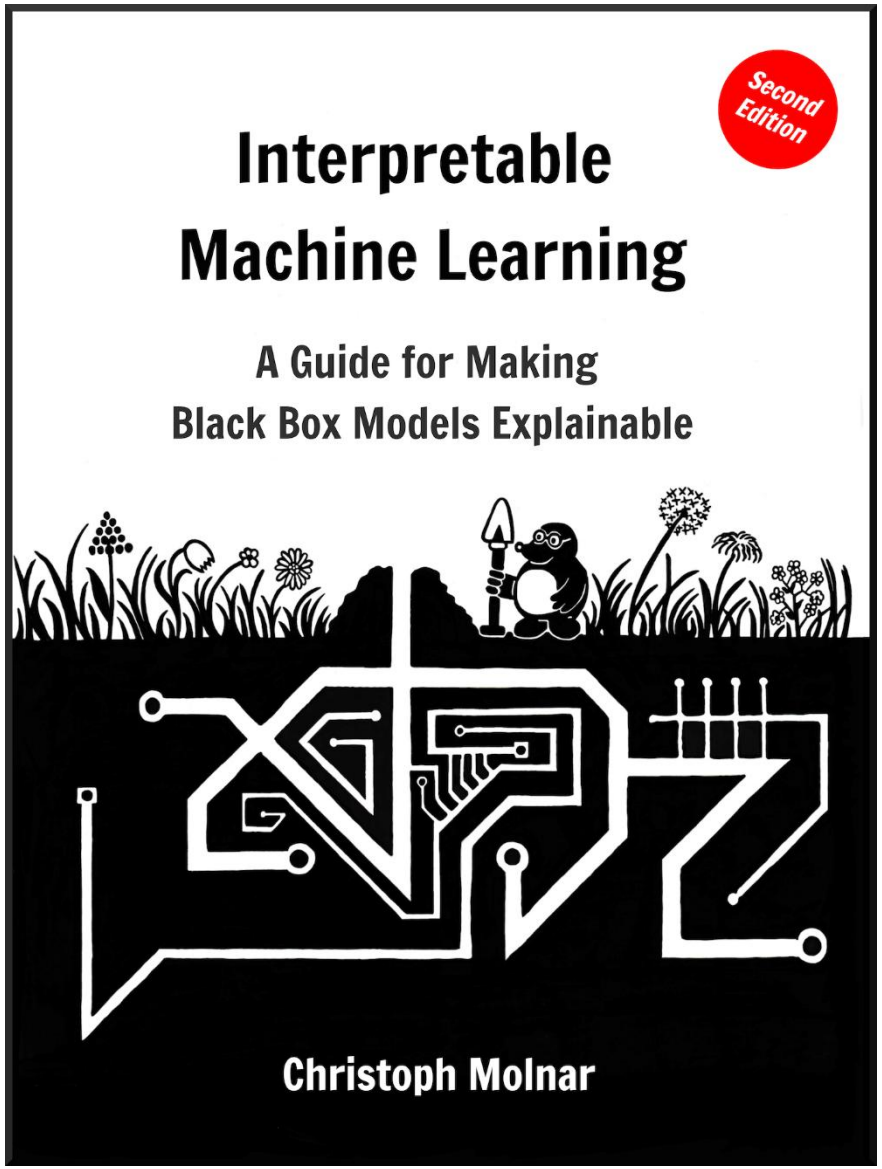
Interpretable Machine Learning

Nov 15 2025

Giảng viên: TS. Lưu Phúc Lợi
Luu.p.loi@googlemail.com
Zalo: 0901802182

Content

- What is Machine Learning (ML) Model Interpretation?
- Why is ML Model Interpretability Importance?
- Interpretation Goals
- ML Model performance vs. Interpretability
- Dimensions of Interpretability
- Feature Interactions
- Hands-on Exercises



https://christophm.github.io/interpretable-ml-book/

What is ML Model Interpretation?

- ML model interpretability tries to understand and explain the steps and decisions a machine learning model takes when making predictions.
- It gives us the ability to question the model's decision and learn about the following aspects.
  - What features/attributes are important to the model?

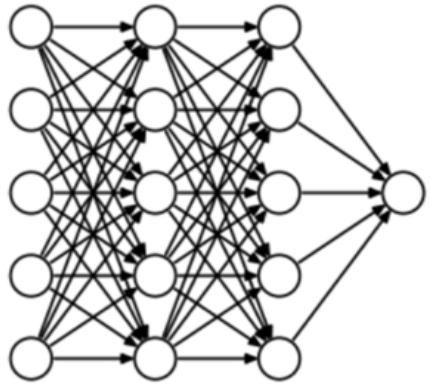
    → You should be able to extract information about what features are important and how features interact to create powerful information.
  - Why did the model come to this conclusion?

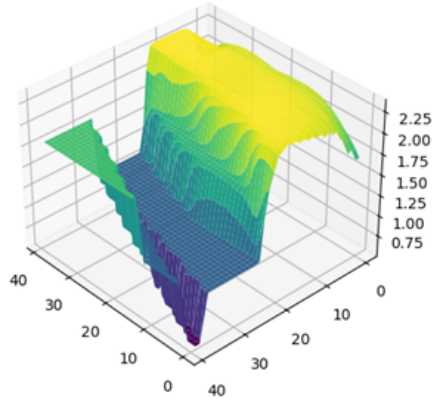
    → You should also have the ability to extract information about specific predictions to validate and justify why the model produced a certain result.

https://www.markovml.com/blog/model-interpreting

WHY INTERPRETABILITY?

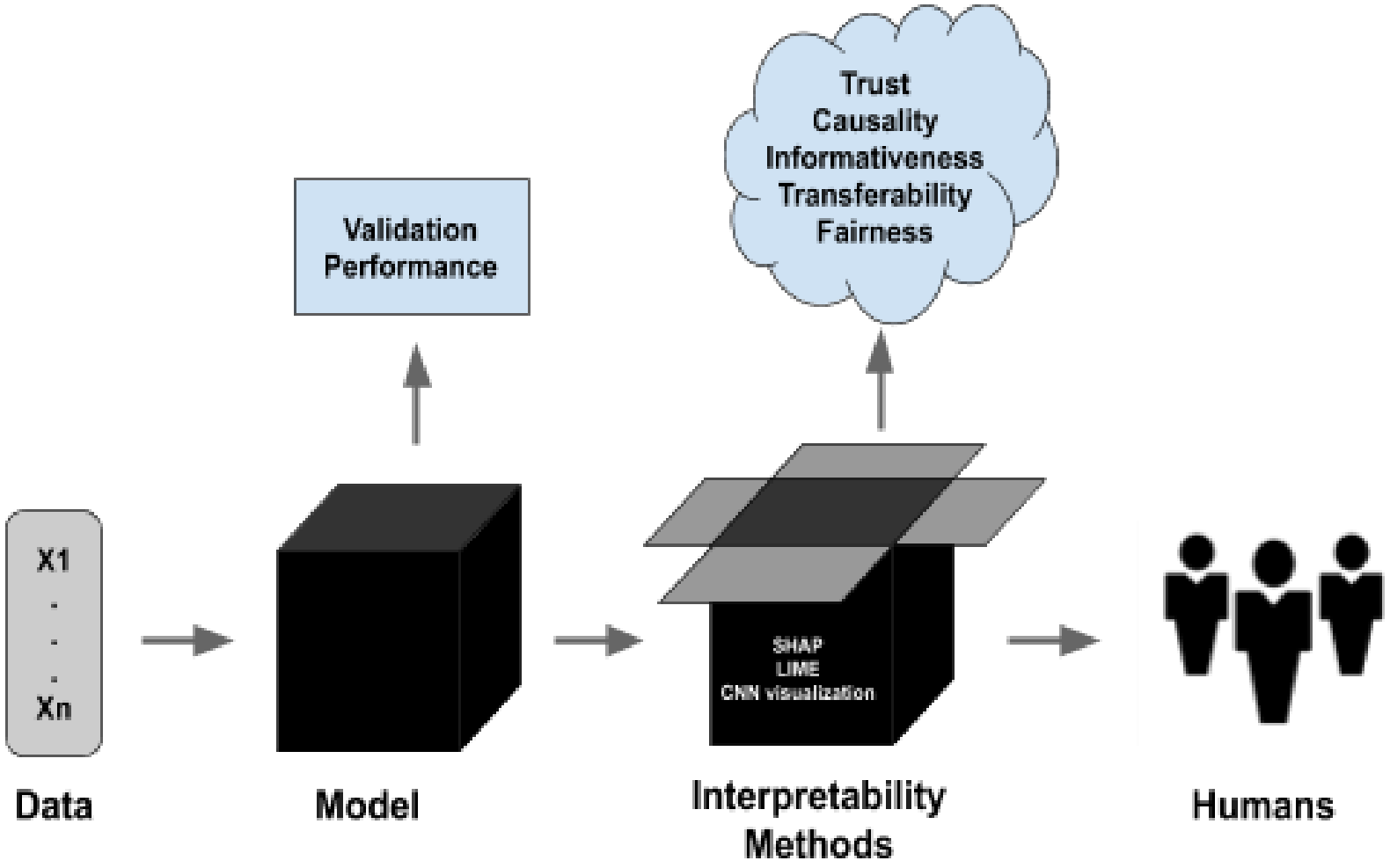
- ML: huge potential to aid decision-making process due to its predictive performance
- ML models are black boxes, e.g., XGBoost, RBF SVM or DNNs
  $\rightsquigarrow$ too complex to be understood by humans
- Some applications are "learn to understand"
- When deploying ML models, lack of explanations
  1. hurts trust
  2. creates barriers





