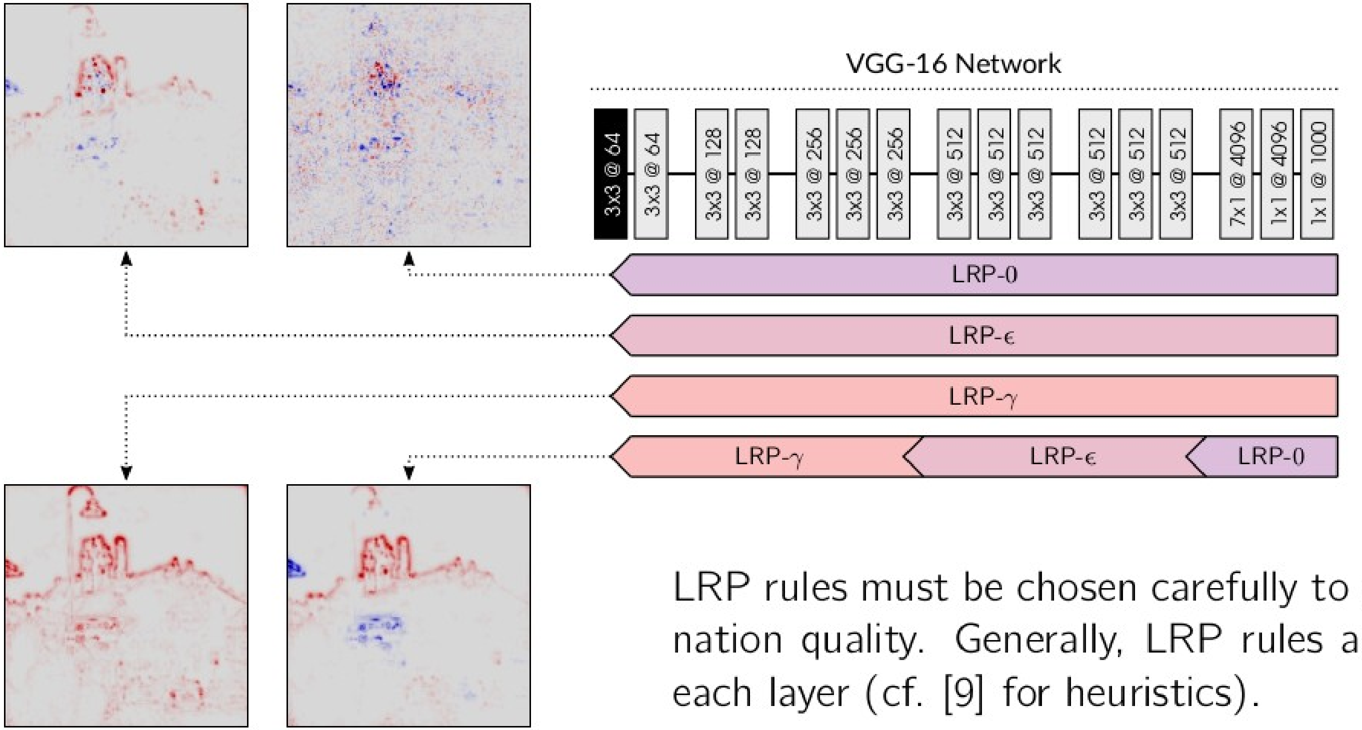
Example: LRP-γ [9]

$$R_j = a_j \cdot \left(\sum_k \frac{(w_{jk} + \gamma w_{jk}^+)}{\sum_{0,j} a_j (w_{jk} + \gamma w_{jk}^+)} R_k \right)$$

- a_j : Activation of neuron j.
- $\left(\sum_k \ldots\right)$: Sensitivity of neural network output to a_j.

i.e. similar interpretation as for Gradient $\times$ Input, but now at each layer.

