Pattern Recognition

Part 10: Explainable Artificial Intelligence (XAI)

Gerhard Schmidt

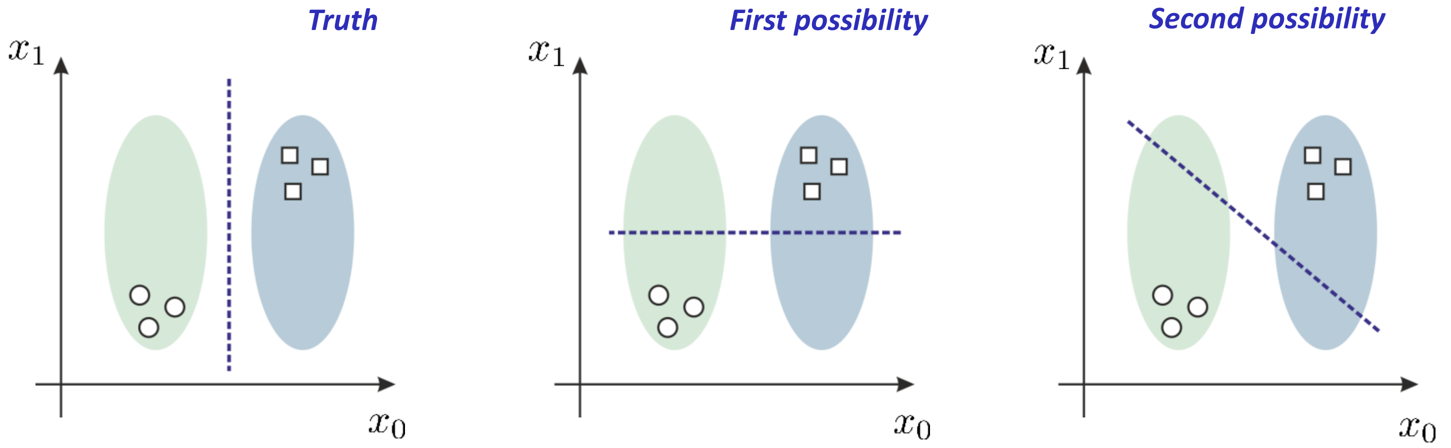
Christian-Albrechts-Universität zu Kiel
Faculty of Engineering
Institute of Electrical and Information Engineering
Digital Signal Processing and System Theory

Christian-Albrechts-Universität zu Kiel

Motivation and Literature

General motivation for XAI:

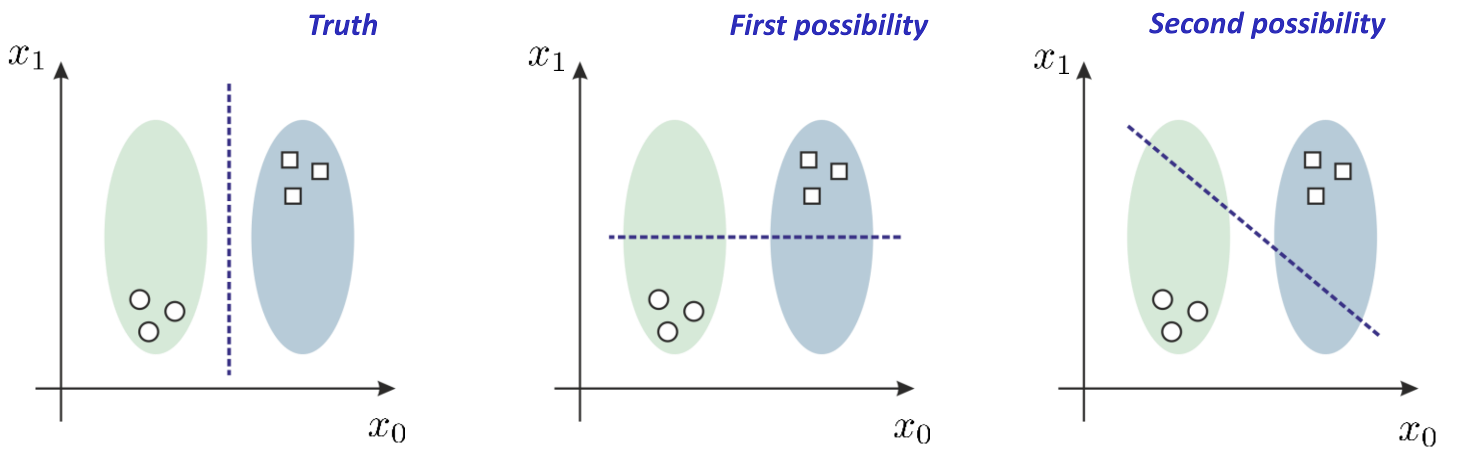
- Are models learning the right relations? How to ***find biases***?
- How ***reliable*** are machine learning models?

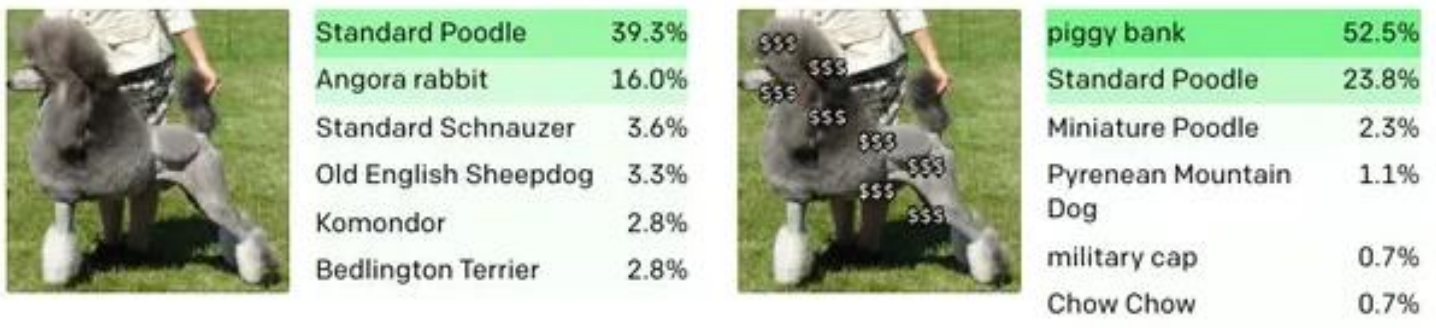


Motivation and Literature

General motivation for XAI:

- Are models learning the right relations? How to ***find biases***?
- How ***reliable*** are machine learning models?
- What are the ***relevant*** information for predictions?





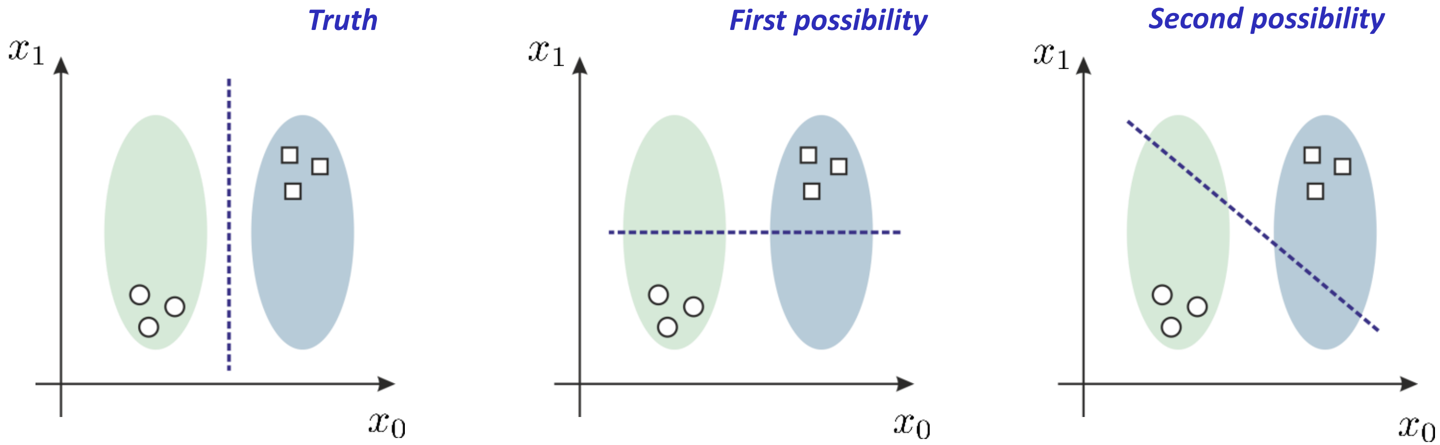
Source: ***Paper Multimodal Neurons in Artificial Neural Networks***