https://slds-lmu.github.io/iml/chapters/

ML Model Interpretability Importance

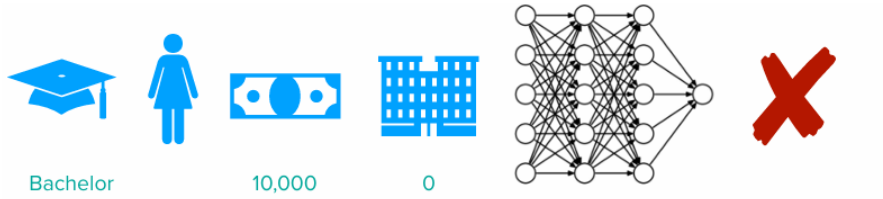


https://www.markovml.com/blog/model-interpreting

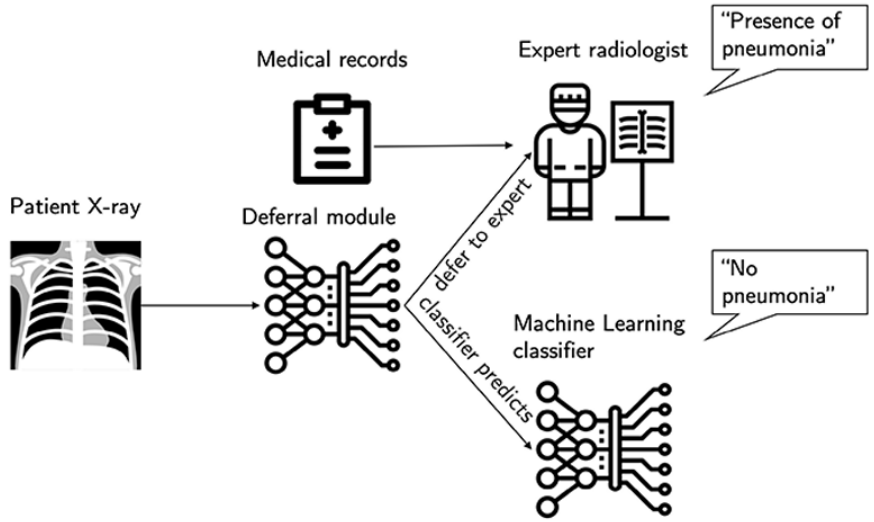
INTERPRETABILITY IN HIGH-STAKES DECISIONS

Examples of critical areas where decisions based on ML models can affect human life

- Credit scoring and insurance applications
  - ▸ Society of Actuaries
    - Reasons for not granting a loan
    - Fraud detection in insurance claims



- Medical applications
  - Identification of diseases
  - Recommendations of treatments
- ...



▸ Miliard (2020)

https://slds-lmu.github.io/iml/chapters/

INTERPRETABILITY IN HIGH-STAKES DECISIONS

Need for interpretability becoming increasingly important from a legal perspective

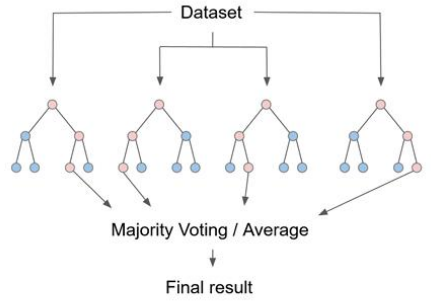
- General Data Protection Regulation (GDPR) requires for some applications that models have to be explainable ▸ Goodman & Flaxman (2017)
  ⇝ *EU Regulations on Algorithmic Decision-Making and a "Right to Explanation"*
- *Ethics guidelines for trustworthy AI* ▸ European Commission (2019)

BRIEF HISTORY OF INTERPRETABILITY

- 18th and 19th century:
  linear regression models (Gauss, Legendre, Quetelet)
- 1940s:
  emergence of sensitivity analysis (SA)
- Middle of 20th century:
  Rule-based ML, incl. decision rules and decision trees
- 2001:
  built-in feature importance measure of random forests
- >2010:
  Explainable AI (XAI) for deep learning
- >2015:
  IML as an independent field of research

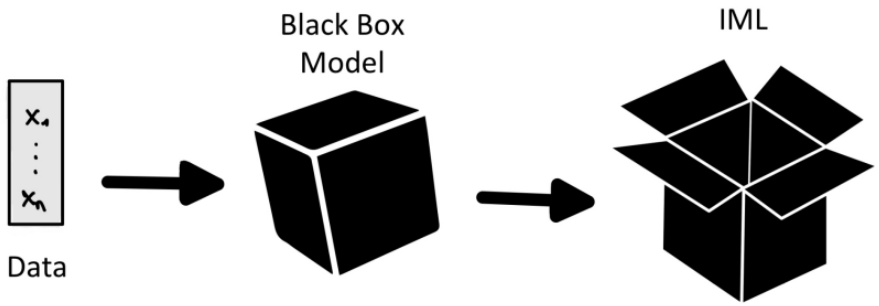
▸ Carl Friedrich Gauss

▸ Wikipedia



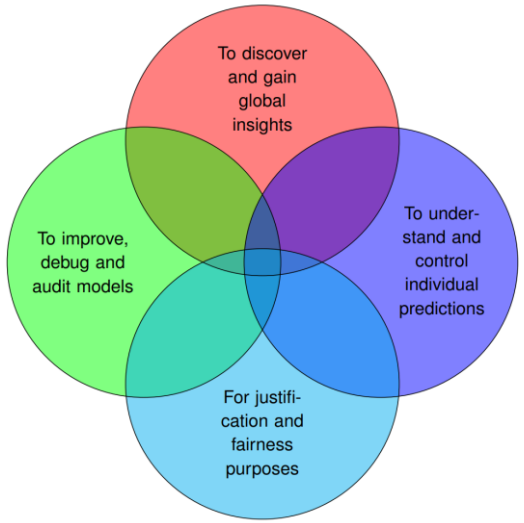
https://slds-lmu.github.io/iml/chapters/

Interpretation Goals



Understand Interpretation Goals:

- Global insights (discovery)
- Improve model (debug and audit)
- Understand and control individual predictions
- Justification and fairness



A related presentation can be found in ▸ Adadi and Berrada 2018.

