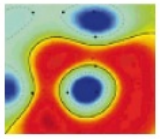
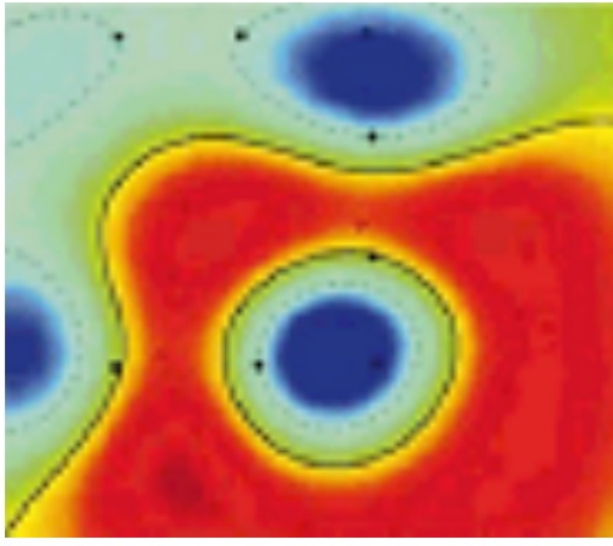


WiSe 2024/25

Machine Learning 1/1-X

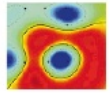
Technische Universität Berlin



Lecture 14 | Explainable AI

Outline

- What is Explainable AI?
- Desiderata of an Explainable AI technique
- Uses of Explainable AI
- Methods for Explainable AI
  - Activation Maximization
  - Shapley Values
  - Taylor Expansions
  - Layer-wise Relevance Propagation

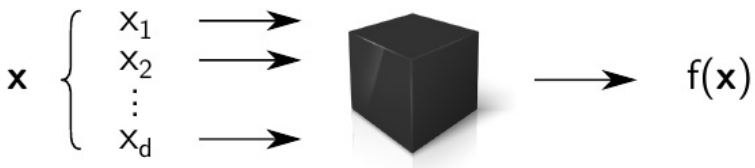


What is Explainable AI?

🔷 函数 f 通常被认为是一个"黑箱"，其参数是通过数据学习得到的，例如使用梯度下降法。优化目标是使预测值 $f(x)$ 尽可能接近训练数据和测试数据的真实值。

Standard machine learning:

- The function f is typically considered to be a "black-box" whose parameters are learned from the data using e.g. gradient descent. The objective to minimize encourages the predictions $f(\boldsymbol{x})$ to coincide with the ground truth on the training and test data.



Machine learning + Explainable AI:

- We do not only look at the outcome $f(\boldsymbol{x})$ of the prediction but also at the way the prediction is produced by the ML model, e.g. which features are used, how these features are combined, or to what input pattern the model responds the most.

  - 模型对哪些输入模式最敏感

What is Explainable AI?

合成一个输入模式，使其能够最强烈地激活机器学习模型的某个特定类别的输出。

Example 1: Synthesize an input pattern that most strongly activates the output of the ML model associated to a particular class.

Image source: Nguyen et al. (2016) Multifaceted Feature Visualization: Uncovering the Different Types of Features Learned By Each Neuron in Deep Neural Networks

在神经网络中，我们可以使用**反向传播优化输入数据**，使其尽可能激活目标类别的神经元。例如：

- 在图像分类任务中，找到一张合成的图片，它能够最大程度地激活"猫"这个类别的神经元。
- 在文本分类任务中，找到一段文本，使其能最大程度地被模型识别为"正面情感"类别。

What is Explainable AI?

Example 2: Highlight features that have contributed for a given data point to the ML prediction.

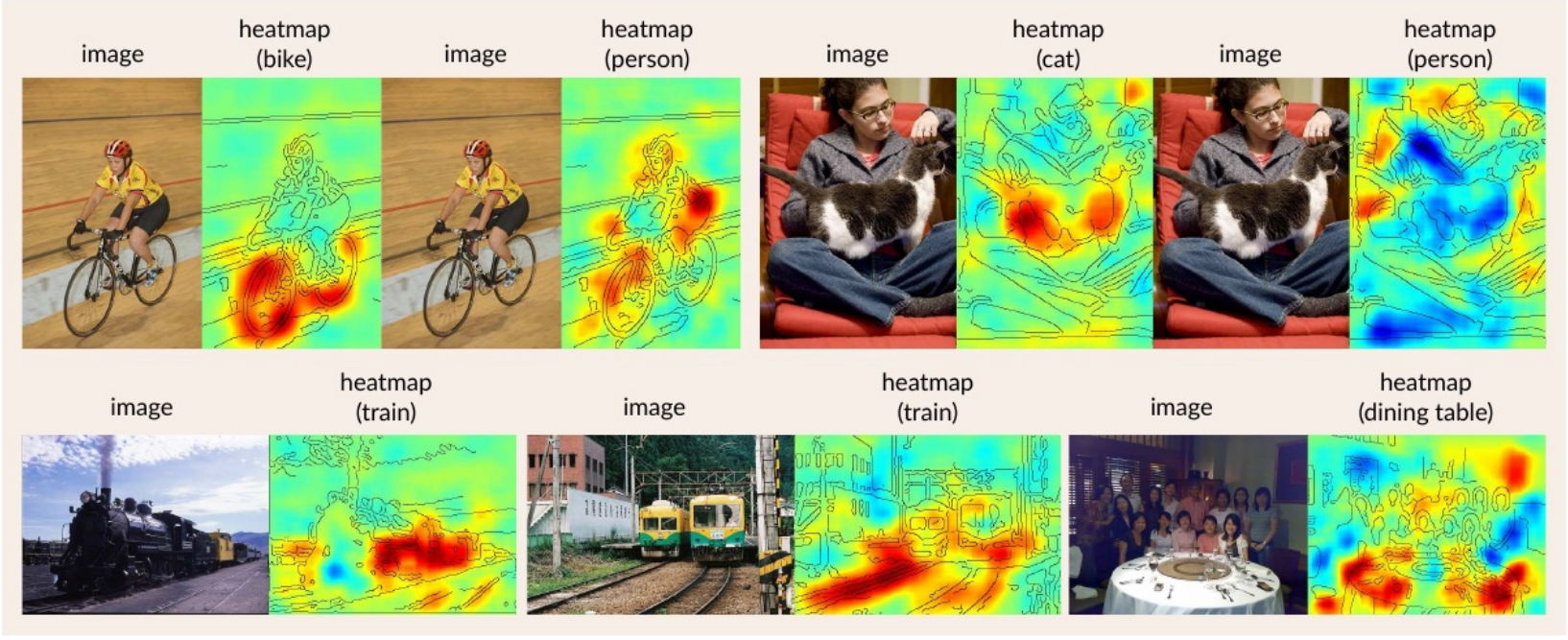
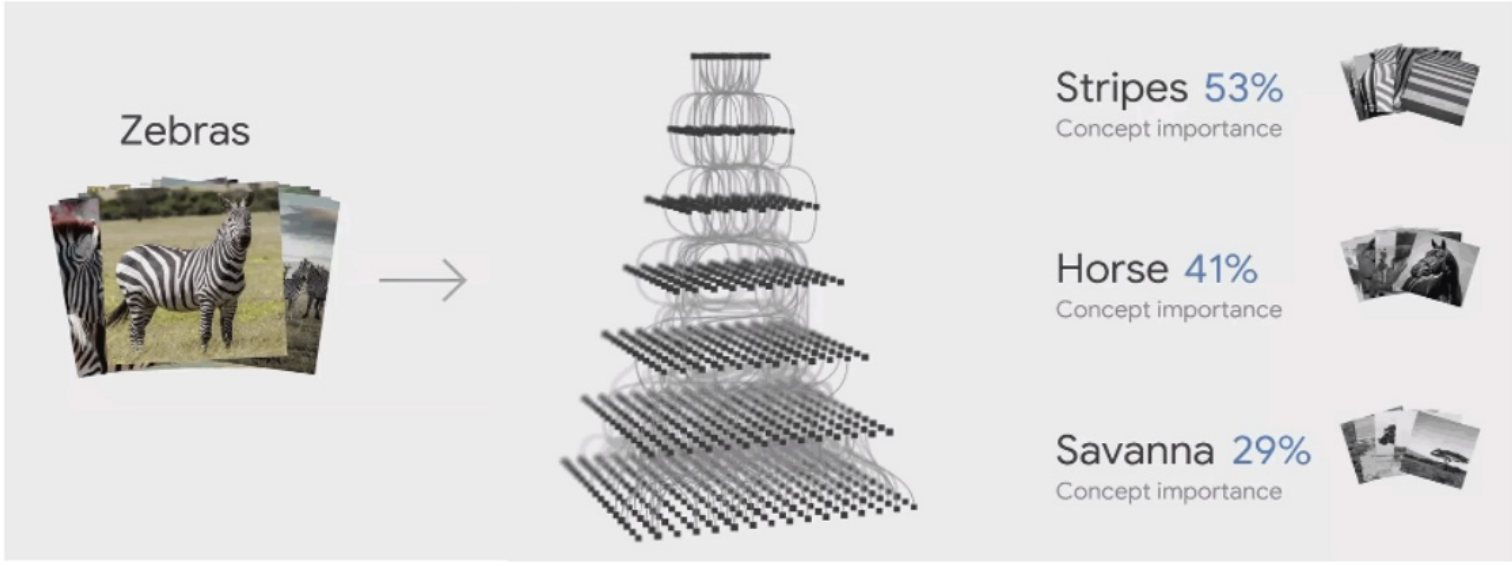