Effect of LRP Rules on Explanation



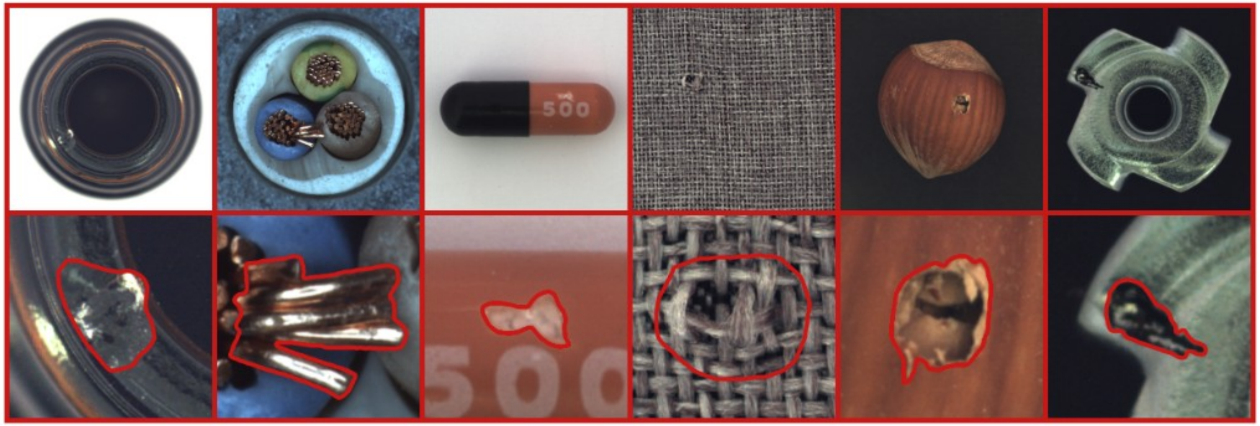
LRP rules must be chosen carefully to deliver best explanation quality. Generally, LRP rules are set different at each layer (cf. [9] for heuristics).

Part 3: Extensions and Applications

- Application to Anomaly Detection
  - Unsupervised XAI
- Applications to Quantum Chemistry
  - Higher-Order Explanations

Anomaly Detection for Industrial Inspection

MVTec Anomaly Dataset

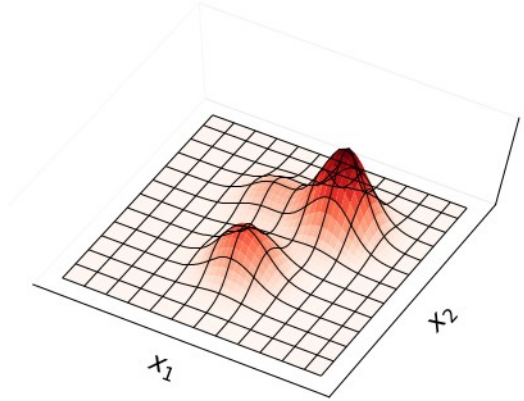


- In many cases we don't have labels that are representative of every possible anomaly. Therefore, we need unsupervised learning.
- Deep networks have been successful on supervised tasks, but other models such as kernels remain popular on unsupervised tasks.

Example: Detecting Anomalous Wood Images

Kernel Density Estimation (KDE)

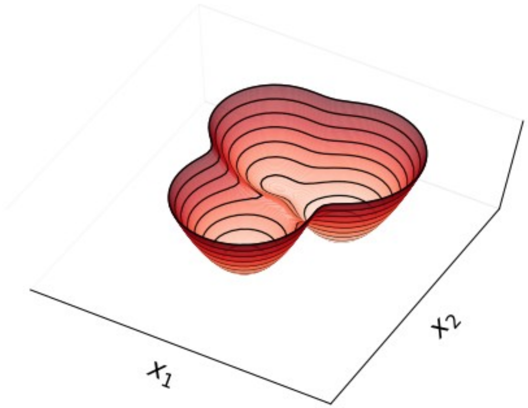
$$f(\boldsymbol{x}) = \sum_{i=1}^{N} \frac{1}{N} \exp(-\gamma \|\boldsymbol{x} - \boldsymbol{x}_i\|^2)$$



Example: Detecting Anomalous Wood Images

Kernel Density Estimation (KDE)

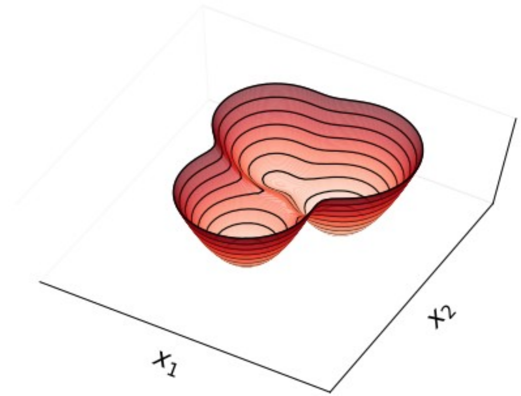
$$o(\boldsymbol{x}) = -\log \sum_{i=1}^{N} \frac{1}{N} \exp(-\gamma \|\boldsymbol{x} - \boldsymbol{x}_i\|^2)$$