ML Model Interpretability Goals

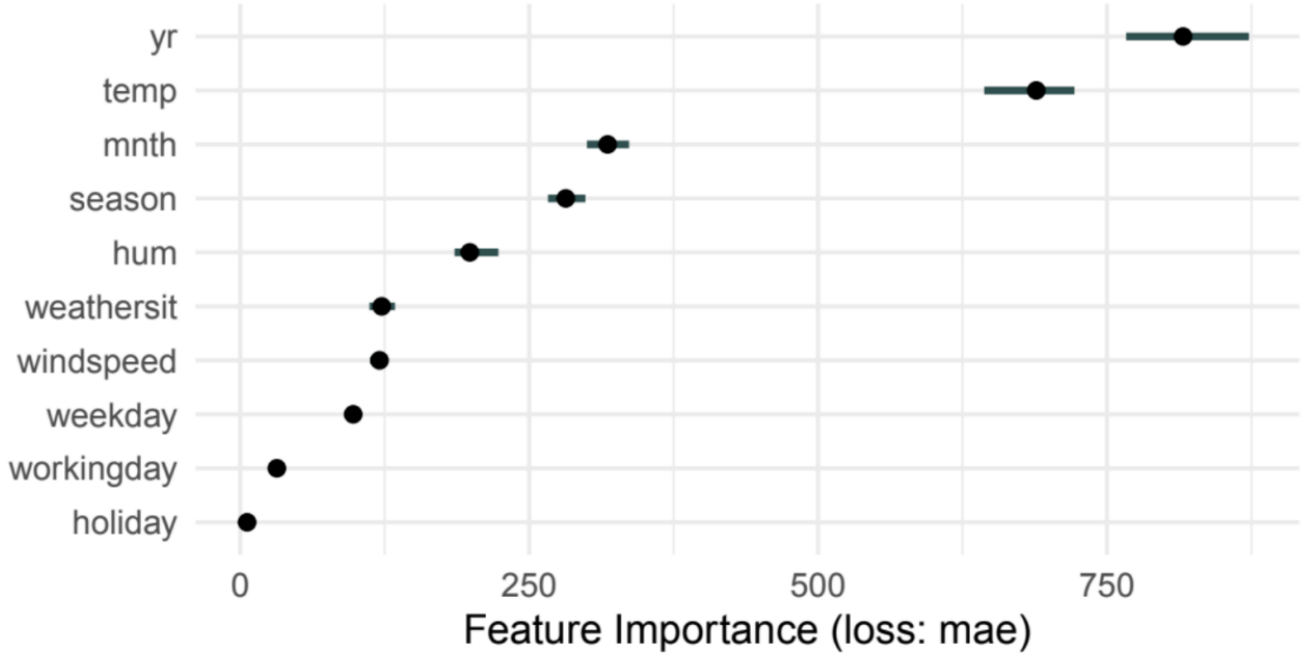
- **Fairness** – An interpretable model used by a company to decide raises and promotions can tell you exactly why any particular person was, or wasn't offered a promotion.
- **Reliability** – Small changes in input won't lead to a domino effect and alter the output drastically.
- **Causality** – Only causal relationships are useful for decision making.
- **Trust** – It's easier for all project stakeholders, especially on the non-technical side, to trust a model that can be explained in layman's terms.

https://neptune.ai/blog/ml-model-interpretation-tools

DISCOVER AND GAIN GLOBAL INSIGHTS

⇝ Gain insights about data, distribution and model

Example: Bike Sharing Dataset (predict number of bike rentals per day)
Exemplary question: Which feature influences the model performance and to what extent?



- Year (`yr`) and Temperature (temp) most important features
- Holiday (`holiday`) less important (Can we drop it?)

https://slds-lmu.github.io/iml/chapters/

IMPROVE, DEBUG AND AUDIT MODELS

⇝ Insights help to identify flaws (in data or model), which can be corrected

Example: Neural Net Tank ▸ gwern.net

A cautionary tale (never actually happened):

- Train a neural network to detect tanks
- Good fit on training data
- Application outside training data: failure
- Reasons vary depending on input
  ⇝ NN based decision on irrelevant points
- E.g. model detects weather based on sky:
  ⇝ All photos with tanks show cloudy sky
  ⇝ Photos without tanks show sunny sky

https://slds-lmu.github.io/iml/chapters/

IMPROVE, DEBUG AND AUDIT MODELS

⇝ Insights help to identify flaws (in data or model), which can be corrected

Comment on tank example:

"We made exactly the same mistake in one of my projects on insect recognition. We photographed 54 classes of insects. Specimens had been collected, identified, and placed in vials. Vials were placed in boxes sorted by class. I hired student workers to photograph the specimens. Naturally they did this one box at a time; hence, one class at a time. Photos were taken in alcohol. Bubbles would form in the alcohol. Different bubbles on different days. The learned classifier was surprisingly good. But a saliency map revealed that it was reading the bubble patterns and ignoring the specimens. I was so embarrassed that I had made the oldest mistake in the book (even if it was apocryphal). Unbelievable. Lesson: always randomize even if you don't know what you are controlling for!"

▸ Thomas G. Dietterich

https://slds-lmu.github.io/iml/chapters/

IMPROVE, DEBUG AND AUDIT MODELS

- Nearly all computer programs have bugs
  - ⇝ Minimizing such bugs extremely relevant
- Process with multiple steps to locate, understand and solve a problem
  - ⇝ Classical debugging
- **In ML** we have a program (learner) writing another program (model)
- How to debug or audit programs which contain ML models?
- Based on a single cross-val score?

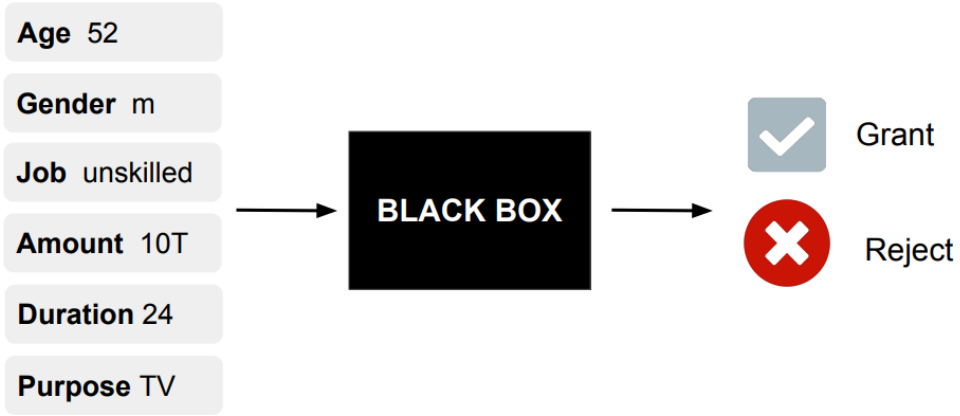
⇝ Being able to interpret your model will always be helpful – if possible!

https://slds-lmu.github.io/iml/chapters/

UNDERSTAND & CONTROL INDIVIDUAL PREDICTIONS

⇝ Explaining individual decisions can prevent unwanted actions based on the model

Example: Credit Risk Application
x: customer and credit information; *y*: grant or reject credit



Questions:

- Why was the credit rejected?
- Is it a fair decision?
- **How should x be changed so that the credit is accepted?**

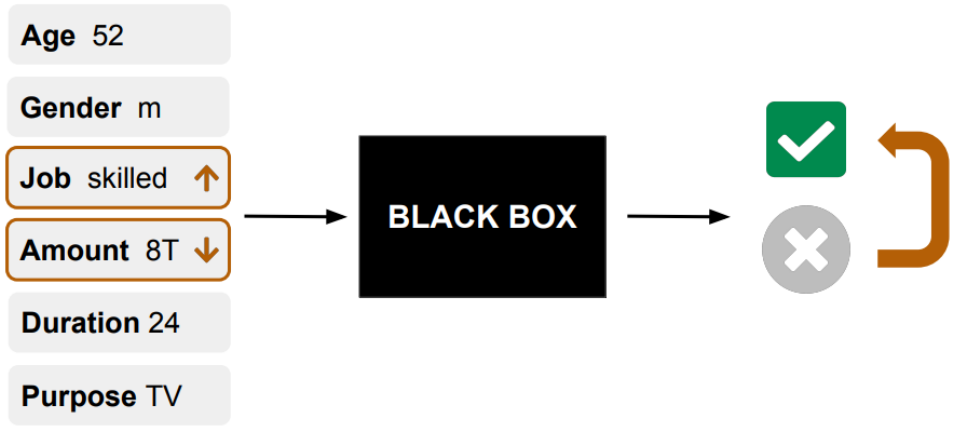
https://slds-lmu.github.io/iml/chapters/

UNDERSTAND & CONTROL INDIVIDUAL PREDICTIONS

⇝ Explaining individual decisions can prevent unwanted actions based on the model

Example: Credit Risk Application
x: customer and credit information; *y*: grant or reject credit