Image source: Lapuschkin et al. (2016) Analyzing Classifiers: Fisher Vectors and Deep Neural Networks

What is Explainable AI?

概念激活向量（TCAV）。突出显示用于解释某个数据点的机器学习预测的中层概念。

Example 3: Concept activation vectors (TCAV). Highlight the mid-level concepts that explain, for a given data point, the ML prediction.



帮助我们理解**模型的决策是基于哪些概念。**

www.youtube.com/watch?v=lyRPyRKHO8M&t=2279s)

示例分析（斑马分类）：

- 假设一个深度神经网络被训练用于识别动物，并正确分类了一张斑马的图片。
- 但是，模型的决定是基于什么？TCAV 可以分析中层特征，发现：
  - **条纹（Stripes）** 对于识别"斑马"最重要，占比 **53%**。
  - **马（Horse）** 的特征也有一定影响，占比 **41%**（因为斑马和马形态相似）。
  - **稀树草原（Savanna）** 也影响预测，占比 **29%**（斑马的自然栖息地）。

Desiderata of an Explanation

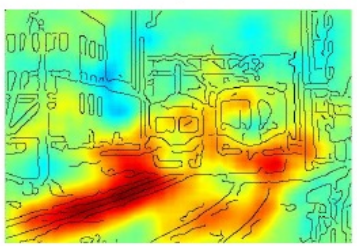
In practice, we would like the explanation technique to satisfy a number of properties:

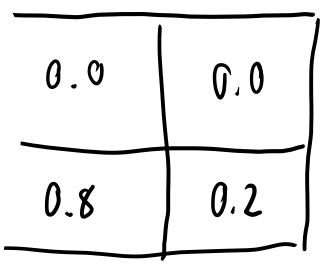
1. **Fidelity:** The explanation should reflect the quantity being explained and not something else.
2. **Understandability:** The explanation must be easily understandable by its receiver.
3. **Sufficiency:** The explanation should provide sufficient information on how the model came up with its prediction.
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.

see also *Swartout & Moore (1993), Explanation in Second Generation Expert Systems.*

image

heatmap (train)

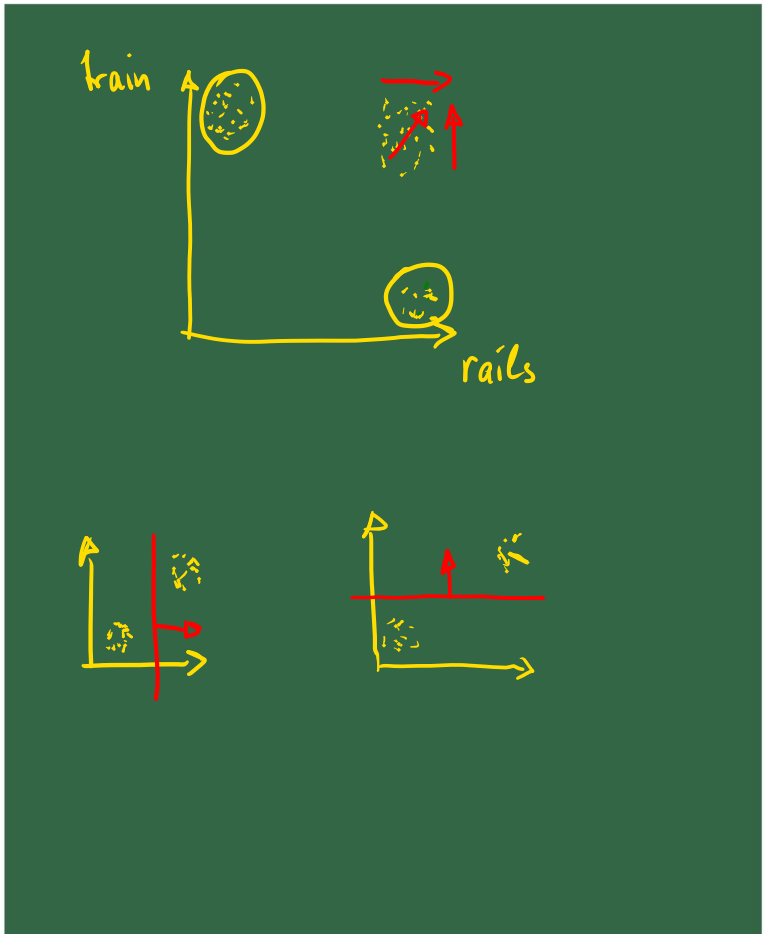




Uses of an Explanation

Verify (and improve?) a ML model

- Verify that the model is based on features which *generalize well* to examples outside the current data distribution (this cannot be done with standard validation techniques!).
- Reliance of the ML models on wrong features is often encountered when there are *spurious correlation* in the data.
- From the explanation, the model's trustworthiness can be reevaluated, and the flawed ML model can be potentially *retrained* based on the user feedback.