Neuralizing the KDE Model

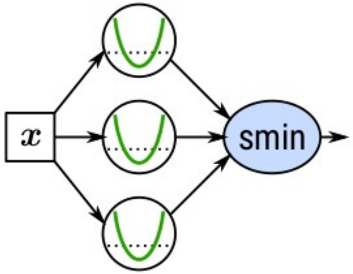
Standard (non-explainable) formulation:

$$o(\boldsymbol{x}) = -\log\Big(\sum_{i=1}^{N} \frac{1}{N}\exp(-\gamma\|\boldsymbol{x}-\boldsymbol{x}_i\|^2)\Big)$$



'Neuralized' formulation:

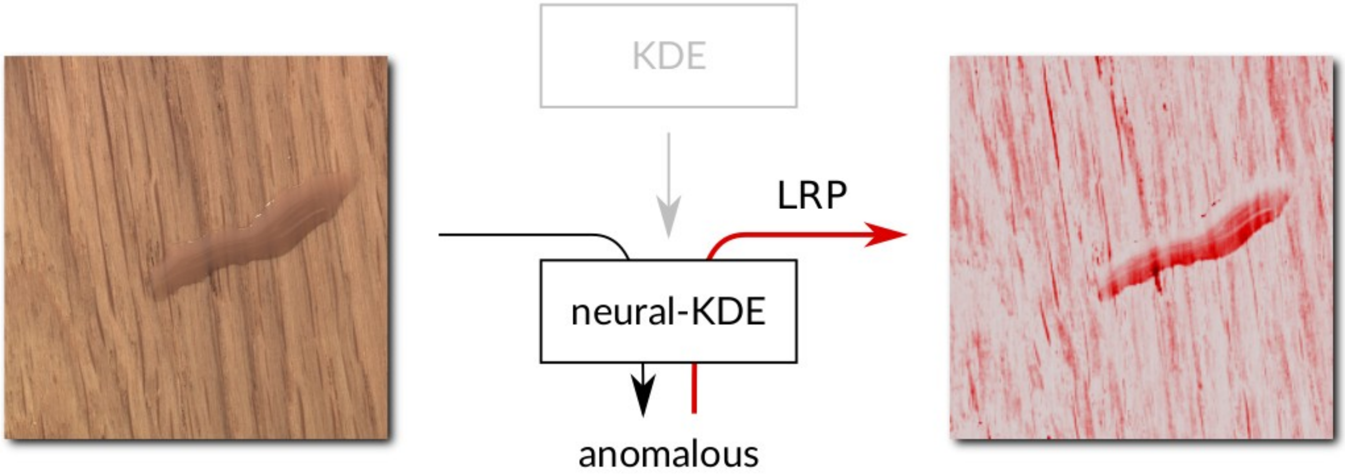
$$\begin{aligned} h_j &= \gamma\|\boldsymbol{x}-\boldsymbol{x}_i\|^2 + \log N && \text{(layer 1)} \\ o(\boldsymbol{x}) &= \underbrace{-\log\textstyle\sum_j \exp(-}_{\text{softmin}} h_j) && \text{(layer 2)} \end{aligned}$$



The KDE model predictions can now be explained with LRP.

Kauffmann et al. (2020) The Clever Hans Effect in Anomaly Detection arXiv:2006.10609

Explaining an Anomaly Decision



Observation:

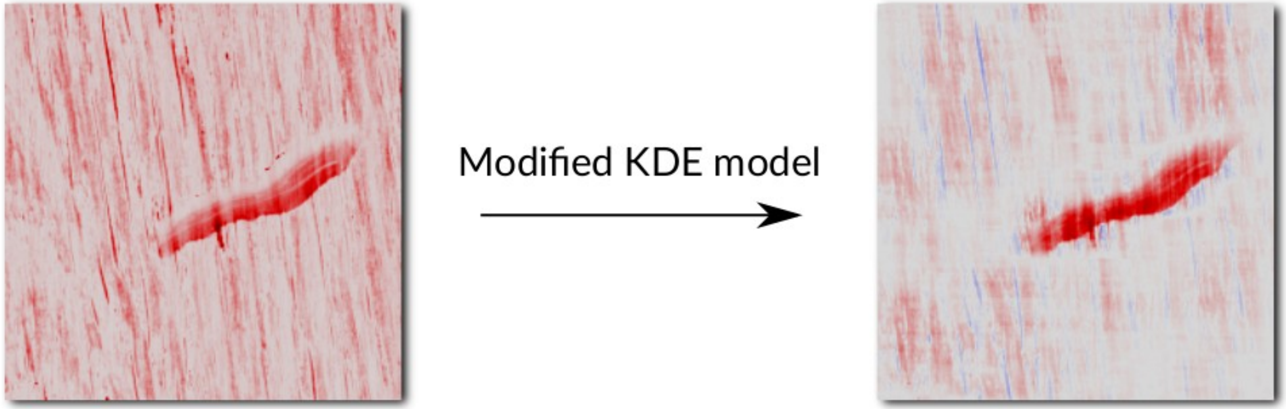
- Both the liquid stain and the wood grain are found to be responsible for the predicted anomaly (the wood grain should not!).