- Why was the credit rejected?
- Is it a fair decision?
- **How should x be changed so that the credit is accepted?**



"If the person was more skilled and the credit amount had been reduced to $8.000, the credit would have been granted."

https://slds-lmu.github.io/iml/chapters/

JUSTIFICATION AND FAIRNESS

$\rightsquigarrow$ Investigate if and why biased, unexpected or discriminatory predictions were made

Example: COMPAS

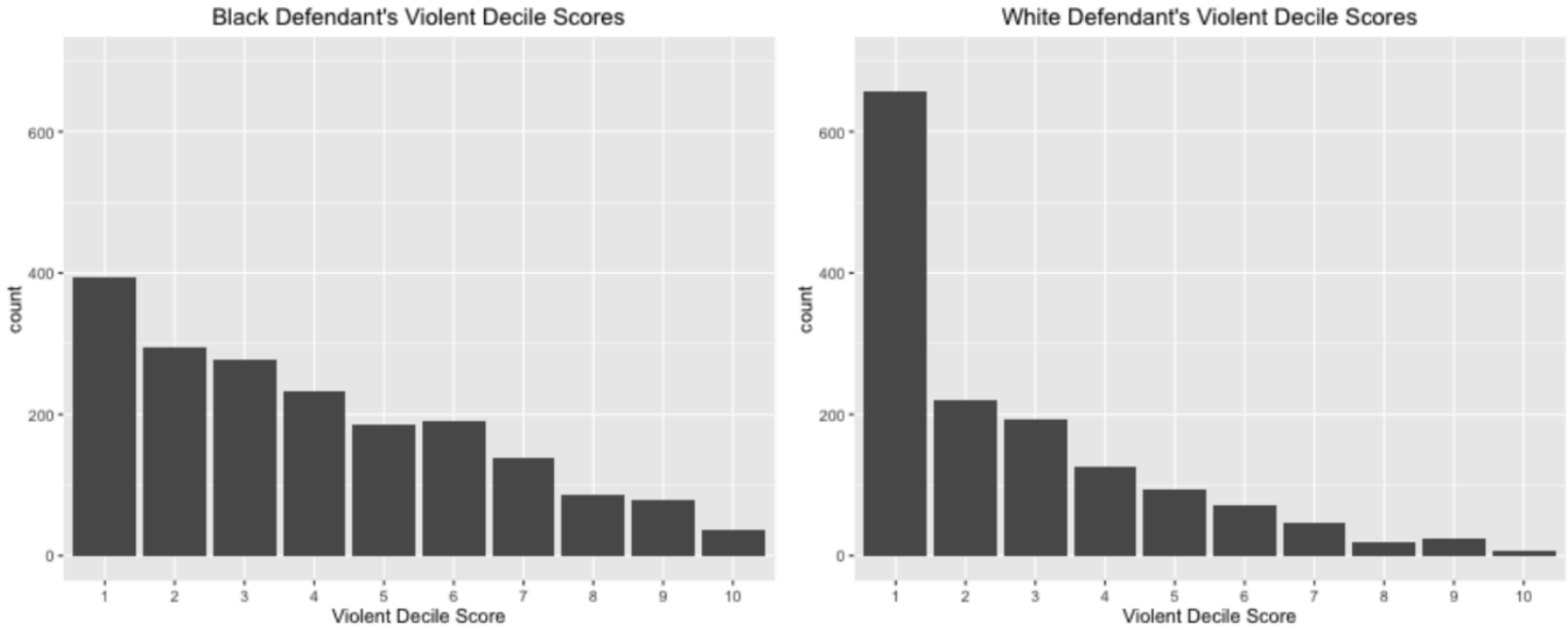
- Correctional Offender Management Profiling for Alternative Sanctions (COMPAS)
- Commercial algorithm used by judges to assess defendant's likelihood of re-offending
- Predict recidivism risk
  - i.e., criminal re-offense after previous crime, resulting in jail booking
  - different risk levels: high risk, medium risk or low risk
- Evaluate risk of recidivism based on questionnaire answered by the defendant

https://slds-lmu.github.io/iml/chapters/

JUSTIFICATION AND FAIRNESS

⇝ Investigate if and why biased, unexpected or discriminatory predictions were made

Descriptive data analysis:



Decile score: 1 (low risk) to 10 (high risk)

COMPAS ▸ Larson et al. 2016

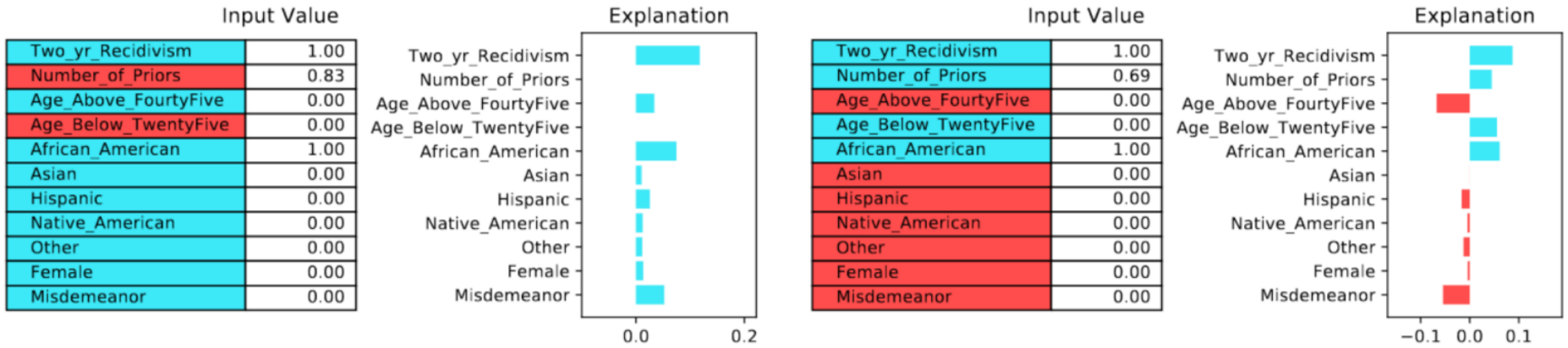
⇝ Model skewed towards low risk for white defendants
⇝ Strong indication that the model is discriminating black defendants
⇝ Use IML to investigate if and how much the model uses the defendants' origin.

https://slds-lmu.github.io/iml/chapters/

JUSTIFICATION AND FAIRNESS

⇝ Investigate if and why biased, unexpected or discriminatory predictions were made

- Feature effects analysis for two exemplary defendants, using different interpretation methods (SHAP and LIME):