验证（并改进？）机器学习模型

- **验证** 模型是否基于**能够很好泛化**到当前数据分布之外的示例的特征（标准验证技术无法做到这一点！）。
- **机器学习模型依赖错误特征** 的情况通常出现在数据中存在**虚假相关（spurious correlation）**时。
- 通过解释，我们可以**重新评估模型的可信度，并基于用户反馈对有缺陷的机器学习模型进行再训练（retrain）**。

Uses of an Explanation

示例：分类器正确地做出了错误的决策

- **图片（左侧）：** 一个骑马的人，图片左下角带有版权标记。
- **热力图（中间）：** 机器学习模型用于识别"马"（horse）的特征重要性图。
- **表格（右侧）：** 在 Pascal VOC 数据集上的 Fisher Vector 模型的类别平均精度。

分析： 🔷 在这个示例中，**分类器准确预测了"马"类别，但它基于错误的特征（如左下角的版权标记）进行判断。**

🔷 这种错误的决策方式**无法通过测试误差检测到**，只有使用解释技术才能发现。

Example: The classifier is right for the wrong reasons

image

heatmap (horse)

average precision of the Fisher Vector model on the PascalVOC dataset

aer	bic	bir	boa	bot
79.08	66.44	45.90	70.88	27.64
bus	car	cat	cha	cow
69.67	80.96	59.92	51.92	47.60
din	dog	hor	mot	per
58.06	42.28	80.45	69.34	85.10
pot	she	sof	tra	tvm
28.62	49.58	49.31	82.71	54.33

- In this example, the classifier accurately predicts the horse class, but based on the wrong features (some copyright tag in the corner).
- This incorrect decision strategy cannot be detected by just looking at the test error.

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

Uses of an Explanation

Learn something about the data *(or about the system that produced the data)*

- **Step 1:** Train a ML model that predicts well the data.
- **Step 2:** Apply XAI to the trained ML model to produce explanations of the ML decision strategy.
- **Step 3:** Based on the XAI explanations, the user can compare his reasoning with that of the ML model, and can potentially refine his own domain knowledge.

学习数据（或生产数据的系统）的信息

步骤 1：训练一个能很好预测数据的机器学习模型。

步骤 2：将**可解释性人工智能（XAI）** 应用于训练好的机器学习模型，以生成模型决策策略的解释。

步骤 3：基于 XAI 生成的解释，用户可以**比较自己的推理与模型的推理**，并**潜在地改进自己的领域知识**。

图片右侧：

- 这是一个**神经影像分析**的示例，展示了如何使用 XAI 解释**深度学习模型的决策过程**。

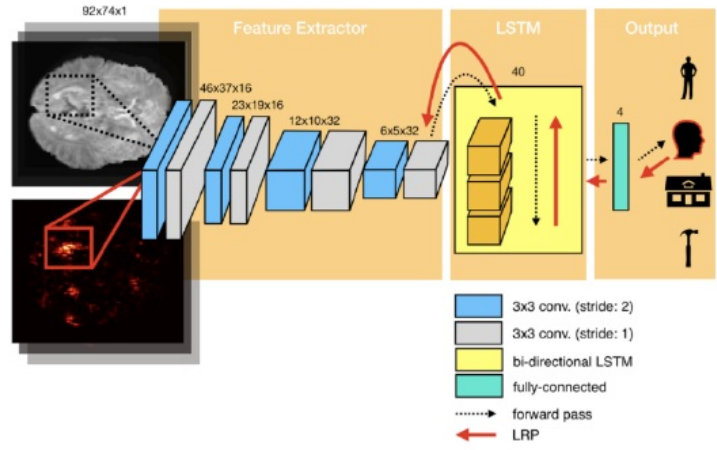


Image source: Thomas et al. (2019) Analyzing Neuroimaging Data Through Recurrent Deep Learning Models

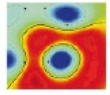
Part II: Methods of XAI

Presented methods

- Activation maximization
- Shapley values
- Taylor expansion
- Layer-wise relevance propagation

Other methods

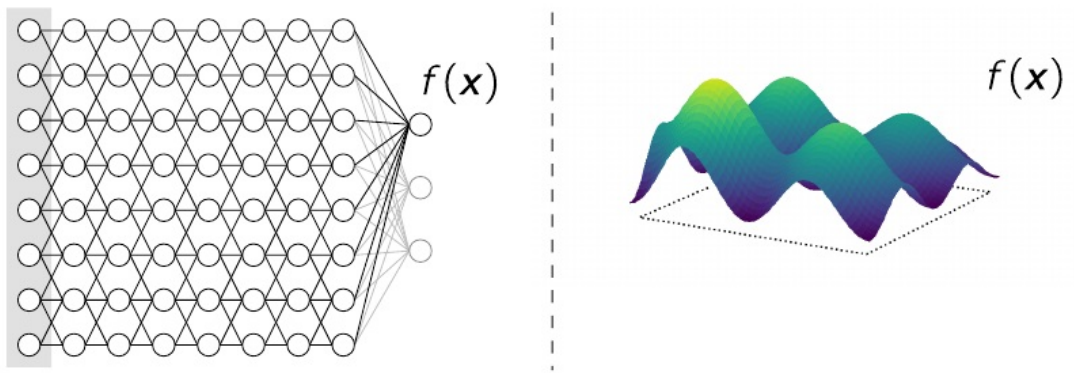
- Surrogate models (LIME)
- Integrated gradients / expected gradients / SmoothGrad
- Influence functions
- ...



Activation Maximization

Assume a trained a ML model (e.g. a neural network), and we would like to understand what concept is associated to some particular output neuron of the ML model, e.g. the output neuron that codes for the class 'cat'. Activation maximization proceeds in two steps:

- **Step 1:** Think of the ML model as a function of the input



- **Step 2:** Explain the function f by generating a maximally activating input pattern:

$$\boldsymbol{x}^\star = \arg\max_{\boldsymbol{x}} f(\boldsymbol{x}, \theta)$$

Activation Maximization

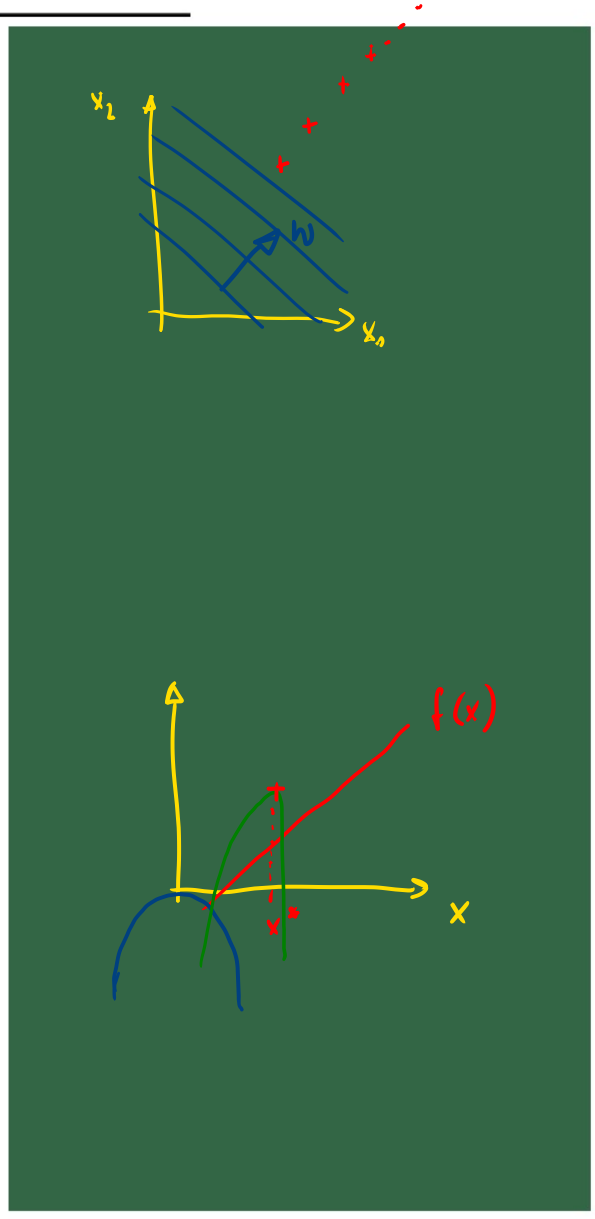
Problem: In most cases $f(\boldsymbol{x})$ does not have single point corresponding to the maximum.

> *E.g. in linear models, $f(\boldsymbol{x}) = \boldsymbol{w}^\top \boldsymbol{x} + b$, we can keep moving the point $\boldsymbol{x}$ further along the direction $\boldsymbol{w}$, and the output continues to grow).*

Therefore, we would like to apply a preference for 'regular' regions of the input domain, i.e.

$$\boldsymbol{x}^\star = \arg\max_{\boldsymbol{x}} f(\boldsymbol{x}) + \Omega(\boldsymbol{x})$$

In practice, the preference can be for data points with small norm (i.e. we set $\Omega(\boldsymbol{x}) = -\lambda\|\boldsymbol{x}\|^2$ so that points with large norm are penalized.)



Activation Maximization: Examples

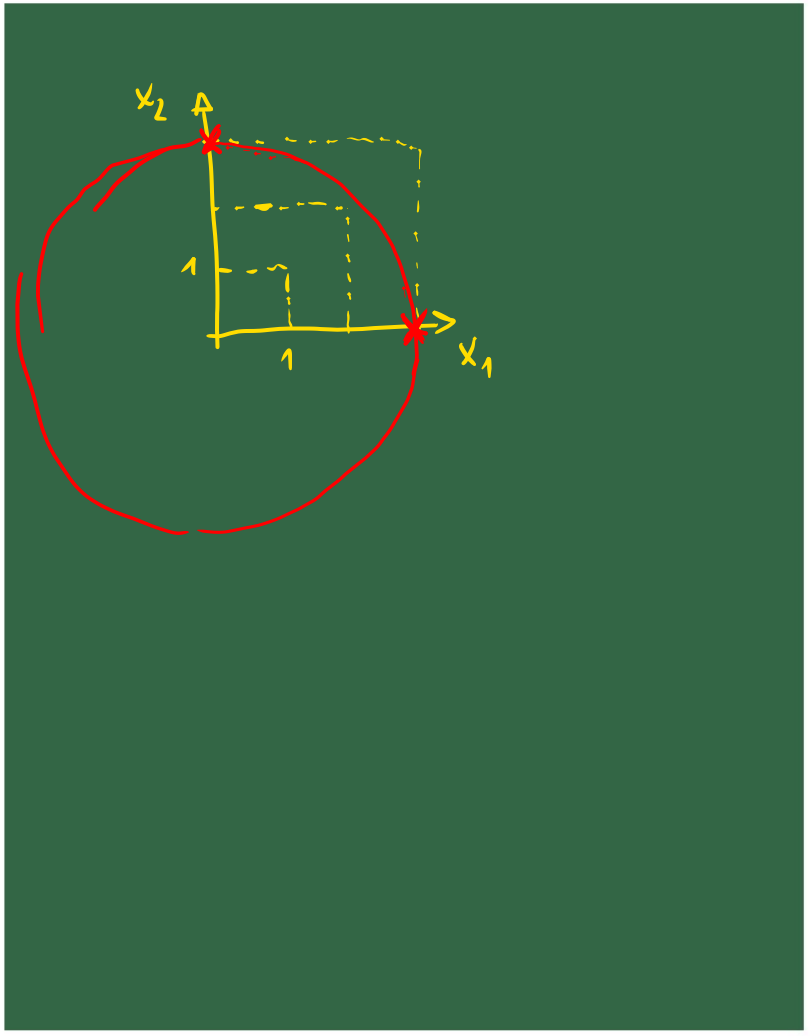
$f(\boldsymbol{x}) = \boldsymbol{w}^\top \boldsymbol{x} + b$ and
$\Omega(\boldsymbol{x}) = -\lambda \|\boldsymbol{x}\|^2$

$$\arg\max_x \underbrace{w^T x + b - \lambda \|x\|^2}_{J(x)}$$

$$\frac{\partial J}{\partial x} = w - 2\lambda x \stackrel{!}{=} 0$$

$$x^* = \frac{1}{2\lambda} \cdot w$$

$f(\boldsymbol{x}) = \max(x_1, x_2)$ and
$\Omega(\boldsymbol{x}) = -\lambda \|\boldsymbol{x}\|^2$



Activation Maximization: Probability View

Assume the model produces a log-probability for class ω_c:

$$f(\boldsymbol{x}) = \log p(\omega_c|\boldsymbol{x})$$

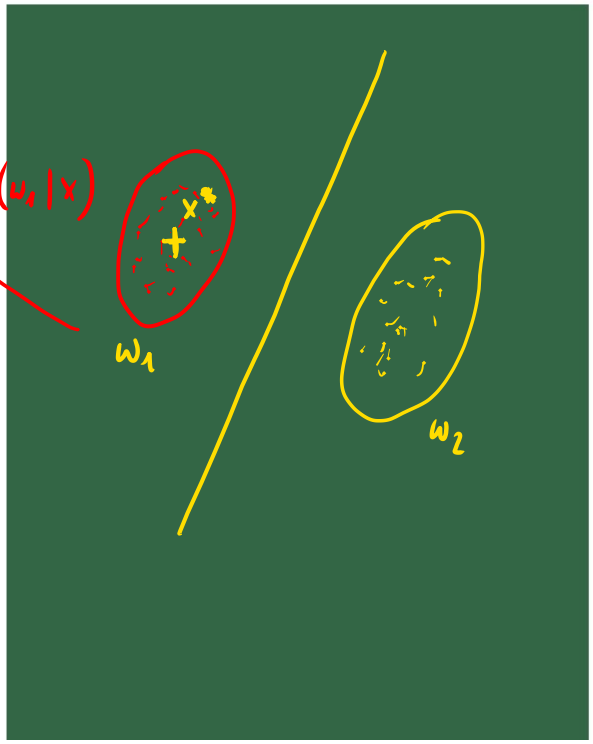
The input $\boldsymbol{x}^\star$ that maximizes this function can be interpreted as the point where the classifier is the most *sure* about class ω_c.