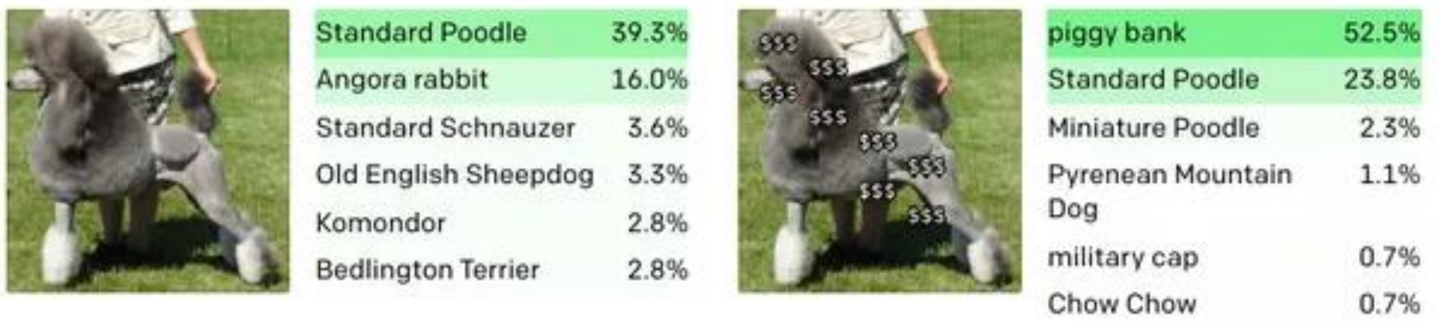
Motivation and Literature

General motivation for XAI:

- Are models learning the right relations? How to ***find biases***?
- How ***reliable*** are machine learning models?
- What are the ***relevant*** information for predictions?
- How to we better understand machine learning and how to ***gain trust***?

➡ Importance for data scientists (validation) as well as for potential users (for example in medicine)





Source: ***Paper Multimodal Neurons in Artificial Neural Networks***

Motivation and Literature

Different approaches for XAI:

Explainable artifical intelligence (XAI)

Model-based approaches

- ***Explainablity*** is already a goal during the generation of the models.

Post-hoc approaches

- It is tried to explain the decisions or estimations ***after the approach is trained***.

Motivation and Literature

Different approaches for XAI:

Explainable artifical intelligence (XAI)

Model-based approaches

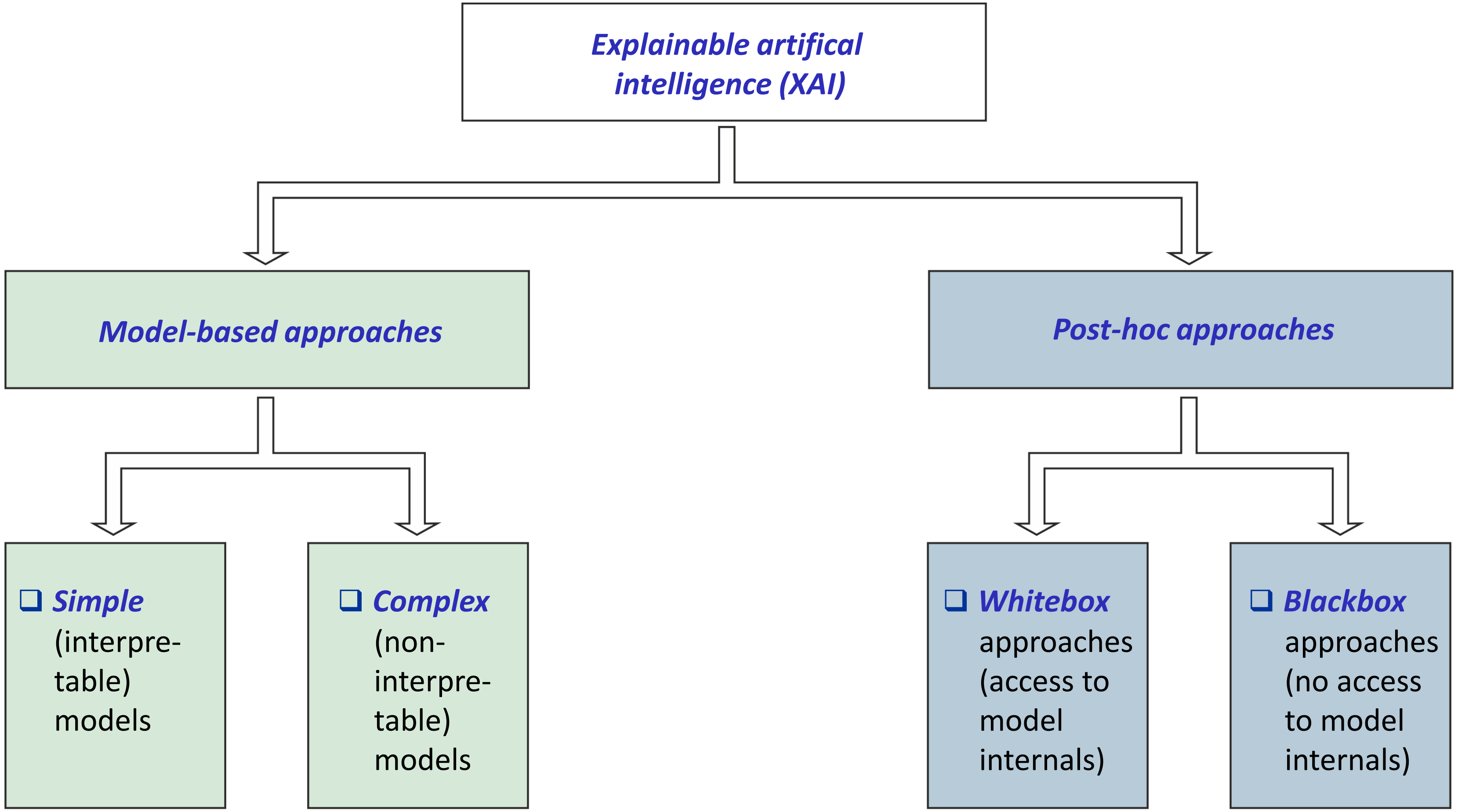
- ***Simple*** (interpre-table) models
- ***Complex*** (non-interpre-table) models

Post-hoc approaches

- It is tried to explain the decisions or estimations ***after the approach is trained***.

Motivation and Literature

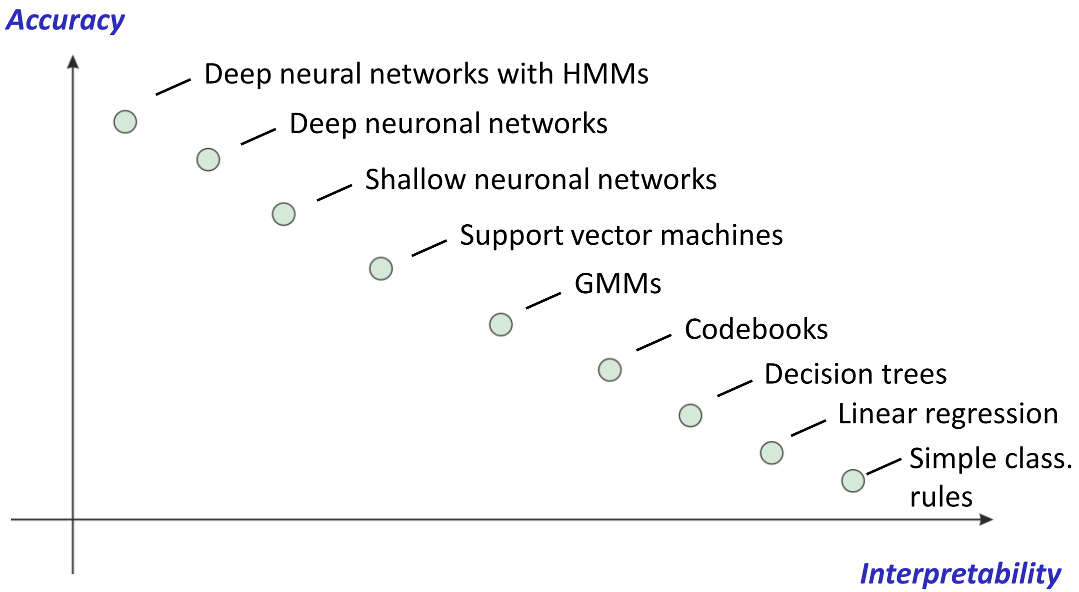
Different approaches for XAI:



Motivation and Literature

Different approaches for XAI:

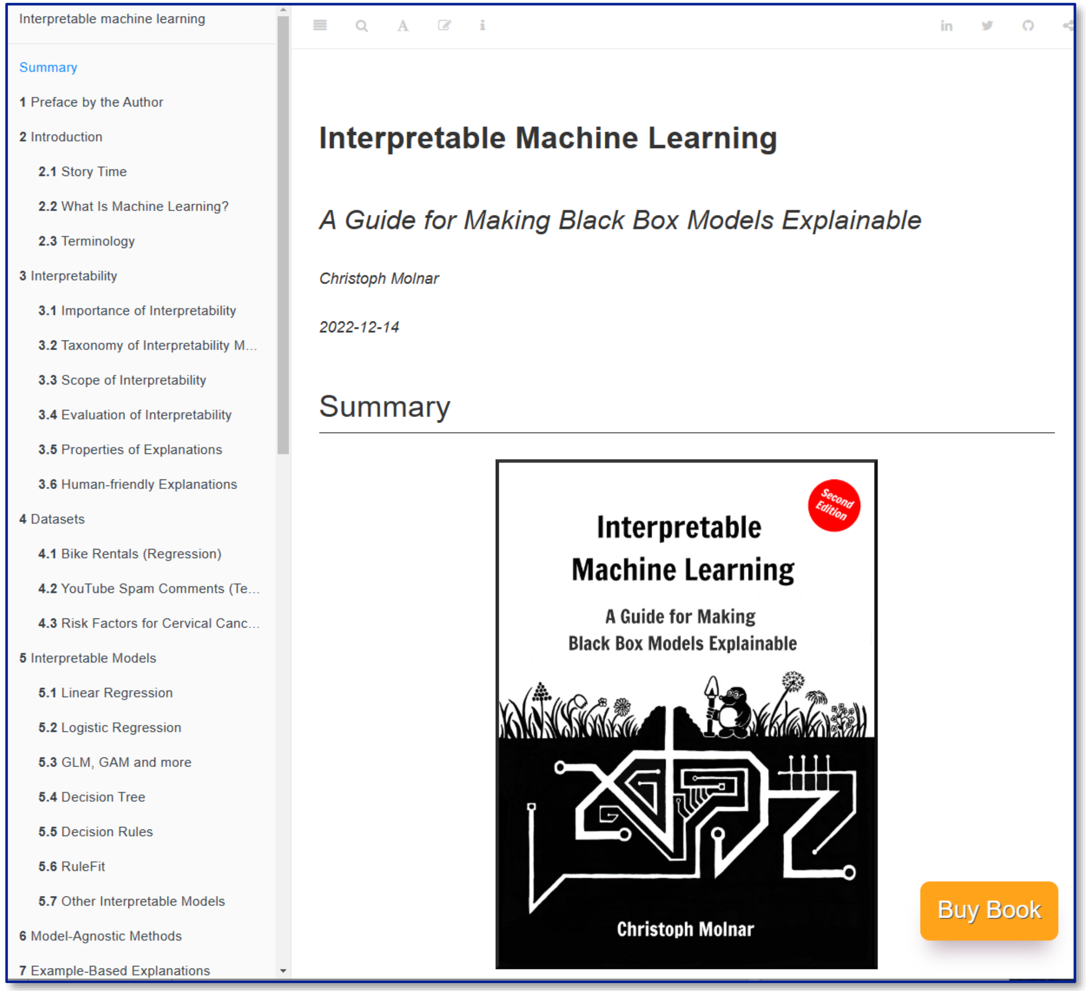
- ***Agnosticity***
  - Model-agnostic: applicable to all machine learning models
  - Model-specific
- ***Scope***
  - Global explanation
  - Local explanation: individual explanations for local areas
- ***Data type***
  - Graphs, images, text/speech, tables, … vectors
- ***Explanation type***
  - Visual, feature importance, data points, surrogate models (model to e.g. explain local decisions of a general model)



Motivation and Literature

Literature:

- C. Molnar: ***Interpretable Machine Learning***, 2022 (available online for free)
- G. Montavon, A. Binder, S. Lapuschkin, W. Samek, K.-R. Müller: ***Layer-Wise Relevance Propagation: An Overview***, 2019
- W. Samek, G. Montavon, S. Lapuschkin, C. J. Anders, K. -R. Müller: ***Explaining Deep Neural Networks and Beyond: A Review of Methods and Applications***, in Proceedings of the IEEE, 2021




Contents

Examples for Glassbox Models – Example Dataset

Example dataset:

- Medical dataset consisting of features such as
  - Age
  - Gender
  - Body mass index (BMI)
  - Amount of cigarettes per day
  - Work type
  - …
- Label
  - Stroke (yes or no)

Examples for Glassbox Models – Example Dataset

Example dataset:

- Medical dataset consisting of features such as
  - Age
  - Gender
  - Body mass index (BMI)
  - Amount of cigarettes per day
  - Work type
  - …
- Label
  - Stroke (yes or no)

- If a new data set is entered and the prediction would be "stroke", questions such as
  - why was this decision made,
  - how would have been the decision if the BMI was decreased by a 1 or 2,
  - How much would be the impact if the person stops smoking

will arise.

Examples for Glassbox Models – Linear Regression

Overview:

- Predicts result as a ***weighted sum of feature inputs***
  - (Affine) Linear relation: Easy interpretable but very limited

Offset *Input features* *Remaining error*

$$y(n) = w_0 + w_1\, x_1(n) + \ldots + w_{N-1}\, x_{N-1}(n) + e(n)$$

Weights

  - Intercept is the model´s prediction without feature inputs
- Various methods for calculation of optimal weights (e.g. least squares)

Pros:

- Good human ***interpretability***

Cons:

- ***Limitation*** by assumption of linearity
- ***Bad performance*** (due to oversimplified reality)
- Fails for classification
  - Output not interpretable as probability
  - No meaningful threshold

Examples for Glassbox Models – Logistic Regression

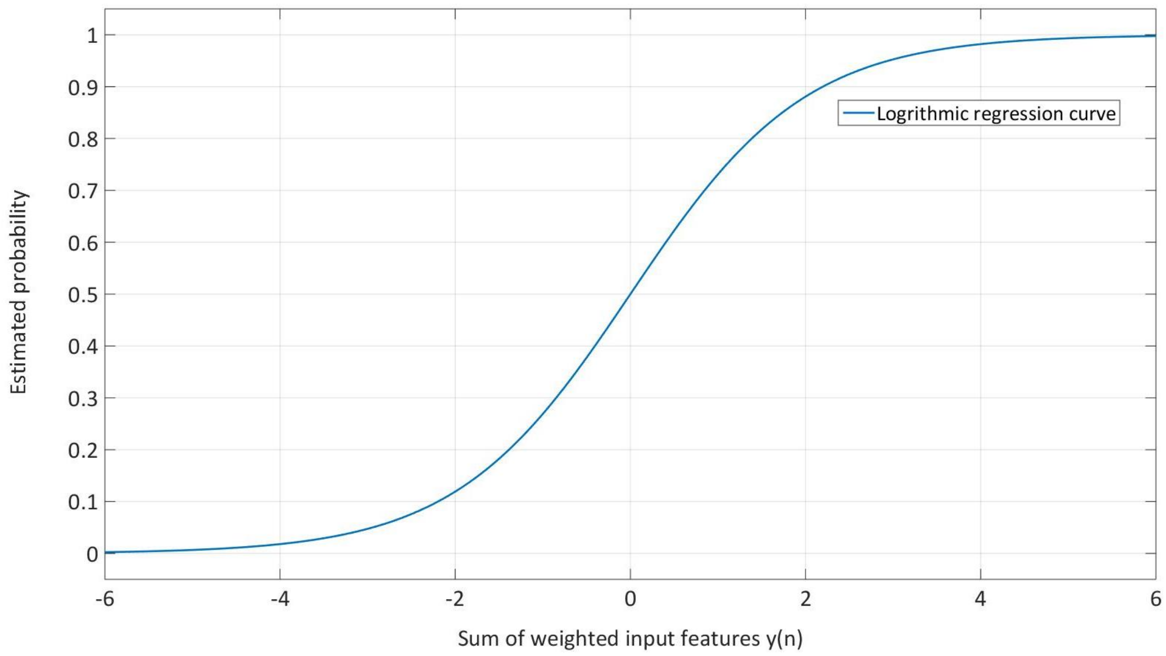
Overview:

- ***Extension*** to linear regression for ***classification problems***
  - Usage of a ***logistic function to squeeze output of a linear equation between 0 and 1***
    - Stroke = 1
    - No stroke = 0

$$f(x) = \frac{1}{1+e^{-x}}$$

  - Connection between linear and logistic regression

$$p(\text{stroke}|\boldsymbol{x}(n)) = \frac{1}{1+e^{-y(n)}} = \frac{1}{1+e^{-w_0 - w_1\,x_1(n) - \ldots - w_{N-1}\,x_{N-1}(n)}}$$



Examples for Glassbox Models – Logistic Regression

Overview:

- ***Extension*** to linear regression for ***classification problems***

$$p\big(\text{stroke}|\boldsymbol{x}(n)\big) = \frac{1}{1+e^{-y(n)}}$$

  - Interpretation of weights:
    - Reformulation of equation above – logarithmic probability ratio

$$\begin{aligned}
R_{\text{prob}}\big(\boldsymbol{x}(n)\big) &= \frac{p\big(\text{stroke}|\boldsymbol{x}(n)\big)}{1-p\big(\text{stroke}|\boldsymbol{x}(n)\big)} \\
&= \frac{\frac{1}{1+e^{-y(n)}}}{1-\frac{1}{1+e^{-y(n)}}} = \frac{\frac{1}{1+e^{-y(n)}}}{\frac{1+e^{-y(n)}-1}{1+e^{-y(n)}}} = \frac{1}{e^{-y(n)}} = e^{y(n)} \\
&= e^{w_0+w_1\,x_1(n)+\ldots+w_{N-1}\,x_{N-1}(n)}
\end{aligned}$$

Examples for Glassbox Models – Logistic Regression

Overview:

- ***Extension*** to linear regression for ***classification problems***

$$p\big(\text{stroke}|\boldsymbol{x}(n)\big) = \frac{1}{1+e^{-y(n)}}$$

  - Probability ratio:

$$R_{\text{prob}}\big(\boldsymbol{x}(n)\big) = \frac{p\big(\text{stroke}|\boldsymbol{x}(n)\big)}{1-p\big(\text{stroke}|\boldsymbol{x}(n)\big)} = e^{w_0+w_1\,x_1(n)+\ldots+w_{N-1}\,x_{N-1}(n)}$$

  - Ratio of ratios if one feature is increased by one:

$$\begin{aligned}\frac{R_{\text{prob}}\big(x_i(n)+1\big)}{R_{\text{prob}}\big(x_i(n)\big)} &= \frac{e^{w_0+w_1\,x_1(n)+\ldots+w_i\,(x_i(n)+1)+\ldots+w_{N-1}\,x_{N-1}(n)}}{e^{w_0+w_1\,x_1(n)+\ldots+w_i\,x_i(n)+\ldots+w_{N-1}\,x_{N-1}(n)}}\\ &= e^{w_0+w_1\,x_1(n)+\ldots+w_i\,(x_i(n)+1)+\ldots+w_{N-1}\,x_{N-1}(n)-w_1\,x_1(n)-\ldots-w_i\,x_i(n)+\ldots-w_{N-1}\,x_{N-1}(n)}\\ &= e^{w_i}\end{aligned}$$

Examples for Glassbox Models – Logistic Regression

Overview:

- ***Extension*** to linear regression for ***classification problems***

$$p\big(\text{stroke}|\boldsymbol{x}(n)\big) = \frac{1}{1+e^{-y(n)}}$$

  - Probability ratio:

$$R_{\text{prob}}\big(\boldsymbol{x}(n)\big) = \frac{p\big(\text{stroke}|\boldsymbol{x}(n)\big)}{1-p\big(\text{stroke}|\boldsymbol{x}(n)\big)} = e^{w_0+w_1\,x_1(n)+\ldots+w_{N-1}\,x_{N-1}(n)}$$

  - Ratio of ratios, if one feature is increased by one:

$$\begin{aligned}\frac{R_{\text{prob}}\big(x_i(n)+1\big)}{R_{\text{prob}}\big(x_i(n)\big)} &= \frac{e^{w_0+w_1\,x_1(n)+\ldots+w_i\,(x_i(n)+1)+\ldots+w_{N-1}\,x_{N-1}(n)}}{e^{w_0+w_1\,x_1(n)+\ldots+w_i\,x_i(n)+\ldots+w_{N-1}\,x_{N-1}(n)}}\\ &= e^{w_0+w_1\,x_1(n)+\ldots+w_i\,(x_i(n)+1)+\ldots+w_{N-1}\,x_{N-1}(n)-w_1\,x_1(n)-\ldots-w_i\,x_i(n)+\ldots-w_{N-1}\,x_{N-1}(n)}\\ &= e^{w_i}\end{aligned}$$

Examples for Glassbox Models – Logistic Regression

Overview:

- Ratio of ratios if one feature is increased by one:

$$R_{\text{prob}}\big(\boldsymbol{x}(n)\big) = \frac{p\big(\text{stroke}|\boldsymbol{x}(n)\big)}{1-p\big(\text{stroke}|\boldsymbol{x}(n)\big)} = e^{w_0+w_1\,x_1(n)+\ldots+w_{N-1}\,x_{N-1}(n)}$$

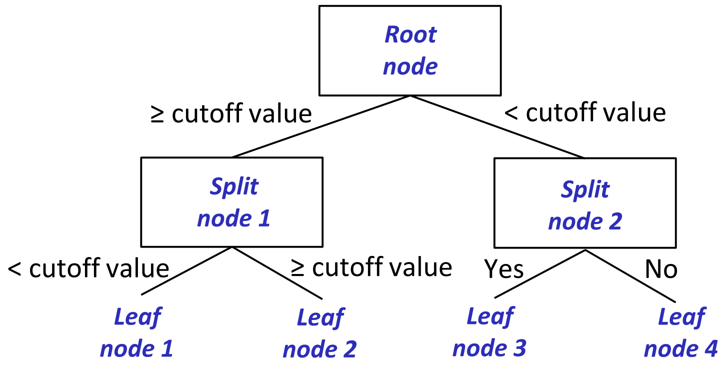
- Increase of one feature by 1 leads to a change of the ratio of the two predictions by

$$\frac{R_{\text{prob}}\big(x_i(n)+1\big)}{R_{\text{prob}}\big(x_i(n)\big)} = e^{w_i}$$

- This means: if the feature $x_i(n)$ is increased by 1, the probability ratio is multiplied by e^{w_i} .

Examples for Glassbox Models – Decision Trees

Overview:



- Decision trees can handle ***non-linear relations***
- ***Depths*** defined by the number of decisions before a leaf node
- Overall importance of a decision by multiplication of all path weights
- Different algorithms to grow a tree
  - Most popular: classification and regression trees (CART)
  - For categorical features: division of data into subsets by grouping
  - Finding the best cutoff per feature and selecting best feature for splitting
  - Search and split recursively until termination criterion is reached

Pros:

- Good human ***interpretability***
  - Natural visualization
  - ***Prediction model***: Changes due to differing inputs predictable
- Trees can capture ***feature interactions***

Cons:

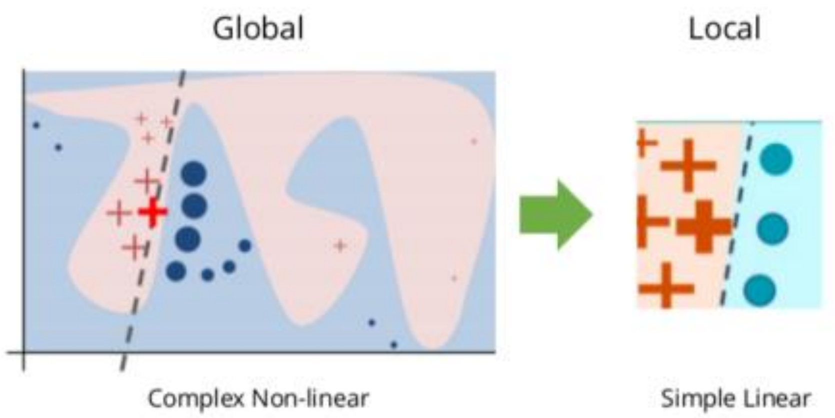
- Fails with linear relations (creates step functions)
- ***Lack of smoothness*** (small changes of input can lead to totally differing decisions)
- ***Unstable*** (slightly different feature sets can lead to totally different decision trees)
- Quickly increasing number of leaf nodes


Contents

Local Interpretable Model-agnostic Explainations (LIME)

General principle:

- ***Explanation of individual instance*** by considering a the ***local region*** ai interpreting the result within this area (local approximation)
- Local ***surrogate model***
- Applicable on black box models
- Applicable on ***many data types***
- Usage of prior knowledge for validation and gaining acceptance
- Local explanation, ***not necessary globally applicable***



From: Marco Tulio Ribeiro, Sameer Singh, Carlos Guestrin, "Why Should I Trust You?: Explaining the Predictions of Any Classifier"

Local interpretable model-agnostic explanations (LIME)

Mathematical expression:

Feature vector (e.g. age, gender, …)

Complex model

Proximity measure

$$exp(\boldsymbol{x}(n)) = \underset{g \subseteq G}{\operatorname{argmin}} \; L(f, g, \pi_{x(n)}) + \Omega(g)$$