Kauffmann et al. (2020) The Clever Hans Effect in Anomaly Detection arXiv:2006.10609

Correcting the Model Weaknesses

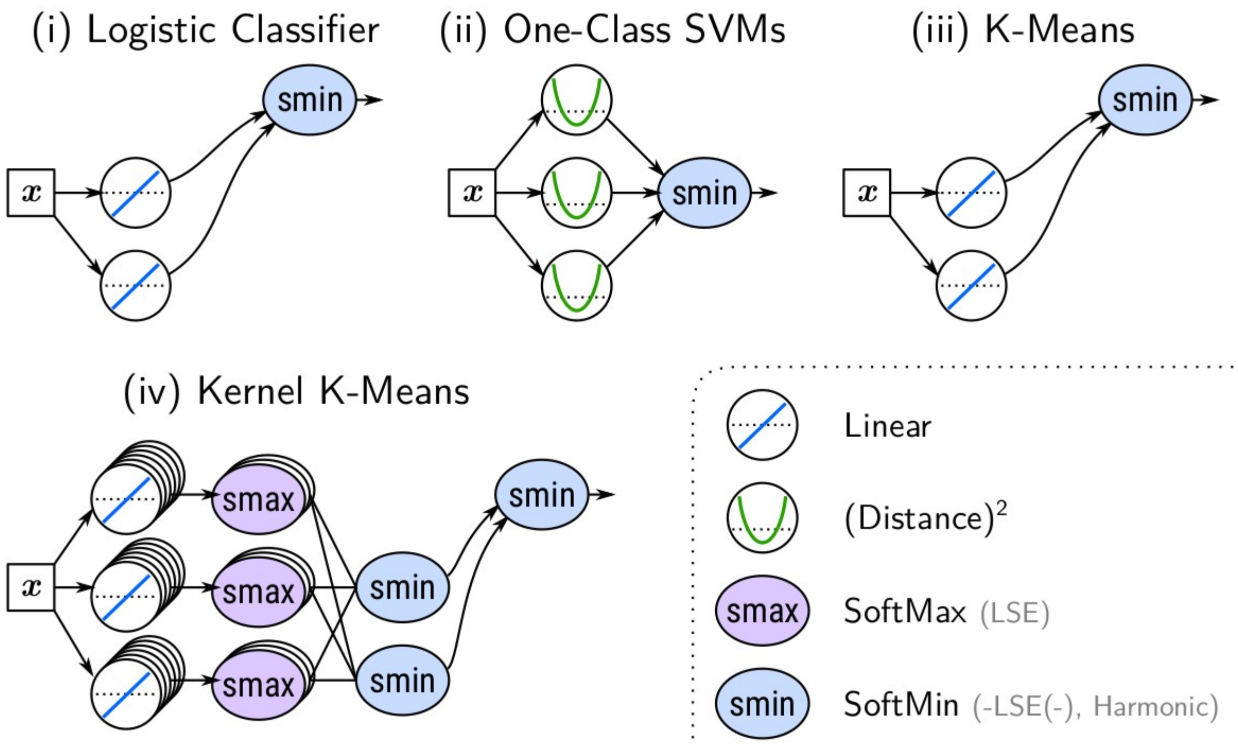
Idea: Replace in the original KDE model the Euclidean metric by a Malahanobis metric with covariance Σ hardcoded to reduce the high horizontal frequencies.

$$f(\boldsymbol{x}) = \sum_{i=1}^{N} \frac{1}{N} \exp(-\gamma\, (\boldsymbol{x} - \boldsymbol{x}_i)^\top \Sigma (\boldsymbol{x} - \boldsymbol{x}_i))$$



- The anomaly decision is now supported by the correct features.