Choose the regularizer $\Omega(\boldsymbol{x}) = \log p(\boldsymbol{x})$, i.e. favor points that are *likely*.

The optimization problem becomes:

$$\begin{aligned}\boldsymbol{x}^\star &= \arg\max_{\boldsymbol{x}} \; \log p(\omega_c|\boldsymbol{x}) + \log p(\boldsymbol{x}) \\ &= \arg\max_{\boldsymbol{x}} \; \log p(\boldsymbol{x}|\omega_c)\end{aligned}$$

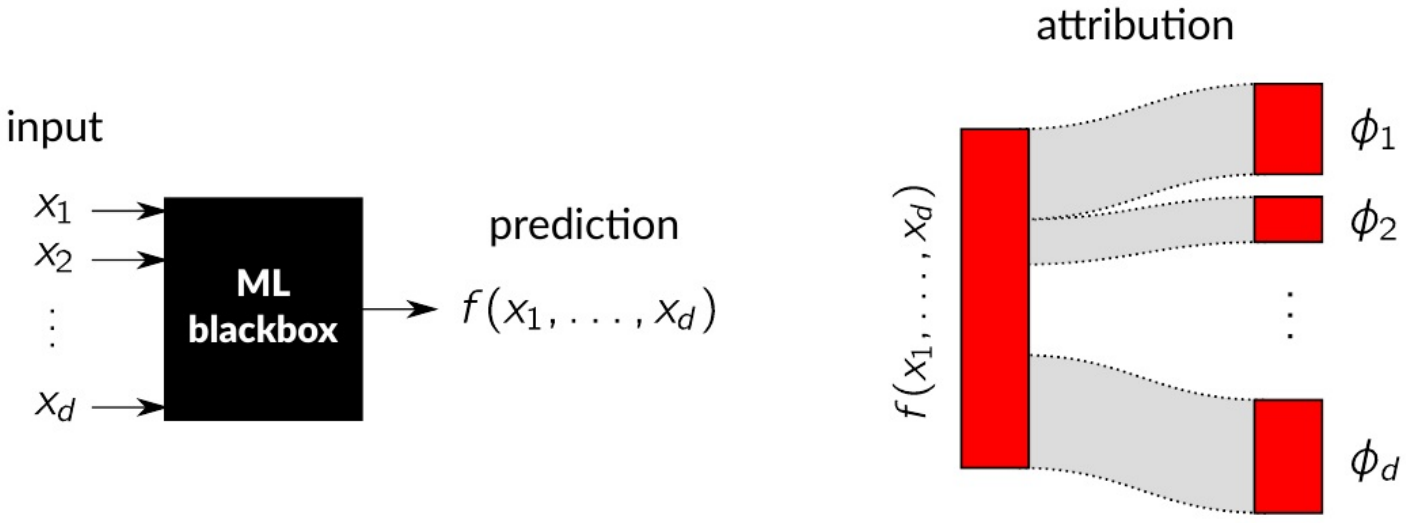
where $\boldsymbol{x}^\star$ can now be interpreted as the most *typical* input for class ω_c.



$$p(\omega_c|\mathbf{x}) = \frac{p(\mathbf{x}|\omega_c)p(\omega_c)}{p(\mathbf{x})}$$

$$\log p(\omega_c|\mathbf{x}) = \log p(\mathbf{x}|\omega_c) + \log p(\omega_c) - \log p(\mathbf{x})$$

Attribution of a Prediction to Input Features



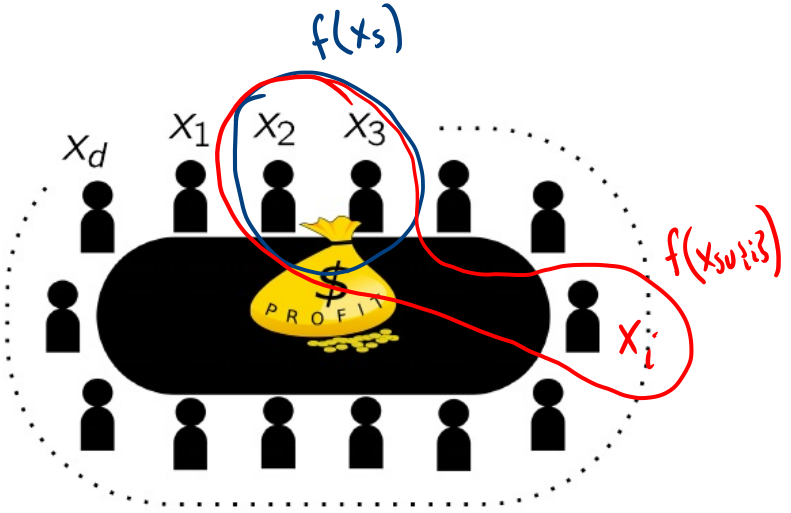
1. The data $\boldsymbol{x} \in \mathbb{R}^d$ is fed to the ML black-box and we get a prediction $f(\boldsymbol{x}) \in \mathbb{R}$.
2. We explain the prediction by determining the contribution of each input feature.

Key property of an explanation: **conservation** ($\sum_{i=1}^{d} \phi_i = f(\boldsymbol{x})$).

Attribution: Shapley Values

- 该框架最初是在博弈论（Shapley 1951）的背景下提出的，用于在合作博弈中分配收益，并最近被应用于机器学习模型。
- 每个输入变量被视为一个玩家，而函数输出则被视为合作玩家所实现的收益。

- 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:

$$\phi_i = \sum_{S:\, i \notin S} \frac{|S|!(d-|S|-1)!}{d!} \left[f(x_{S\cup\{i\}}) - f(x_S) \right]$$

表示当加入特征 x_i 后，模型的预测值变化。

where $(x_S)_S$ are all possible subsets of features contained in the input x.