Family of interpretable models

Simple model

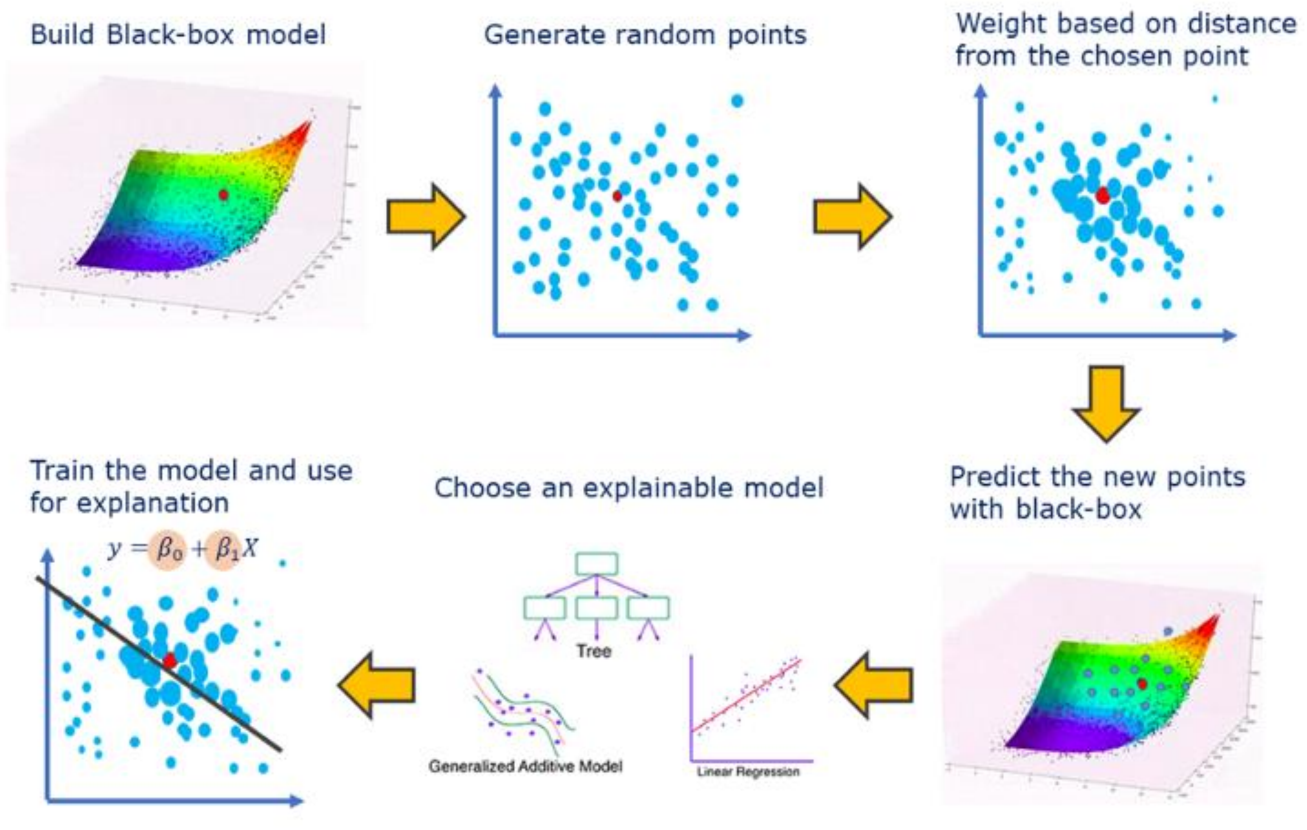
Complexity constraint

- The local explanation for the instance x is the model that ***minimizes loss L***
- The result of the local approximation measures how close the explanation g is to the original model f
- The ***local area*** is defined by the ***proximity measure*** $\pi_{x(n)}$
- The model complexity should be kept low (e.g. less features)
- G is the family of possible explanation models

Local interpretable model-agnostic explanations (LIME)

Principle of LIME algorithm:

- ***Selection of instance***, that should be explained by local approximation
- Generation of ***new data points by perturbation*** and ***prediction*** of black box model for new data set
- ***Weighting*** of new data ***according to their proximity*** to the selected instance
- Training of a weighted and interpretable model on the new dataset
- Explanation of the prediction of the local model



Source: towardsdatascience.com/lime-explain-machine-learning-predictions-af8f18189bfe

Local interpretable model-agnostic explanations (LIME)

Application of LIME:

- Implementation for example with linear regression as surrogate model
- Necessity of ***choosing the number of features in advance*** (tradeoff between interpretability and fidelity)
- Several training methods for training of model with fixed feature number, e.g. Lasso
    - Training of a Lasso model starting with a high regularization parameter λ yielding into no feature weight differing from zero
    - Retrain model while slowly decreasing λ until the determined number of features is reached
- Creation of ***new data points in dependence of the data type***
    - Tabular data: individual perturbation of each feature by variation in statistical properties
    - Text and images: Turn on and off single words or pixels

Local interpretable model-agnostic explanations (LIME)

Pros:

- Surrogate model approach: ***free choice of explanation model*** leads to very high interpretability
- Can be used for tabular data, text and images
- Fidelity measure can be used for an ***impression of the reliability*** of the explanation model
- ***Easy usage*** due to implementation in Python and R
- Explanations of surrogate model can be based on ***other features than the original model***

Cons:

- ***Definition of proximity*** is unsolved for tabular data
- No generic solution for choice of kernels for ***definition of proximity measure***
  - Approach: Testing of different kernel setting until explanation is satisfying
- ***Predefinition of complexity*** (compromise of fidelity and interpretability)
- ***Instability*** of explanations (differing explanations for very close points possible)
- High risk of manipulations to hide biases

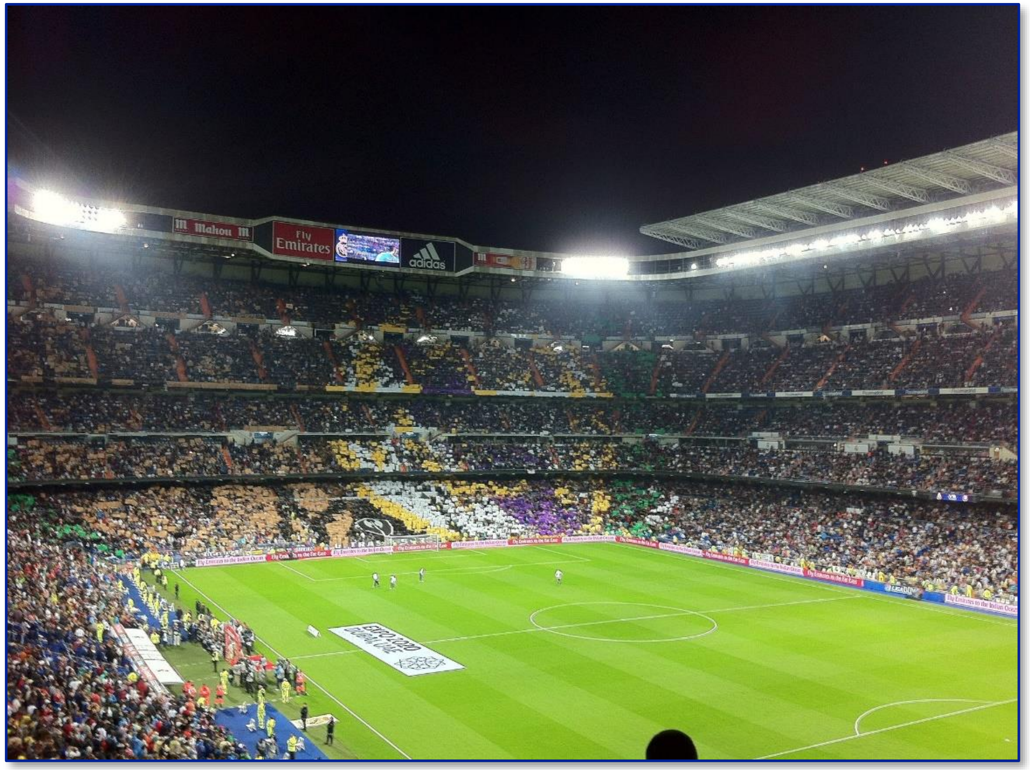

Contents

Shapley Additive Explanation (SHAP)

Origin:

- Based on Shapley values (Shapley, 1953)
- Originally invented for cooperative game theory
    - Divide prize money with respect to the contribution of each team member

- ***Idea: Remove one instance*** (team member, feature, …) and ***simulate the result***
    - Contribution of the instance itself
    - Contribution by joint impact trough relations to other instances
    - Consideration of instances in all possible subsets



Shapley Additive Explanation (SHAP)

Shapley values:

- Shapley value is the ***average of all marginal contributions across all possible coalitions***

Example for interpretation of Shapley values:

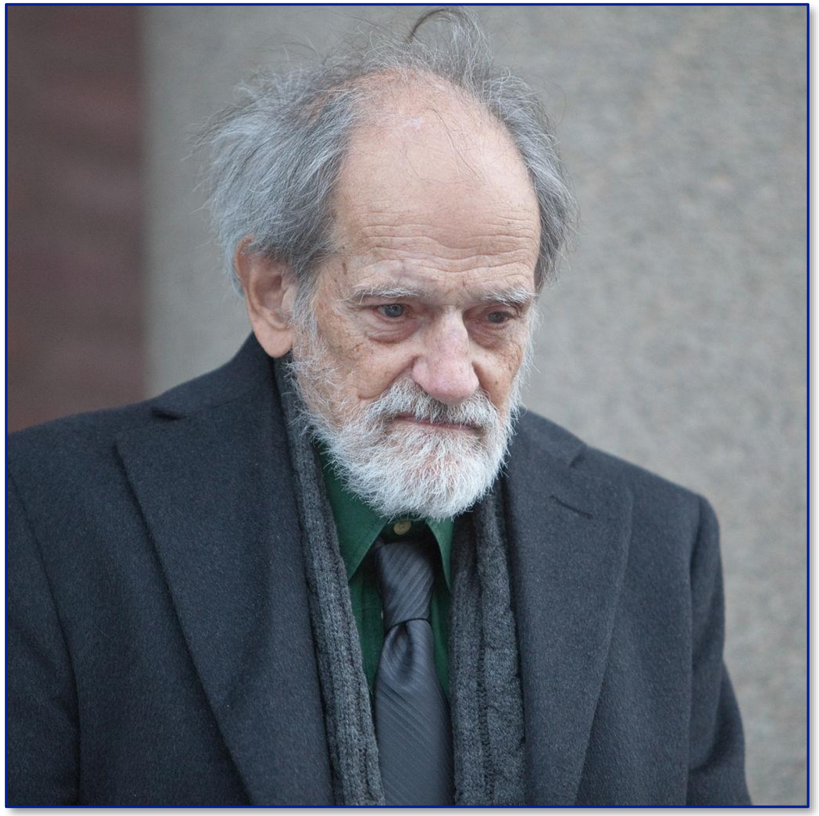
- Fair distribution of the prize money among all player of a soccer team
- Possible coalitions
  - All are playing except player 1
  - All are playing except player 2
  - …
  - All are playing except players 1 and 2
  - All are playing except players 2 and 3
  - …

Shapley Additive Explanation (SHAP)

Example for interpretation of Shapley values:

- Calculate the predicted apartment price with and without a desired feature
- Take the difference for the marginal contribution
- Shapley value is the (weighted) average of marginal contributions

- All Shapley values: complete distribution of the prediction (minus the average) among all features



Lloyd Stowell Shapley (1923 – 2016), Nobel price winner
Source: Wikipedia

Shapley Additive Explanation (SHAP)

Math behind Shapley values:

- ***Linear model prediction***

$$\hat{f}\big(\boldsymbol{x}(n)\big) = \beta_0 + \beta_1\, x_1(n) + \ldots + \beta_p\, x_p(n)$$

- ***Contribution*** of the j-th feature on the prediction

$$\begin{aligned}\phi_j\Big(\hat{f}\big(\boldsymbol{x}(n)\big)\Big) &= \beta_j\, x_j(n) - \mathrm{E}\{\beta_j\, x_j\} \\ &= \beta_j\, x_j(n) - \beta_j\, \mathrm{E}\{x_j\}\end{aligned}$$

- ***Summation*** of all feature contributions for one instance

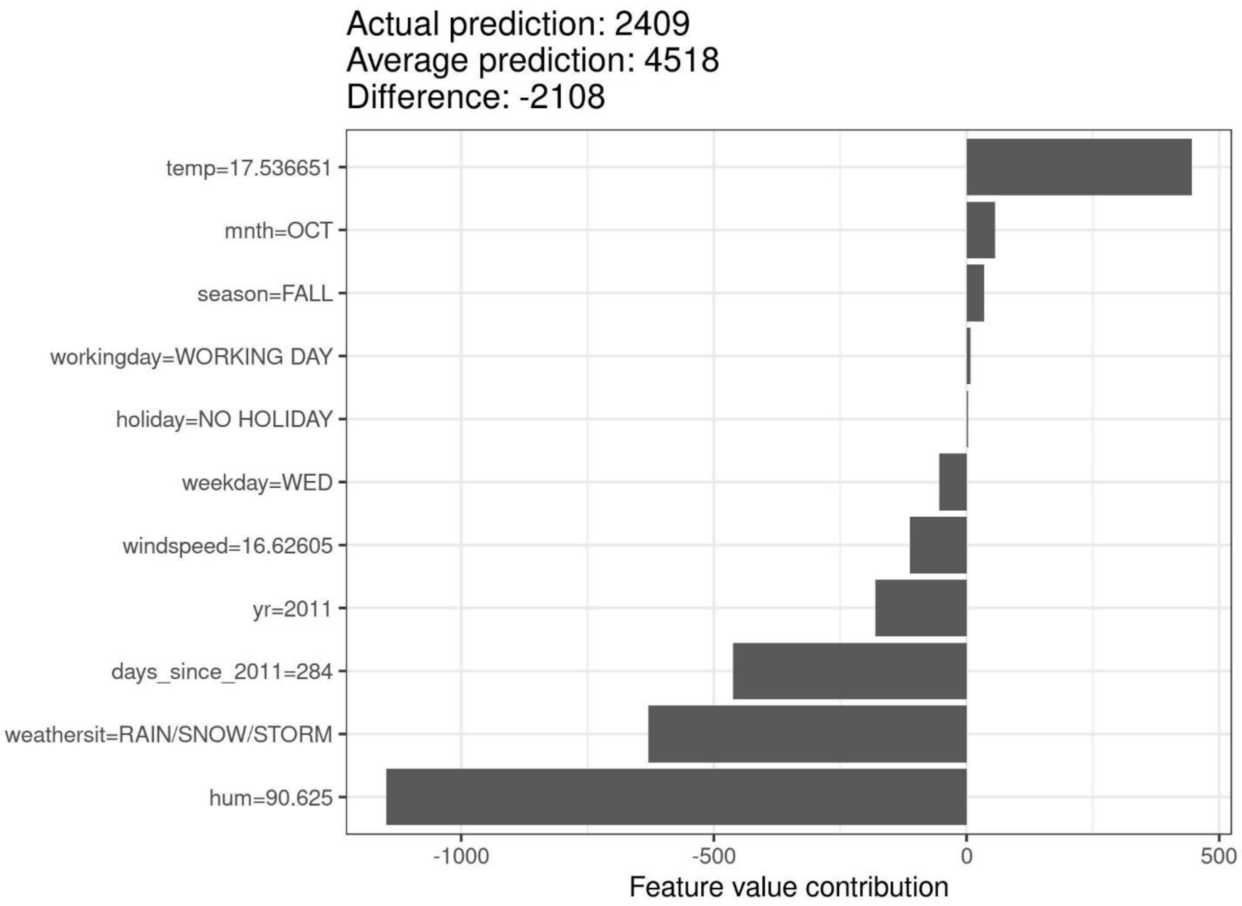
$$\begin{aligned}\sum_{j=1}^{N} \phi_j\big(\hat{f}\big(\boldsymbol{x}(n)\big)\big) &= \sum_{j=1}^{N} \beta_j\, x_j(n) - \mathrm{E}\{\beta_j\, x_j\} \\ &= \hat{f}\big(\boldsymbol{x}(n)\big) - \mathrm{E}\{\hat{f}(\boldsymbol{x})\}\end{aligned}$$

- Feature contributions can be negative
- ***Generalization*** of contribution of j-th feature to all kinds of models

Shapley Additive Explanation (SHAP)

Example for Shapley analysis:

- For the bike rental dataset, we also train a random forest to predict the number of rented bikes for a day, given weather and calendar information. The explanations created for the random forest prediction of a particular day.