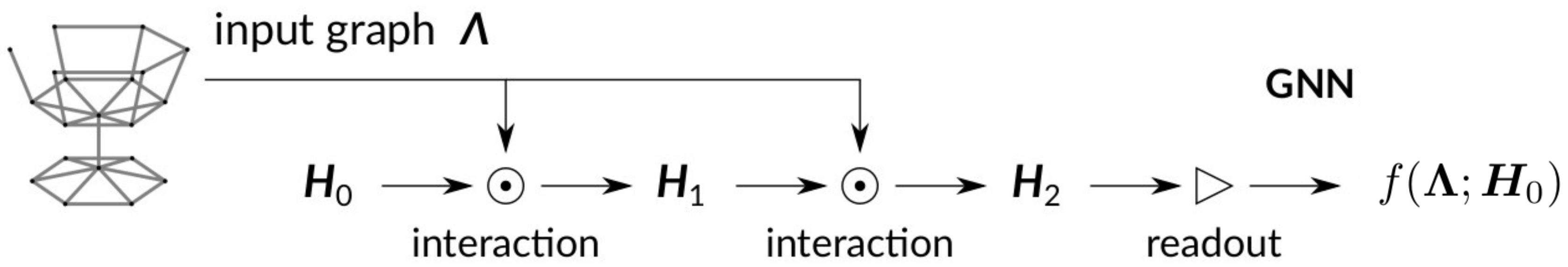
Neuralization-Propagation as a General Technique



Kauffmann et al. (2019): From Clustering to Cluster Explanations via Neural Networks.

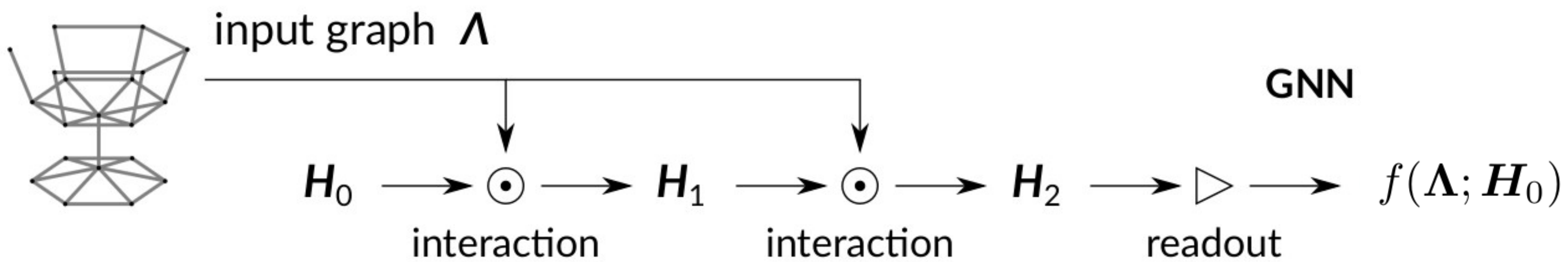
Kauffmann et al. (2020): Towards explaining anomalies: A deep Taylor decomposition of one-class models.

Explaining Graph Neural Networks



Schnake et al. (2020) Higher-Order Explanations of Graph Neural Networks via Relevant Walks

Explaining Graph Neural Networks

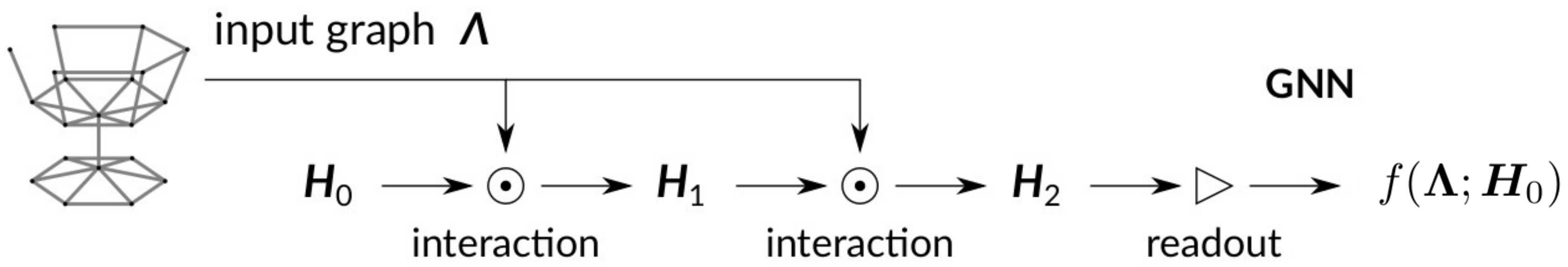


Observations:

- The input $\boldsymbol{\Lambda}$ occurs at every layer of the network.
- The function f is piecewise polynomial with $\boldsymbol{\Lambda}$.