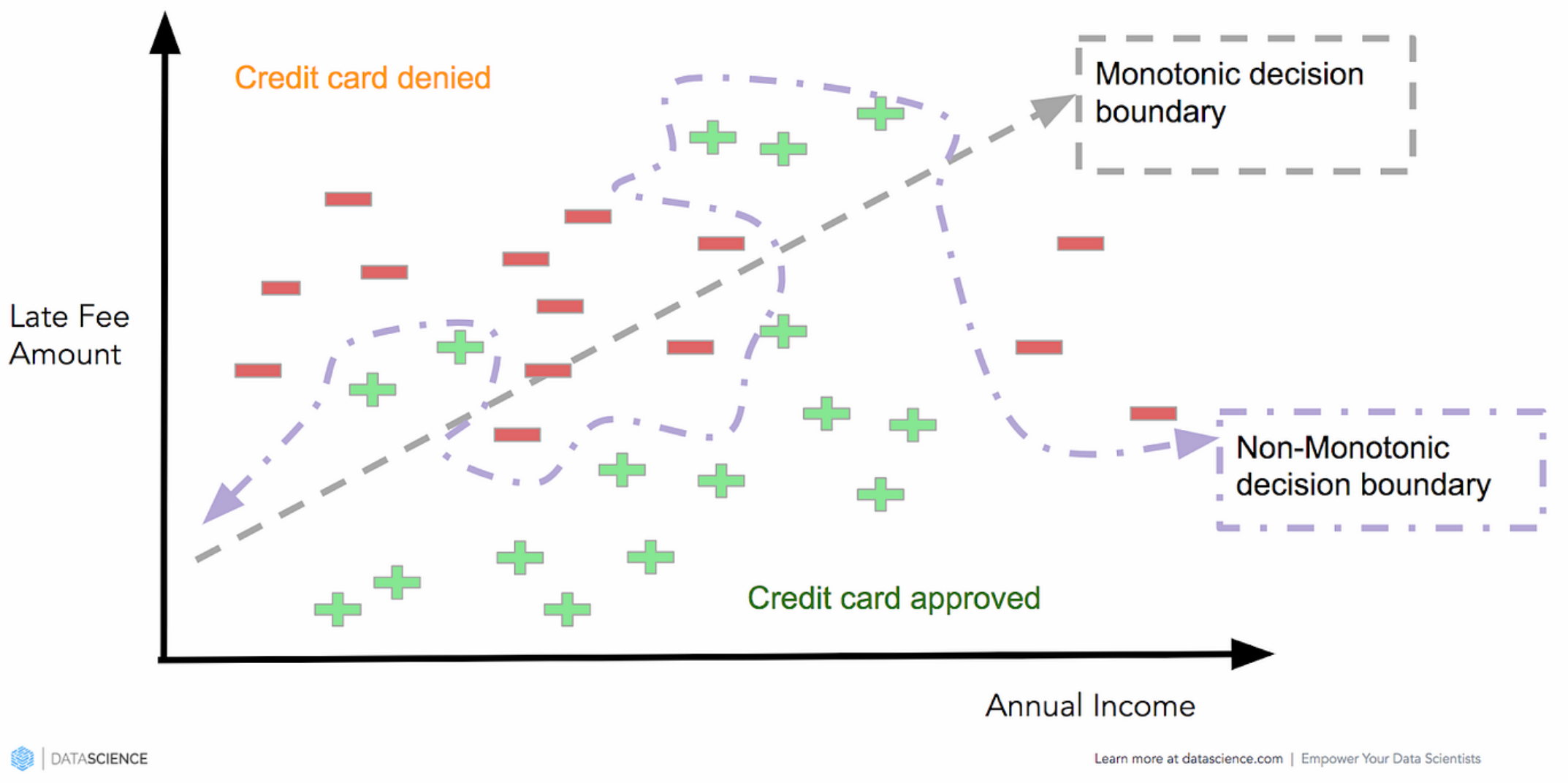
SHAP LIME

▸ Alvarez-Melis and Jaakkola 2018

⇝ Methods give for every feature a number mirroring the impact on violence score.
⇝ Race (african american) has a noticeable positive impact on violent score

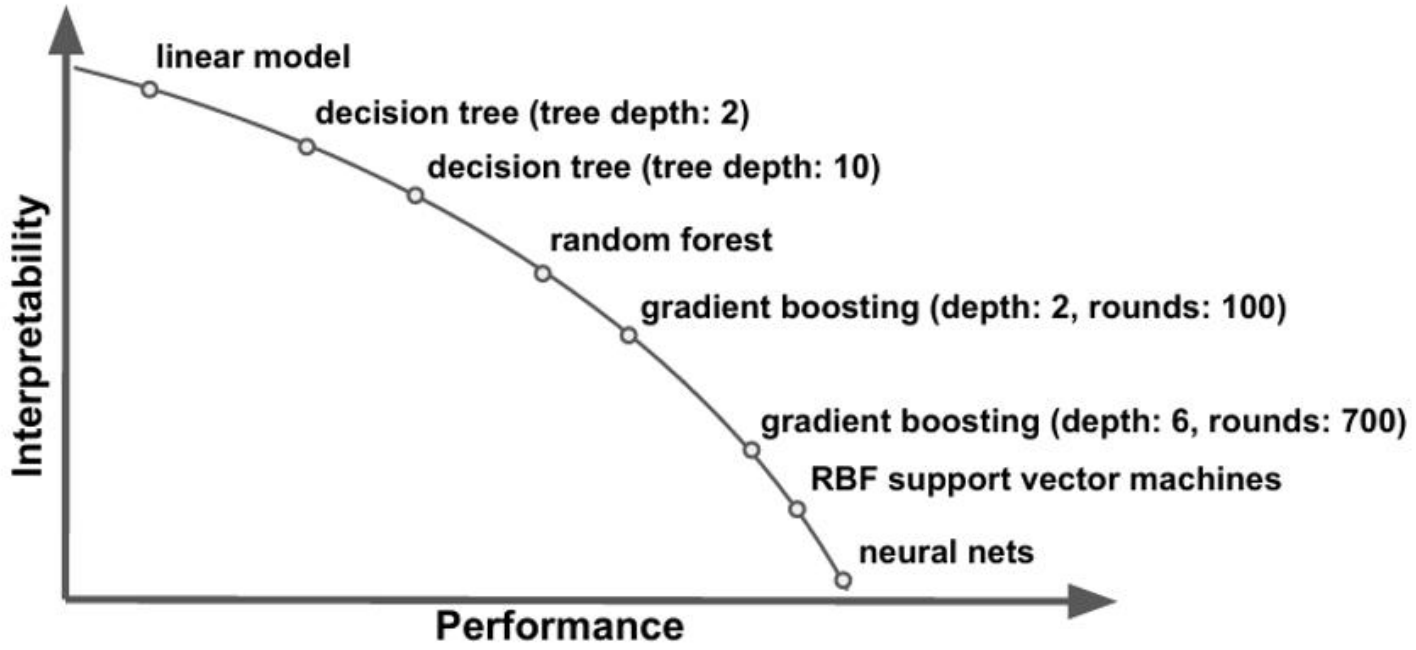
https://slds-lmu.github.io/iml/chapters/

ML Model performance vs. interpretability



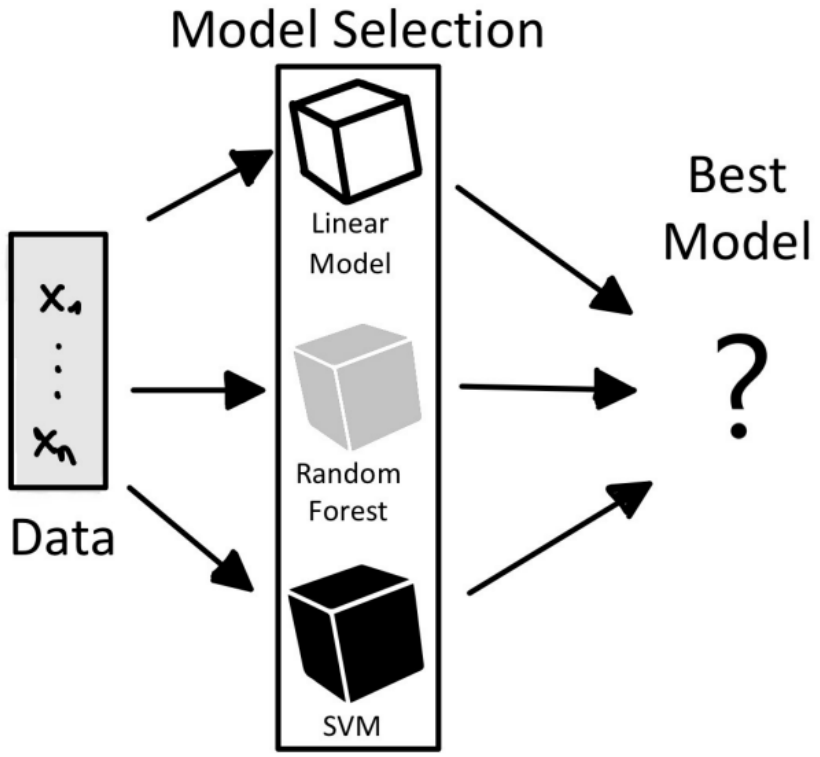
ML Model performance vs. interpretability