Example taken from https://christophm.github.io/interpretable-ml-book/shapley.html

Shapley Additive Explanation (SHAP)

Math behind Shapley values:

- Properties of the Shapley value
  - ***Efficiency***: Contributions must add up to the difference of the prediction for x and the average.
  - ***Symmetry***: If two features contribute equally, the Shapley values should be the same.
  - ***Dummy***: A feature which has no influence at all, has the Shapley value 0.
  - ***Additivity***: For a application with combined features the Shapley values can be added up.
  - Estimation of the Shapley value because of the exponential increase for a increasing feature set
  - Approximation with Monte-Carlo sampling

Shapley Additive Explanation (SHAP)

Pros:

- Prediction is fairly distributed among all features
  - ***Full explanation and solid theory***
  - Almost no assumptions (no validation of assumptions that cause errors)
- Allows contrastive explanations

Cons:

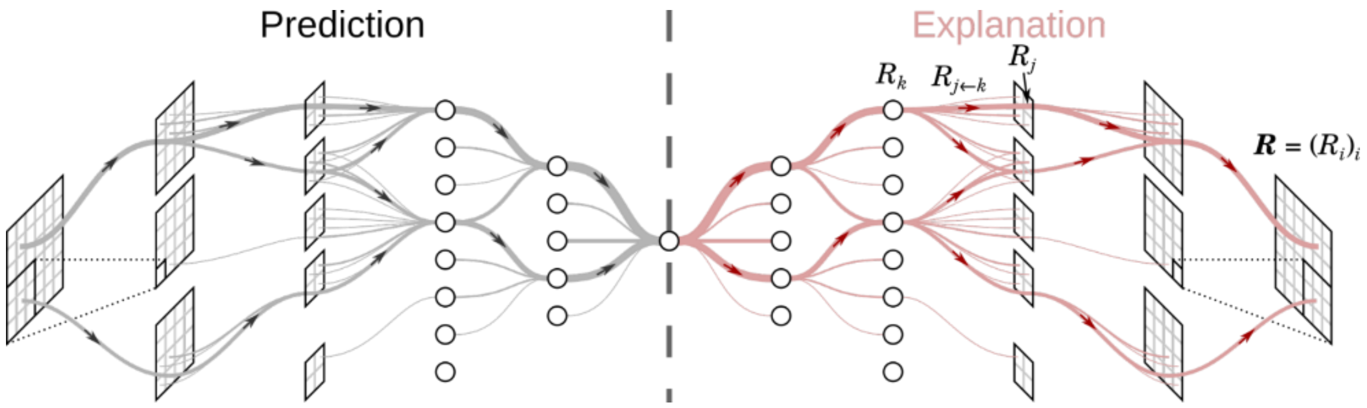
- ***High computation time*** (2^k possible coalitions for k features)
  - Sampling of coalitions to limit complexity leads to increasing variance
  - No rule of thumb for this tradeoff
- ***Misinterpretation*** of Shapley values possible
- Not applicable to sparse explanations
- ***No prediction model*** (No possibility of predicting the output for slight changes of input)
- Inclusion of ***unrealistic*** data instances if features are ***correlated***


Contents

Layer-wise Relevance Propagation (LRP)

Basics:

- Mainly developed to explain NNs and kernel machines (SVMs)
- Explain ***relevance of inputs for prediction*** by layer-wise backpropagation of the model´s output
- Mainly used to ***highlight pixels in images*** that are relevant for the model´s prediction
- Also used for ***videos and text***

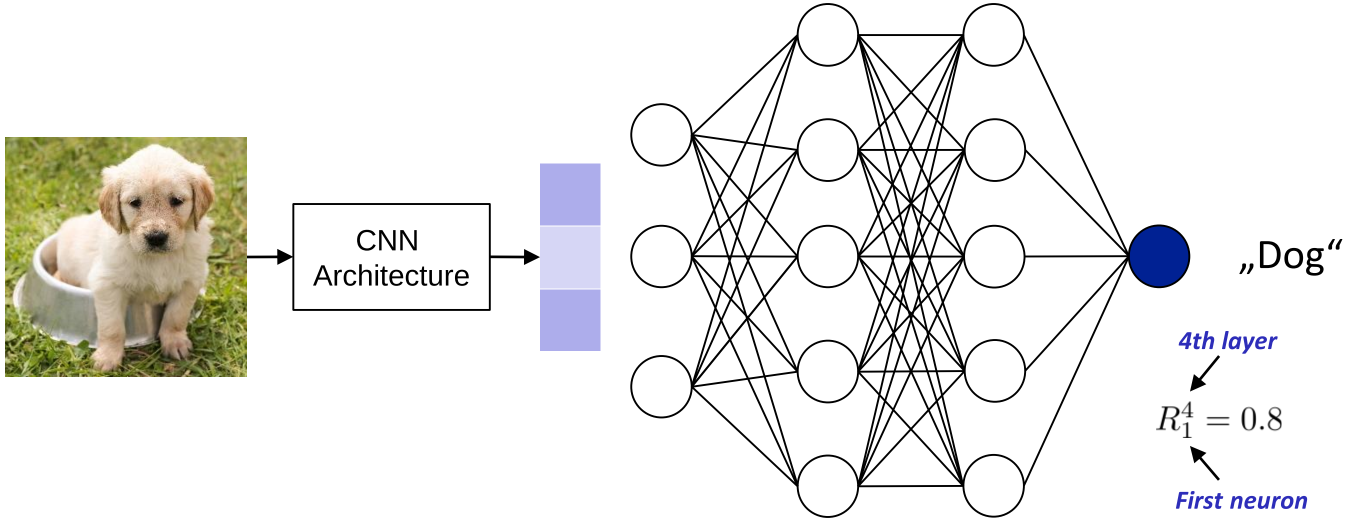


Source: www.hhi.fraunhofer.de/en/departments/...
...ai/technologies-and-solutions/layer-wise-relevance-propagation.html

Layer-wise Relevance Propagation (LRP)

Principle:

- ***Start*** with relevance of ***output*** neuron
- ***Conversion property***
  - What is received by the output neuron must be redistributed to the lower layer in equal amount
  - Analogous to Kirchhoff´s law
- ***Intuitive*** meaning: High values: high relevance for output



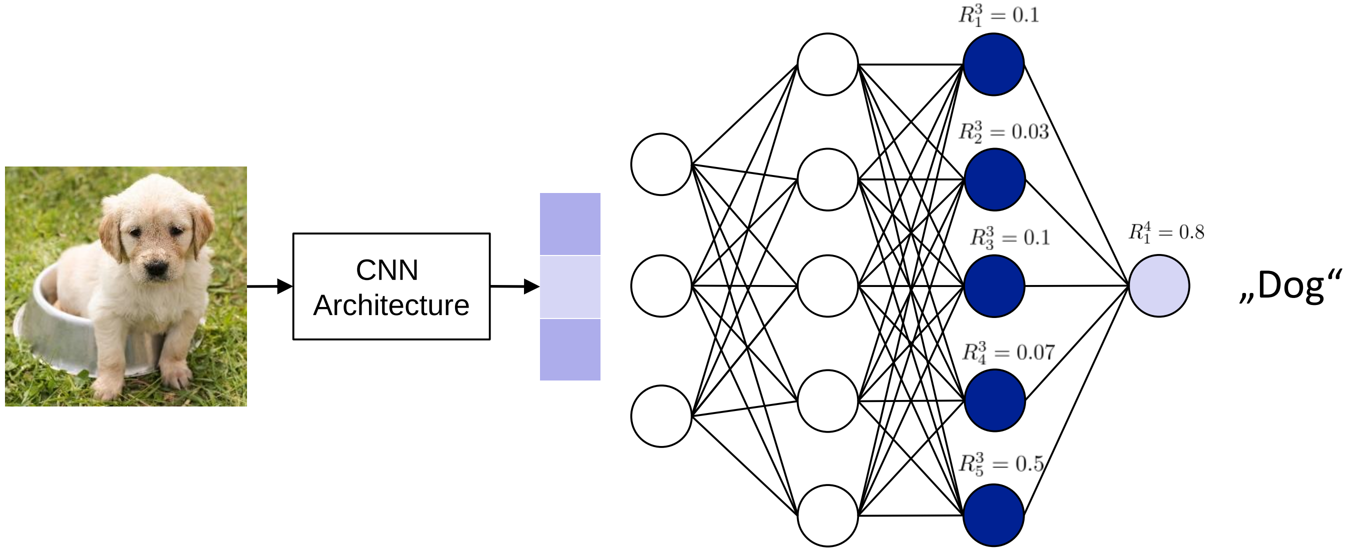
Layer-wise Relevance Propagation (LRP)

Principle:

- Layer-wise backpropagation
  - Calculation of ***relevance Rj of lower layer*** between neurons j and k of the consecutive layers

Activation of neuron *Weight between neurons*

$$R_j = \sum_k \frac{z_{jk}}{\sum_j z_{jk}} R_k \quad \text{with} \quad z_{jk} = a_j w_{jk}$$



  - z_{jk} is the extent that neuron j has contributed to make neuron k relevant
  - Conversion theory: $\sum_j R_j = \sum_k R_k$
  - Each step of propagation procedure can be modeled as an individual Taylor decomposition over the local quantities in the graph
  - ***Termination*** if input features are reached

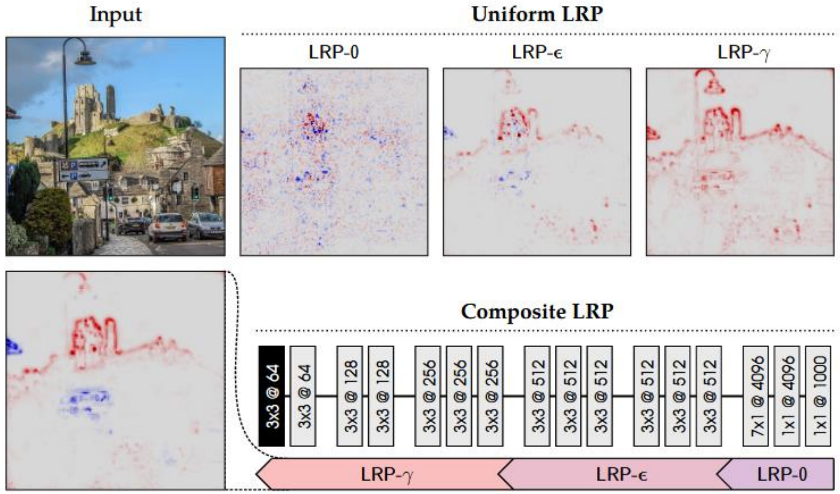
Layer-wise Relevance Propagation (LRP)