Schnake et al. (2020) Higher-Order Explanations of Graph Neural Networks via Relevant Walks

Explaining Graph Neural Networks



Idea:

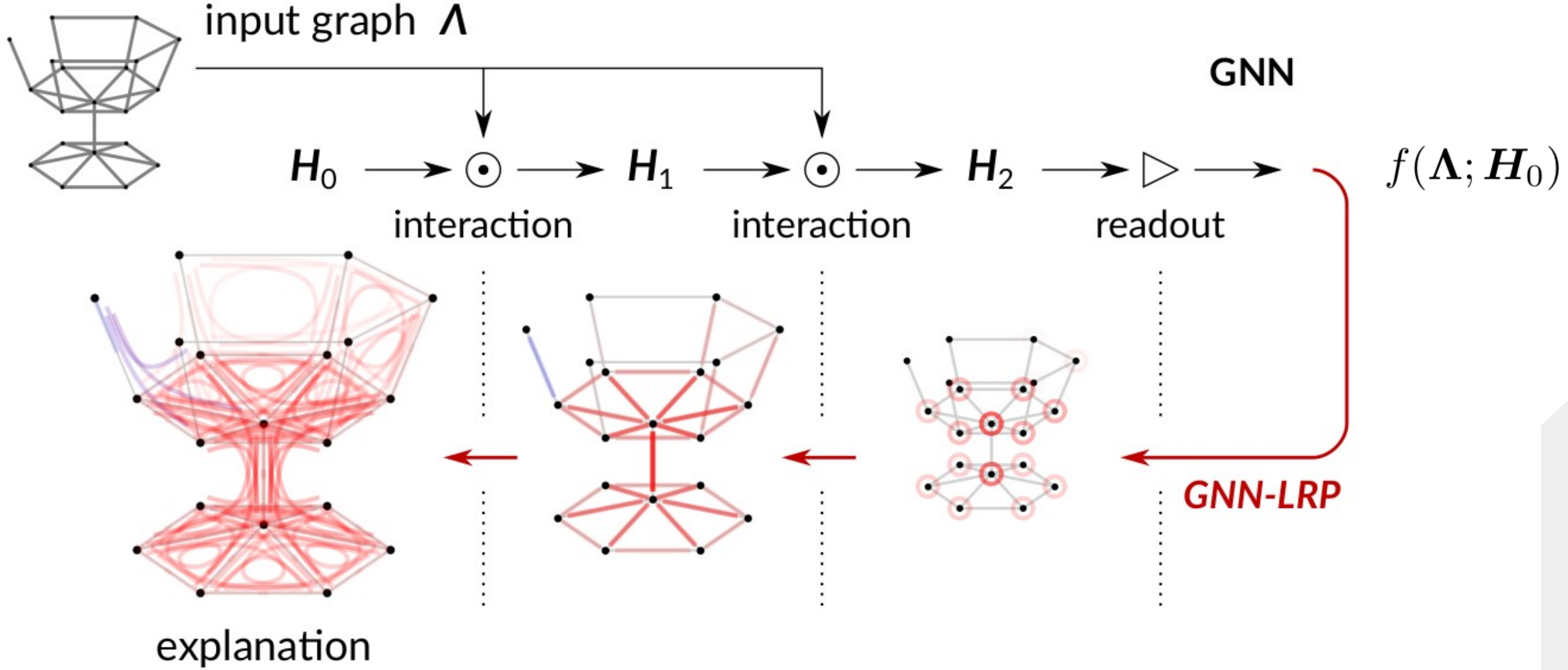
- First consider input in the last layer, i.e., $\Lambda^{(l)}$, and attribute on $\Lambda^{(l)}$.

- Then express contribution of each variable in $\Lambda^{(l)}$ in terms of the input $\Lambda^{(l-1)}$.

- When we arrive at layer zero, we have identified the contribution of all paths between nodes at each layer (can be interpreted as walk into the graph).