⇝ Many disciplines with required trust rely on traditional models, e.g., linear models, with less predictive performance



https://slds-lmu.github.io/iml/chapters/

Dimensions of Interpretability

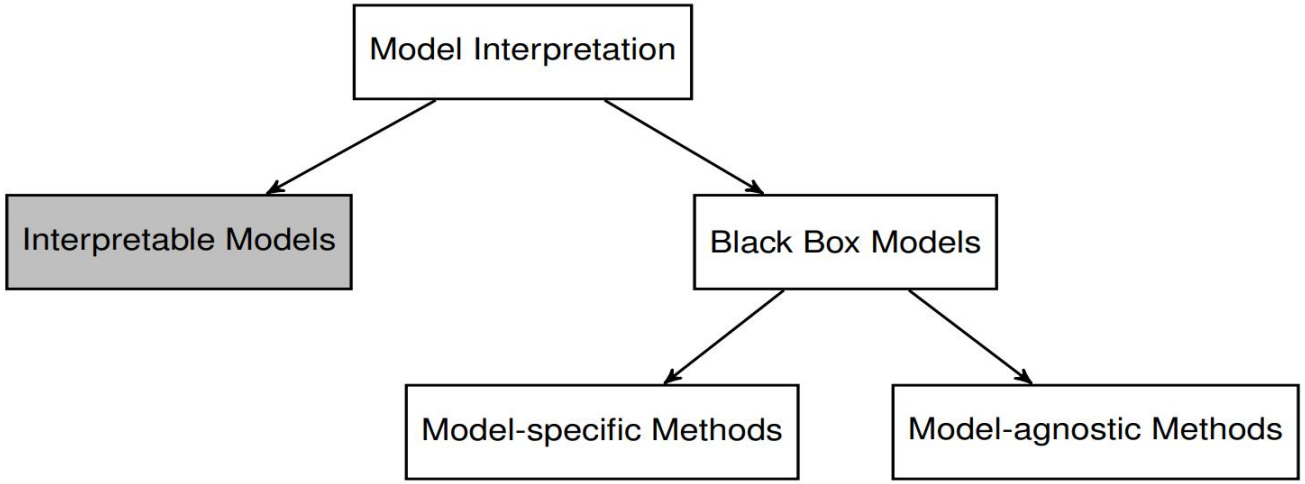


Learning goals

- Intrinsic vs. model-agnostic methods
- Different types of explanations
- Local vs. global methods
- Model or learner explanations – with or without refits
- Levels of interpretability

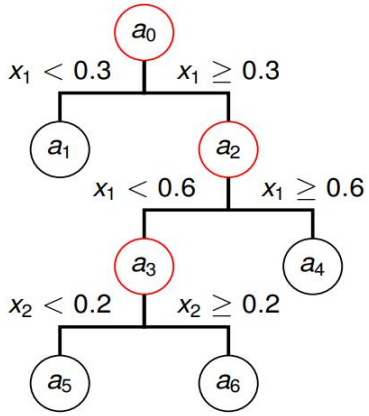
https://slds-lmu.github.io/iml/chapters/

INTRINSIC VS. MODEL-AGNOSTIC



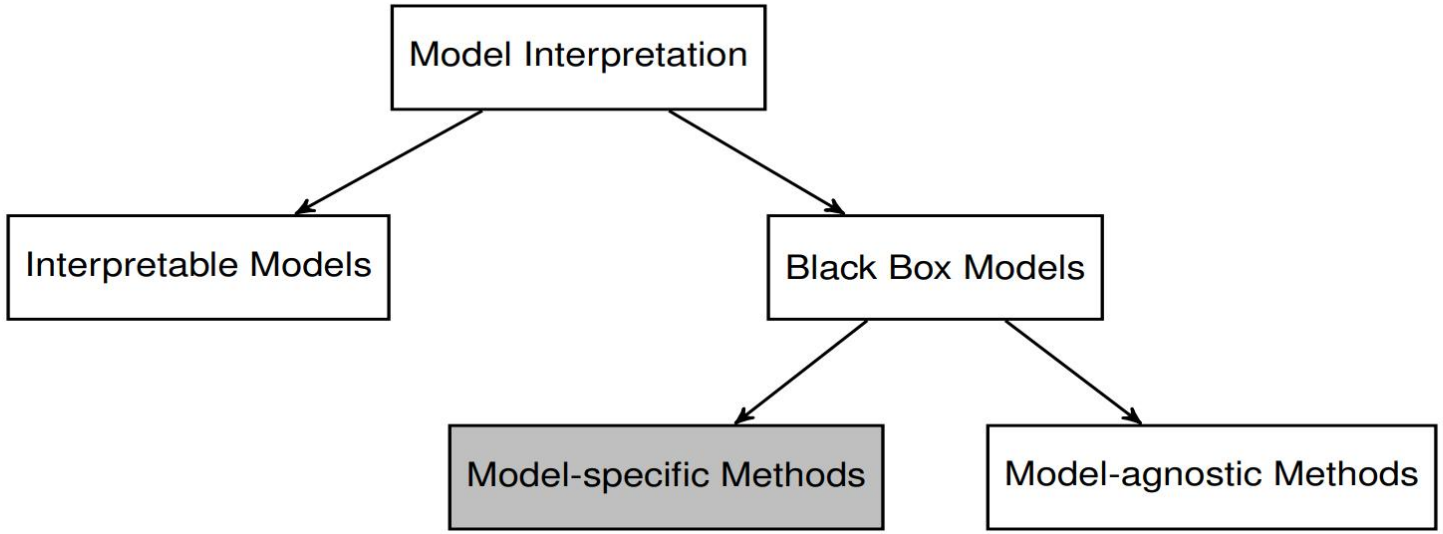
Intrinsically interpretable models:

- Examples: linear model, decision tree, decision rule, GLMs
- Interpretable because of simple model structure, e.g., weighted combination of feature values or tree structure
- Difficult to interpret with many features / complex interactions



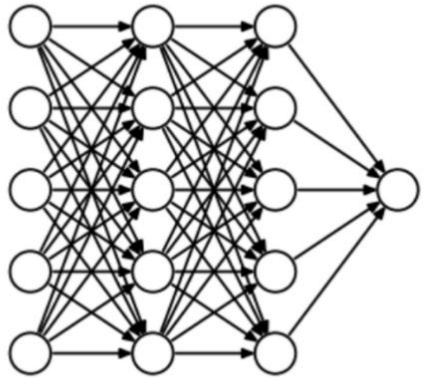
https://slds-lmu.github.io/iml/chapters/

INTRINSIC VS. MODEL-AGNOSTIC



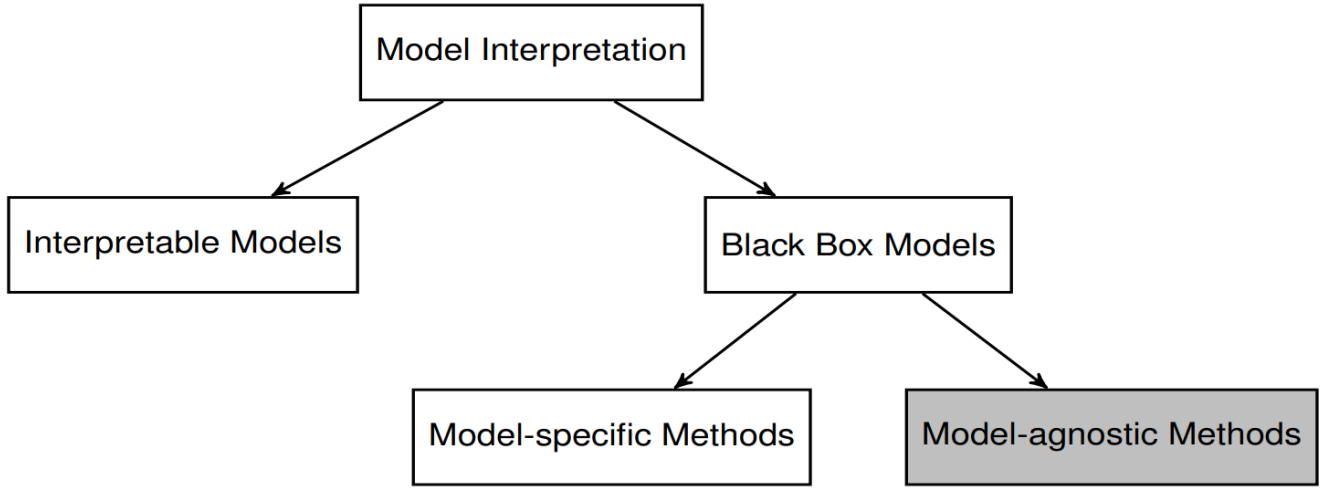
Model-specific methods:

- Interpretation method applicable to a specific ML model
- Example: implicitly integrated feature interpretation methods in tree based models, e.g., Gini Importance
- Advantage: Can exploit model structure
- Visualize activations of NNs