LRP rules:

- Different propagation rules with different properties
- ***Basic rule*** (LRP-0): redistribution proportional to the contribution of each input to the neuron activations

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

- Picks many local artifacts of the function
- Overly complex non-focused explanation of the picture
- Used for ***upper layers***
  - Basic rule is close to the function and its gradient
  - Insensitive of entanglements between different classes which are likely for upper classes



Source: www.hhi.fraunhofer.de/en/departments/...
...ai/technologies-and-solutions/layer-wise-relevance-propagation.html

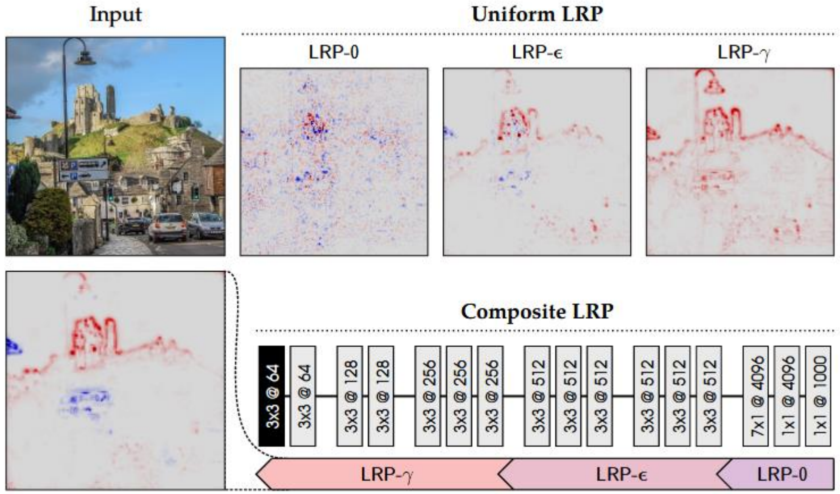
Layer-wise Relevance Propagation (LRP)

LRP rules:

- ***Epsilon rule*** (LRP-ϵ): Extension of basic rule by adding a ***small positive term in the denominator***
  - To absorb some relevance when the contributions of the activation neuron are weak or contradictory

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

- ***Removes noise*** elements and keeps only a limited number of features
- Sparse explanation leads to limited understandability
- Used for ***middle layers***
  - Focus on the most salient explanation factors



Source: www.hhi.fraunhofer.de/en/departments/...
...ai/technologies-and-solutions/layer-wise-relevance-propagation.html

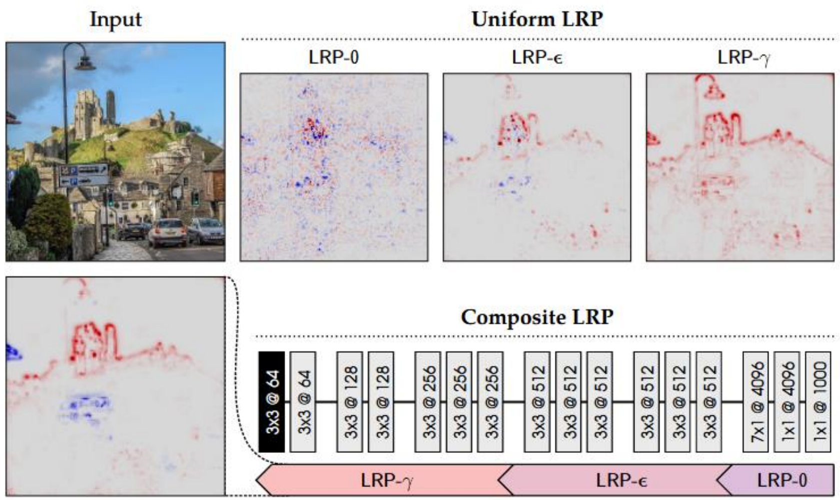
Layer-wise Relevance Propagation (LRP)

LRP rules:

- ***Gamma rule*** (LRP-γ): Enhancement of basic rule by ***favoring the effect of positive contributions*** over negative ones
  - γ controls how much positive contribution is favored
  - More stable explanations because prevalence of positive contributions has a limiting effect of the possibility how much positive and negative relevance can grow while propagation phase

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

- Easily ***understandable*** because of dense highlighting
- But ***inclusion of unrelated concepts*** (less faithful)
- Used in ***lower layers***
  - Very close to relevance map: Requirement of smooth and less noisy explanations



Source: www.hhi.fraunhofer.de/en/departments/...
...ai/technologies-and-solutions/layer-wise-relevance-propagation.html

More propagation rules possible...

Layer-wise Relevance Propagation (LRP)

Examples:

- What animal is depicted?

1. Wähle ein Bild

2. Stelle eine Frage

Welches Tier ist abgebildet

3. Die KI antwortet:

Hund (98%)
Welpe (1%)
Hunde (0%)

4. Im Bild markierte Stellen waren für die die Antwort entscheidend und ausgeblendete Bereiche waren nicht relevant

Die VQA-Berechnung dauerte 1.852 Sekunden

Layer-wise Relevance Propagation (LRP)

Examples:

- What time of year is right now?

Fraunhofer HHI BIFOLD

1. Wähle ein Bild

2. Stelle eine Frage

Welche Jahreszeit ist gerade

3. Die KI antwortet:
Sommer (40%)
Frühling (34%)
Herbst (17%)

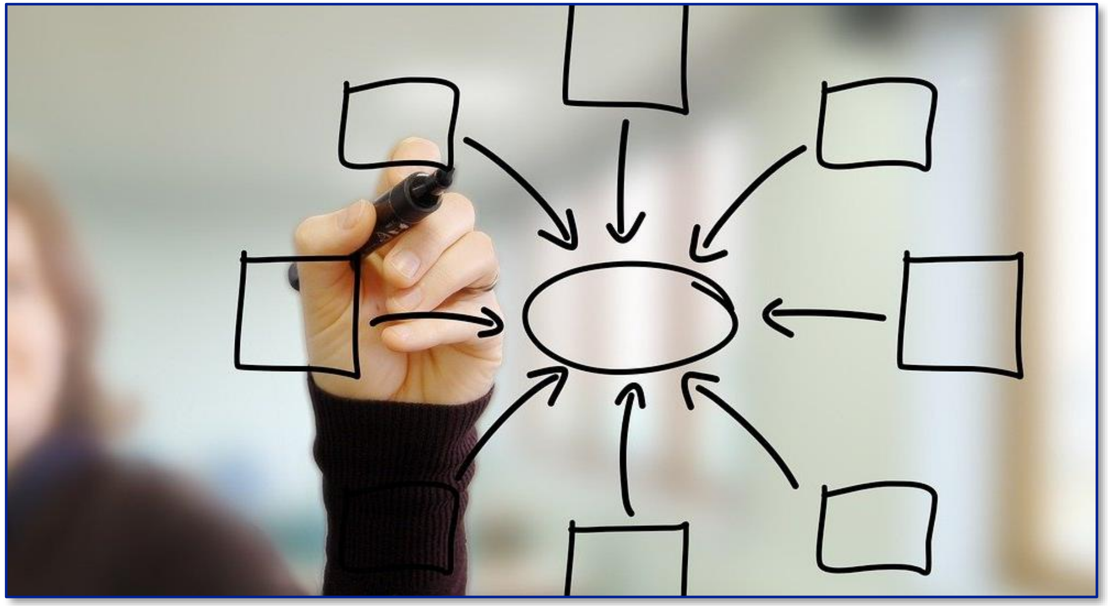
4. Im Bild markierte Stellen waren für die die Antwort entscheidend und ausgeblendete Bereiche waren nicht relevant

Die VQA-Berechnung dauerte 2.325 Sekunden

Summary – Aspects for and against XAI Methods

Aspects of decision between XAI methods:

- Faithfulness and traceability (critical applications)
- A priori knowledge
- Favored explanation method/output model (visual, text, etc.)
    - Interpretability vs. fidelity of explanation
- Applicable to the input data
- Complexity of input data



Summary and Outlook

Summary:

- Motivation
- Glassbox models
- Local interpretable model-agnostic explanations (LIME)
- Shapley additive explanations (SHAP)
- Layer-wise relevance propagation (LRP)

That's it:

- Thanks for listening /attending the lecture.