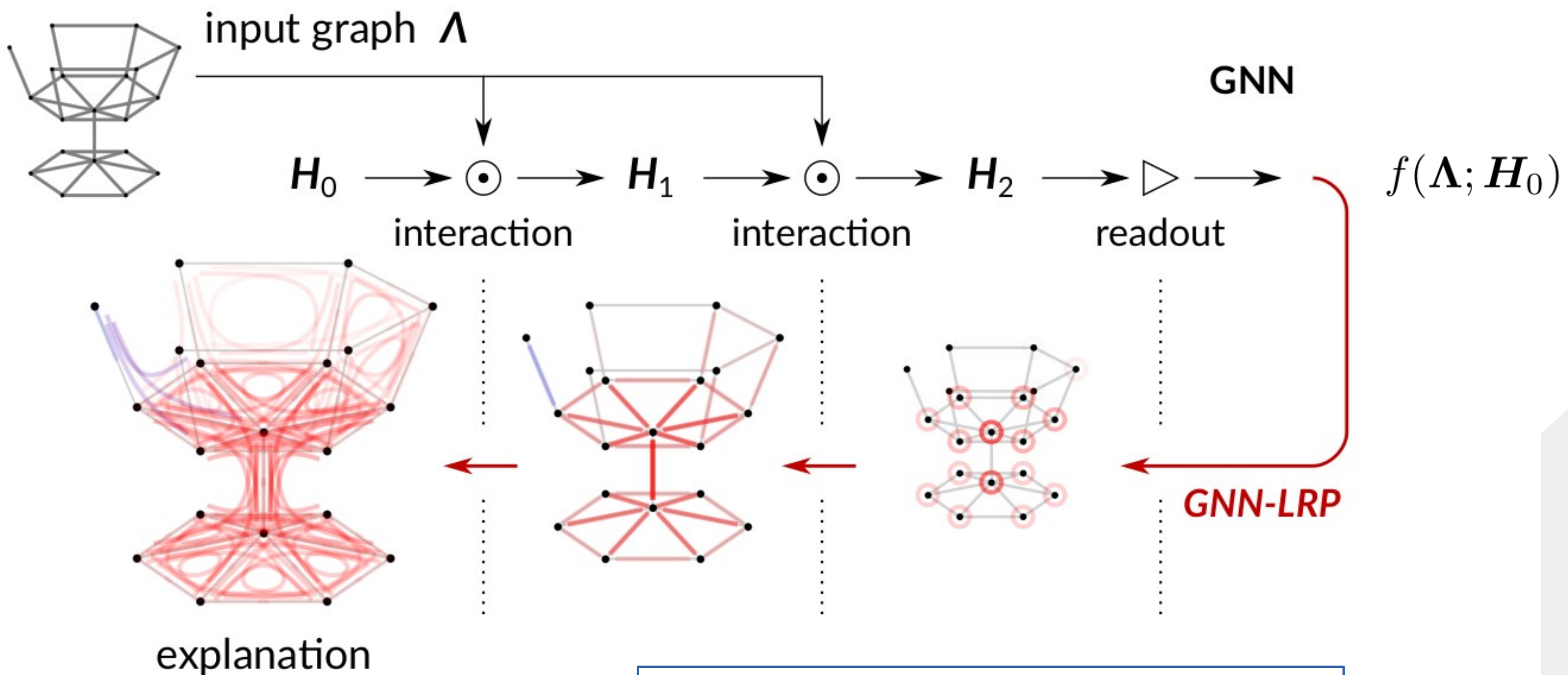
Schnake et al. (2020) Higher-Order Explanations of Graph Neural Networks via Relevant Walks

Explaining Graph Neural Networks



Schnake et al. (2020) Higher-Order Explanations of Graph Neural Networks via Relevant Walks