- S 表示输入特征 $\mathbf{x}$ 的所有可能子集。
- $f(\mathbf{x}_S)$ 表示模型在特征子集 S 下的预测值。

Attribution: Shapley Values

α_S 是 Shapley 值公式中的加权因子，用于衡量子集 S 在所有可能排列中的重要性。

它的作用可以理解为：

- 衡量不同子集 S 对最终 Shapley 值的贡献权重，确保所有可能的排列都被公平计算。
- 保证 Shapley 值满足博弈论中的对称性、公平性等性质。

Recall: $\phi_i = \sum_{S:\, i \notin S} \underbrace{\frac{|S|!(d-|S|-1)!}{d!}}_{\alpha_S} \underbrace{\left[f(x_{S\cup\{i\}}) - f(x_S)\right]}_{\Delta_S}$

注意 i

$1 \cdot (1+1) = 2$

$0 \cdot (1+1) = 0$

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

ϕ_1 :

S	α_S	Δ_S
$\{\}$	$\frac{1!\cdot(3-1)!}{3!} = 1/3$	0
$\{2\}$	$\frac{1!\cdot(3-1-1)!}{3!} = 1/6$	1
$\{3\}$	$1/6$	1
$\{2,3\}$	$1/3$	2

$\frac{2!\,(3-2-1)!}{3!}$

$\phi_1 = 1$

ϕ_2 :

S	α_S	Δ_S
$\{\}$	$1/3$	0
$\{1\}$	$1/6$	1
$\{3\}$	$1/6$	0
$\{1,3\}$	$1/3$	1

$\phi_2 = 1/2$

ϕ_3 : symmetry / conservation

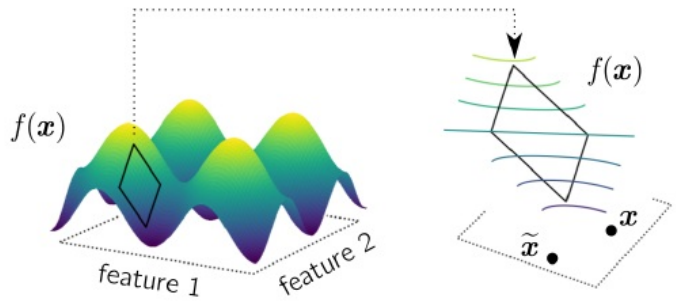
$\phi_3 = 1/2$

S	$f(\mathbf{x}_S)$	$f(\mathbf{x}_{S\cup\{1\}})$	Δ_S
$\emptyset$	$f(0,0,0) = 0$	$f(1,0,0) = 0$	0
$\{2\}$	$f(0,1,0) = 0$	$f(1,1,0) = 1$	1
$\{3\}$	$f(0,0,1) = 0$	$f(1,0,1) = 1$	1
$\{2,3\}$	$f(0,1,1) = 0$	$f(1,1,1) = 2$	2

S	$f(\mathbf{x}_S)$	$f(\mathbf{x}_{S\cup\{2\}})$	Δ_S
$\emptyset$	$f(0,0,0) = 0$	$f(0,1,0) = 0$	0
$\{1\}$	$f(1,0,0) = 0$	$f(1,1,0) = 1$	1
$\{3\}$	$f(0,0,1) = 0$	$f(0,1,1) = 0$	0
$\{1,3\}$	$f(1,0,1) = 1$	$f(1,1,1) = 2$	1

Attribution: Taylor Expansions

- Many ML models $f(\boldsymbol{x})$ are complex and nonlinear when taken globally but are simple and linear when taken locally.



- The function can be approximated locally by some Taylor expansion:

$$f(\boldsymbol{x}) = \underbrace{f(\widetilde{\boldsymbol{x}})}_{=0} + \sum_{i=1}^{d} \underbrace{[\nabla f(\widetilde{\boldsymbol{x}})]_i \cdot (x_i - \widetilde{x}_i)}_{\phi_i} + \ldots$$

- First-order terms ϕ_i of the expansion can serve as an explanation.
- The explanation $(\phi_i)_i$ depends on the choice of root point $\widetilde{\boldsymbol{x}}$.

Attribution: Taylor Expansions

Example: Attribute the prediction $f(\boldsymbol{x}) = \boldsymbol{w}^\top \boldsymbol{x}$ with $\boldsymbol{x} \in \mathbb{R}^d$ on the d input features.

$\tilde{x} = 0$

$$f(x) = \underbrace{f(\tilde{x})}_{=0} + \sum_{i=1}^{d} \underbrace{\underbrace{[\nabla f(\tilde{x})]_i}_{w_i} \cdot \underbrace{(x_i - \tilde{x}_i)}_{x_i}}_{\phi_i = w_i x_i}$$

Attribution: Taylor Expansions

Limitations: Gradient information is too localized.

- Cannot handle *saturation effects* and *discontinuities* e.g. cannot explain the function

$$f(\boldsymbol{x}) = \sum_{i=1}^{d} x_i - \max(0, x_i - \theta)$$

at the point $\boldsymbol{x} = (2, 2)$.

This limitation can be overcome by looking at the structure of the model and decompose the problem of explanation in multiple parts ($\rightarrow$ next slide).

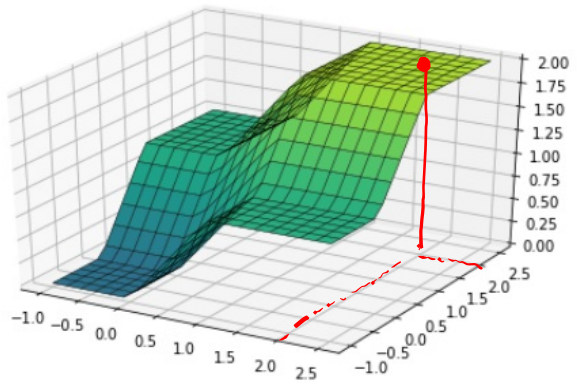
局限性（Limitations）：梯度信息过于局部化

- **无法处理 饱和效应（saturation effects）** 和 **不连续性（discontinuities）**，例如，无法解释如下函数：

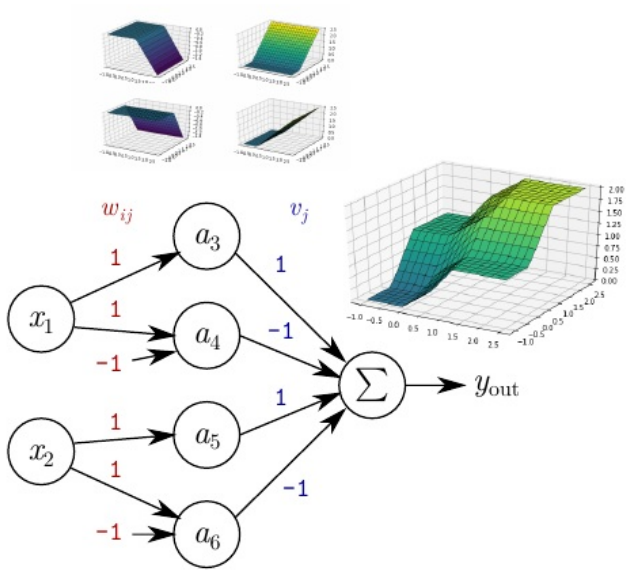
$$f(\mathbf{x}) = \sum_{i=1}^{d} x_i - \max(0, x_i - \theta)$$

在点 $\mathbf{x} = (2, 2)$ 处的行为。

- 这种局限性可以通过**分析模型的结构，并将解释问题分解为多个部分**来克服（→ 下一页）。



Attribution: Look at the Structure of the Model



Observation:

- The function implemented by a ML model is typically a composition of simple elementary functions.
- These functions are simpler to analyze than the whole input-output function.

Idea:

- Treat the problem of explanation as propagating the prediction backward in the input-output graph.
- The layer-wise relevance propagation (LRP) method implements this approach and can be used to explain ML models ($\rightarrow$ next slide).