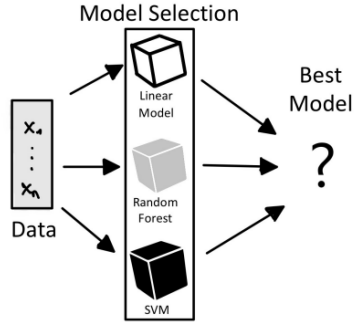
https://slds-lmu.github.io/iml/chapters/

INTRINSIC VS. MODEL-AGNOSTIC



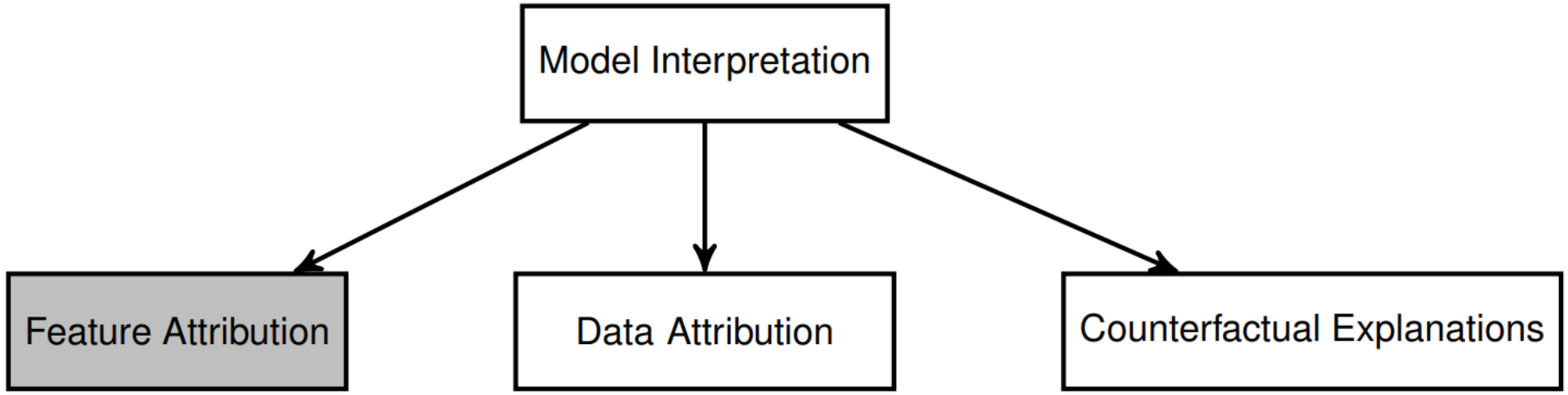
Model-agnostic methods:

- In ML: Tune over many model classes
  ⇝ Unknown which model is best / deployed
  ⇝ Need for interpretation methods applicable to any model
- Applied after training (post-hoc)
- Applicable to intrinsically interpretable models
  ⇝ provides insights into other types of explanations



https://slds-lmu.github.io/iml/chapters/

TYPES OF EXPLANATIONS

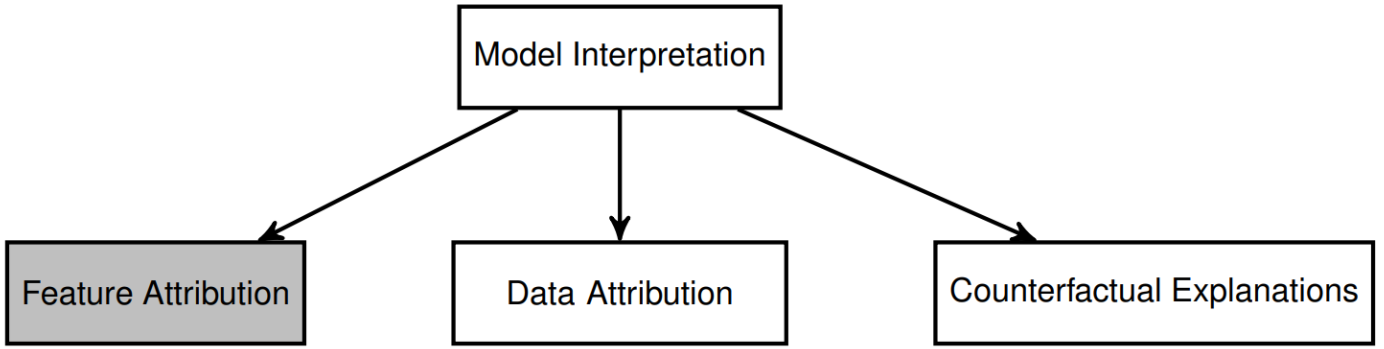


Feature Attribution:

- Produce explanations on a per-feature level, e.g., feature effects or feature importance
- Vary feature values, inspect change of model prediction, model variance or model error

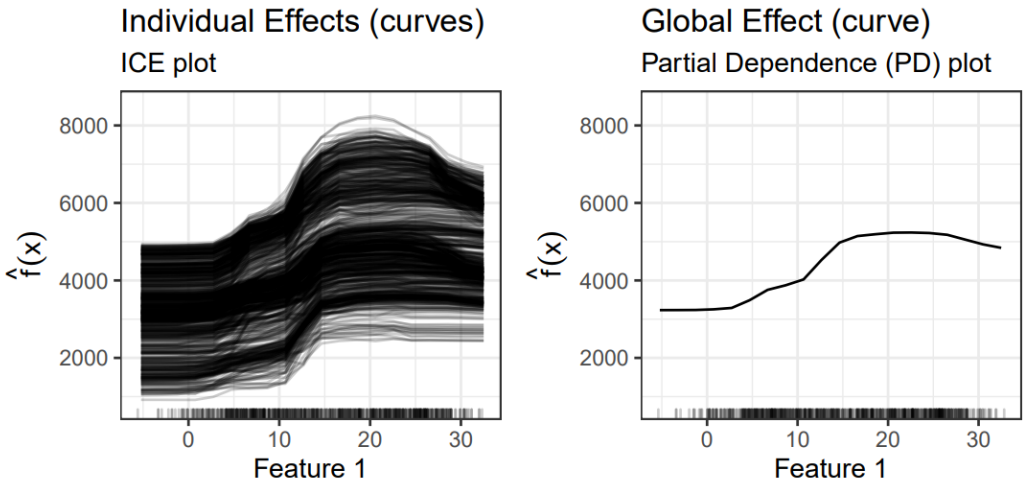
https://slds-lmu.github.io/iml/chapters/

TYPES OF EXPLANATIONS



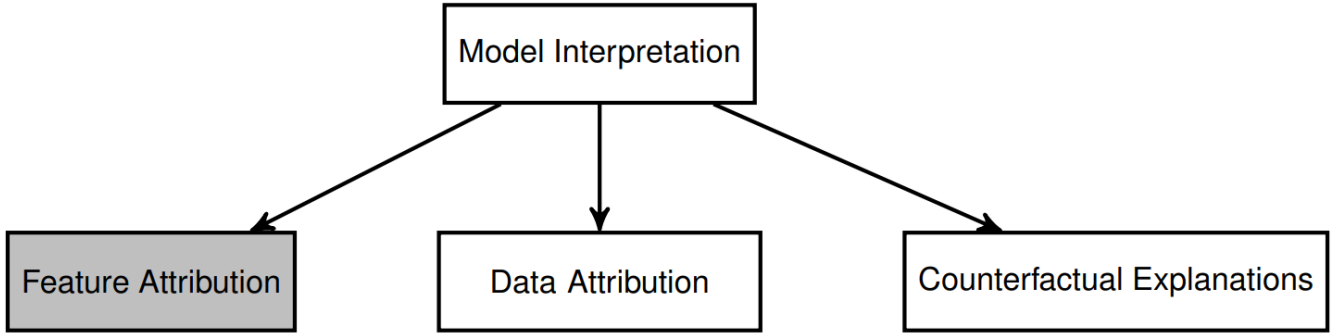
Feature Effects indicate the change in prediction due to changes in feature values.