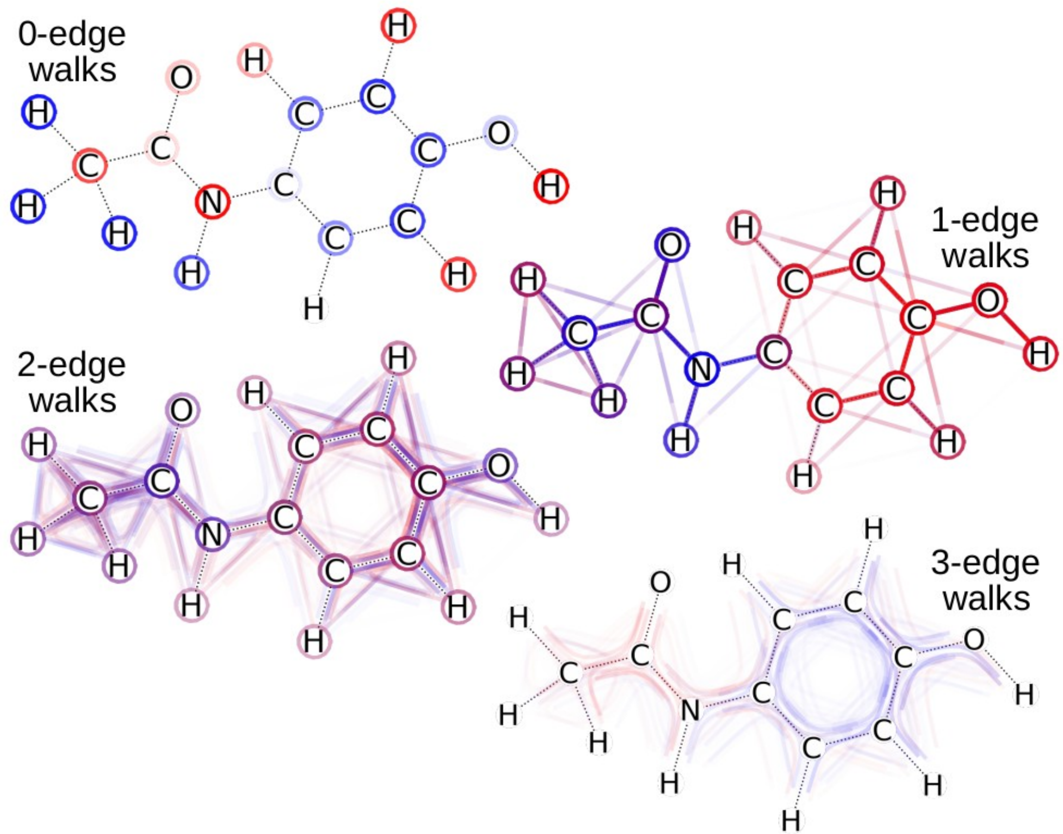
Explaining Graph Neural Networks



Can be interpreted as a *higher-order* analysis of the function *f*.

Explaining Molecular Polarizability with GNN-LRP

Example:
Paracetamol molecule

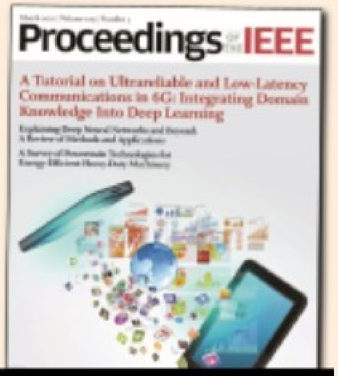


Our Review Paper

W Samek, G Montavon, S Lapuschkin, C Anders, KR Müller

Explaining Deep Neural Networks and Beyond: A Review of Methods and Applications

Proceedings of the IEEE, 109(3):247-278, 2021



With the broader and highly successful usage of machine learning (ML) in industry and the sciences, there has been a growing demand for explainable artificial intelligence (XAI). Interpretability and explanation methods for gaining a better understanding of the problem-solving abilities and strategies of nonlinear ML, in particular, deep neural networks, are, therefore, receiving increased attention. In this work, we aim to: 1) provide a timely overview of this active emerging field, with a focus on " post hoc " explanations, and explain its theoretical foundations; 2) put interpretability algorithms to a test both from a theory and comparative evaluation perspective using extensive simulations; 3) outline best practice aspects, i.e., how to best include interpretation methods into the standard usage of ML; and 4) demonstrate successful usage of XAI in a representative selection of application scenarios. Finally, we discuss challenges and possible future directions of this exciting foundational field of ML.

Check our Website

Online demos, tutorials, code examples, software, etc.

References

[1] 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)

[2] J Kauffmann, KR Müller, G Montavon. Towards Explaining Anomalies: A Deep Taylor Decomposition of One-Class Models, Pattern Recognition, 107198, 2020

[3] T Schnake, O Eberle, J Lederer, S Nakajima, K T. Schütt, KR Müller, G Montavon. Higher-Order Explanations of Graph Neural Networks via Relevant Walks, IEEE TPAMI, 2021

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

[5] W Samek, G Montavon, S Lapuschkin, C Anders, KR Müller. Explaining Deep Neural Networks and Beyond: A Review of Methods and Applications. Proceedings of the IEEE, 109(3):247-278, 2021