归因：观察模型的结构

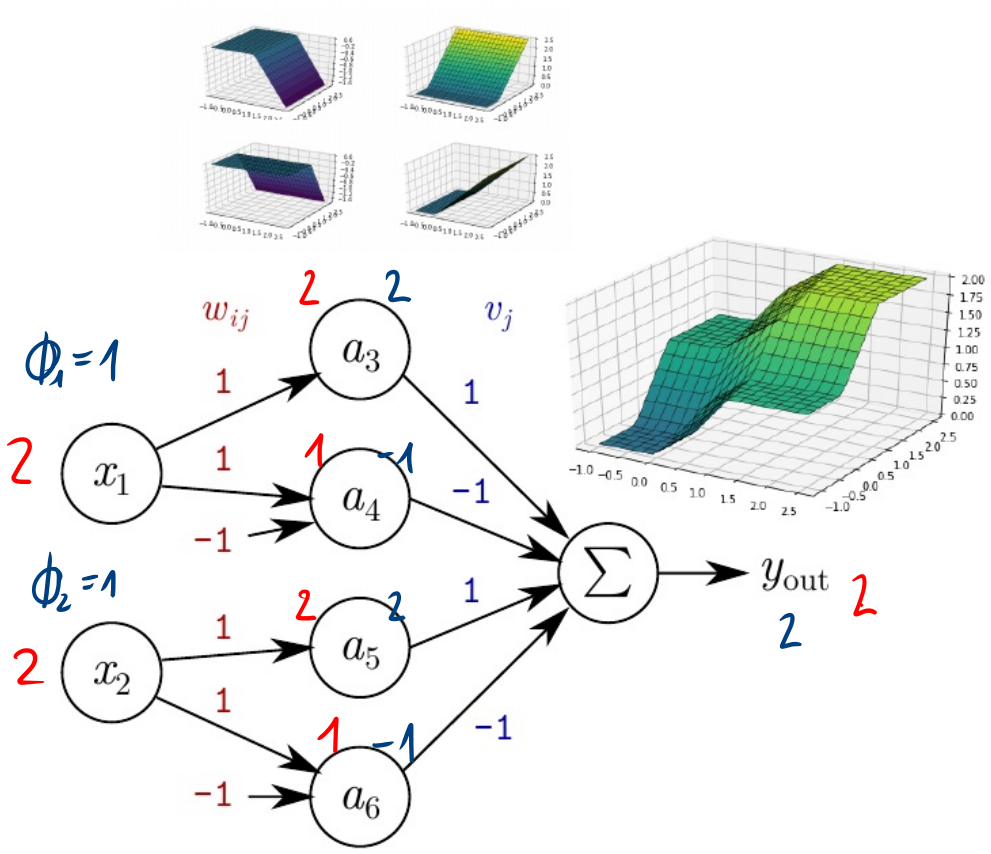
观察（Observation）：

- 机器学习模型实现的函数通常是多个简单基本函数的组合。
- 这些基本函数比整个输入-输出函数更容易分析。

思路（Idea）：

- 将解释问题视为在输入-输出图中向后传播预测值的问题。
- 分层相关传播（Layer-wise Relevance Propagation, LRP）方法可以实现该思路，并用于解释机器学习模型（→ 下一页）。

Attribution: The LRP Method



Example: Consider y_{out} to be the quantity to explain:

$$R_{\text{out}} \leftarrow y_{\text{out}}$$

- **Step 1:** Propagate on the hidden layer

$$\forall_{j=3}^{6} : R_j \leftarrow \frac{a_j v_j}{\sum_{j=3}^{6} a_j v_j} R_{\text{out}}$$

- **Step 2:** Propagate on the first layer

$$\forall_{i=1}^{2} : R_i \leftarrow \sum_{j=3}^{6} \frac{x_i w_{ij}}{\sum_{i=1}^{2} x_i w_{ij}} R_j$$

如何计算 LRP 归因?

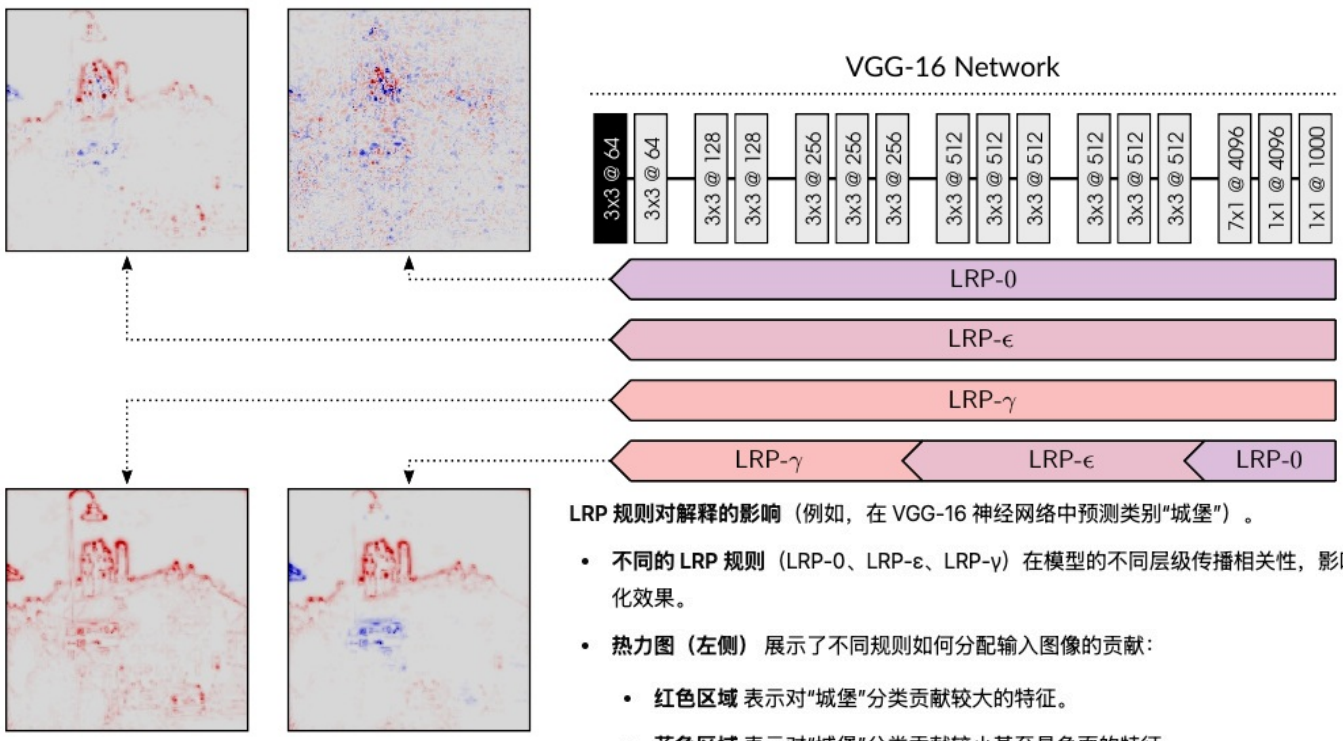
1. **从输出层开始，分配所有预测值的贡献。**
2. **逐层向后传播贡献值**，计算中间神经元的贡献。
3. **最终归因到输入特征**，得到每个特征对预测值的贡献程度。

示例解析

- **在隐藏层（Step 1）**，每个神经元的贡献是按比例分配的，即神经元的激活值 a_j 乘以它的权重 v_j。
- **在输入层（Step 2）**，每个特征的贡献是按它的连接权重 w_{ij} 进行分配的。

Attribution: The LRP Method

Effect of LRP rules on the explanation (e.g. class 'castle' predicted by a VGG-16 neural network.)



LRP 规则对解释的影响（例如，在 VGG-16 神经网络中预测类别"城堡"）。

- **不同的 LRP 规则**（LRP-0、LRP-ε、LRP-γ）在模型的不同层级传播相关性，影响最终解释的可视化效果。
- **热力图（左侧）** 展示了不同规则如何分配输入图像的贡献：
  - **红色区域** 表示对"城堡"分类贡献较大的特征。
  - **蓝色区域** 表示对"城堡"分类贡献较小甚至是负面的特征。

解释

- **VGG-16 网络** 是一个预训练的 CNN（卷积神经网络），用于图像分类。
- **LRP 规则** 允许我们查看**模型如何做出预测**，通过不同的 LRP 规则可以调整解释的粒度：
  - **LRP-0** 直接反向传播。
  - **LRP-ε（带小正则项）** 可以减少噪声，提高稳定性。
  - **LRP-γ（增强重要特征）** 让更重要的特征获得更高的权重。

XAI for Unsupervised Learning

示例：聚类（Clustering）

假设我们使用 **K-means 聚类算法**，我们希望解释某个数据点 x 的**聚类分配（cluster assignment）**。

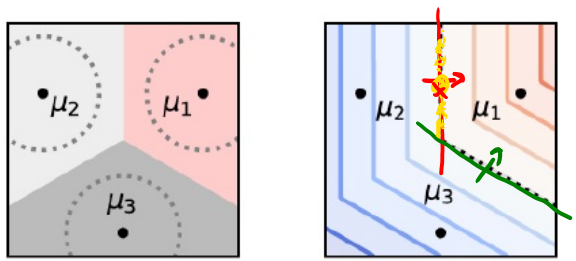
步骤 1

将 K-means 结果转换为等效的神经网络，该神经网络可对 x 进行聚类分配。

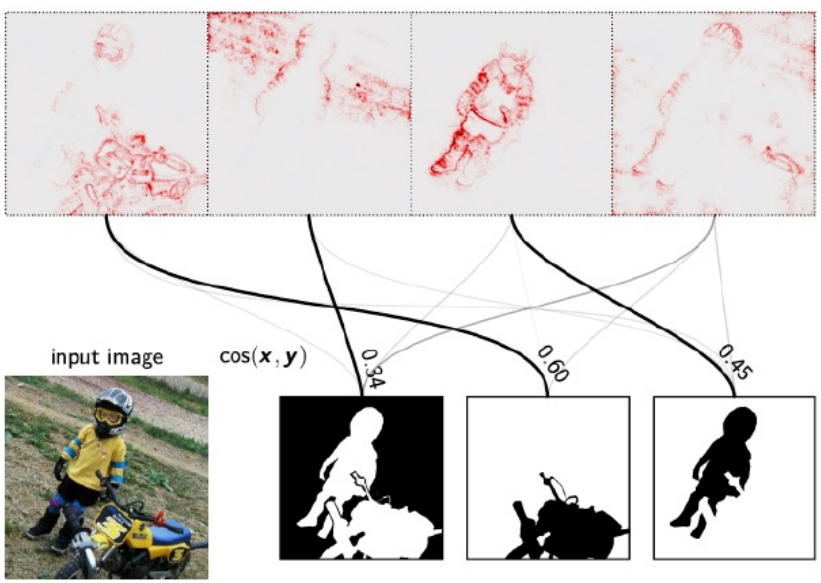
步骤 2

对神经网络应用 LRP（层级相关传播），以解释数据点 x 的聚类归属。

Example Clustering: Consider the K-means algorithm, and we would like to explain the cluster assignment of a data point $\boldsymbol{x}$.



Step 1: Transform the K-means solution into an equivalent neural network that assigns a cluster to $\boldsymbol{x}$.



Step 2: Apply LRP to the neural network to explain the cluster assignment.

XAI for Clustering

Step 1: K-means as a Neural Network:

$$\begin{aligned}
\text{decide cluster } c \quad &\Leftrightarrow \quad \min_{k \neq c} \{ \|\boldsymbol{x} - \boldsymbol{\mu}_k\|^2 \} > \|\boldsymbol{x} - \boldsymbol{\mu}_c\|^2 \\
&\Leftrightarrow \quad \min_{k \neq c} \{ \|\boldsymbol{x} - \boldsymbol{\mu}_k\|^2 - \|\boldsymbol{x} - \boldsymbol{\mu}_c\|^2 \} > 0 \\
&\Leftrightarrow \quad \min_{k \neq c} \{ \underbrace{\boldsymbol{w}_k{}^\top \boldsymbol{x} + b_k}_{h_k} \} > 0
\end{aligned}$$

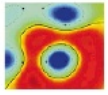
Step 2: LRP applied to the neural network:

$$R_k = \frac{\exp(-\beta h_k)}{\sum_k \exp(-\beta h_k)} R_{\text{out}}$$

$$R_i = \sum_k \frac{w_{ik}(x_i - m_{ik})}{\sum_i w_{ik}(x_i - m_{ik})} R_k$$

Summary

- Explainable AI is an important addition to classical ML models (e.g. for validating a ML model or extracting knowledge from it).
- Many XAI methods have been developed, each of them, with their strengths and limitations:
  - *Activation maximization* can be used to understand what a ML model has learned, but is unsuitable for explaining an individual prediction $f(\boldsymbol{x})$.
  - *Shapley value* has strong theoretical foundations, but is computationally unfeasible for high-dimensional input data.
  - *Taylor expansions* are simple and theoretically founded for simple models, but the expansion does not extrapolate well in complex nonlinear models.
  - *LRP* leverages the structure of the ML model to handle nonlinear decision functions, but requires to carefully choose the propagation rules.



Summary

🔷 **可解释性人工智能（XAI）是传统机器学习模型的重要补充**（例如用于**验证机器学习模型或从中提取知识**）。

🔷 **目前已开发出多种 XAI 方法，每种方法都有其优点和局限性：**

1. **激活最大化（Activation Maximization）**
   - 可以用于理解**机器学习模型学习到了什么**，但**不适用于解释单个预测结果** $f(x)$。
2. **Shapley 值（Shapley Value）**
   - 具有**强大的理论基础**，但在**高维输入数据**情况下计算成本过高，不具备可行性。
3. **泰勒展开（Taylor Expansions）**
   - **对于简单模型**来说，泰勒展开**计算简单且有理论依据**。
   - 但是，该方法**在复杂的非线性模型中**无法很好地推广（外推能力较差）。
4. **层级相关传播（LRP, Layer-wise Relevance Propagation）**
   - **利用机器学习模型的结构**来处理**非线性决策函数**，比梯度方法更稳定。
   - 但需要**谨慎选择传播规则**，否则可能导致解释偏差。