- Model-agnostic methods: ICE curves, PD plots . . .
- Pendant in linear models: Regression coefficient θ_j
- Further examples: Saliency Maps, model-agnostic methods such as SHAP and LIME



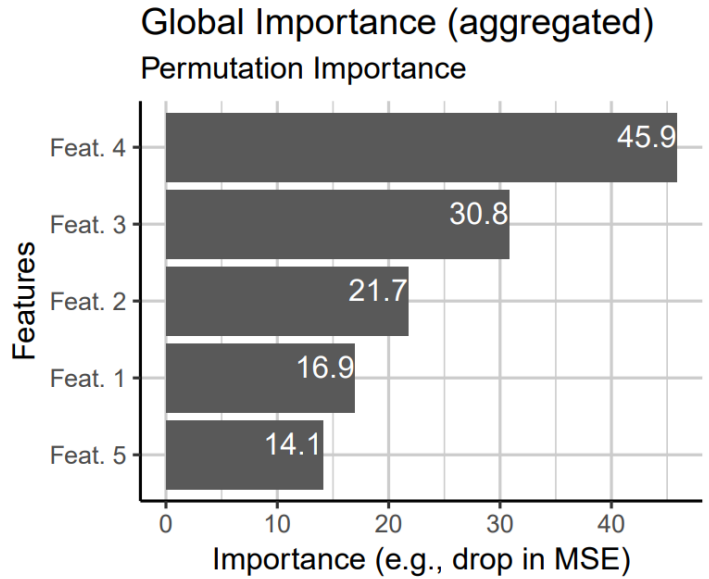
https://slds-lmu.github.io/iml/chapters/

TYPES OF EXPLANATIONS



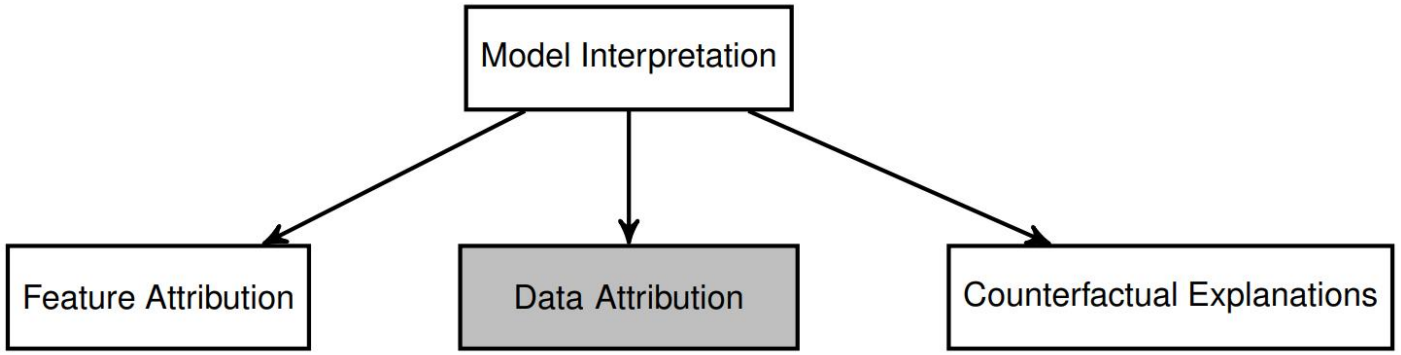
Feature importance methods rank features by how much they contribute to the predictive performance or prediction variance of the model.

- Model-agnostic methods: PFI, . . .
- Pendant in linear models: t-statistic, p-value (significant effect)



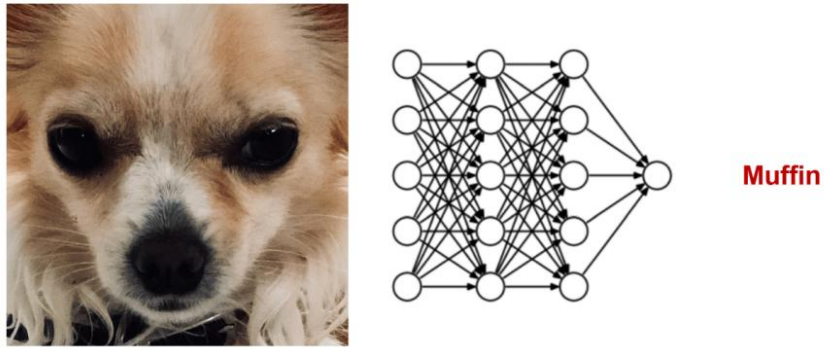
https://slds-lmu.github.io/iml/chapters/

TYPES OF EXPLANATIONS



Data Attribution: Identify training instances most responsible for a decision (e.g. Influence Functions)

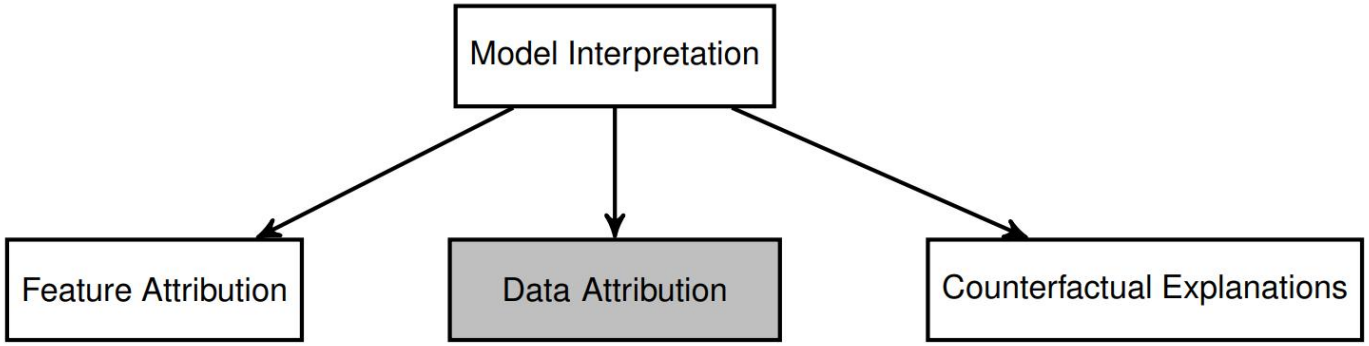
Example: Consider a model which should distinguish muffins and dogs



How does this incorrect prediction come about?

https://slds-lmu.github.io/iml/chapters/

TYPES OF EXPLANATIONS



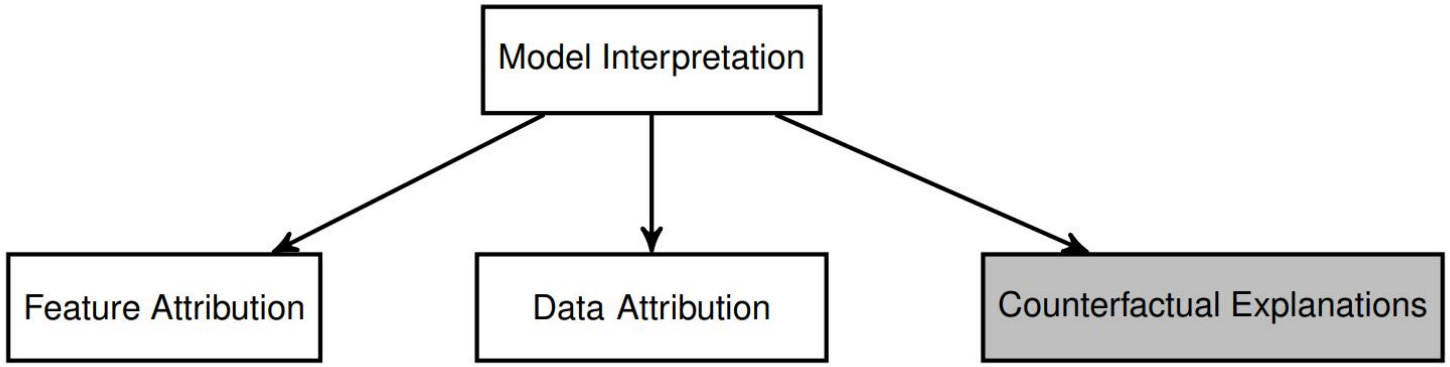
Data Attribution: Identify training instances most responsible for a decision (e.g. Influence Functions)
Look at training data: Which data points caused the model prediction?

Method searches for the most similar images and bases the decision on them

- ⇝ Training images looking most like new input show a muffin
- ⇝ Wrong output (muffin instead of dog)

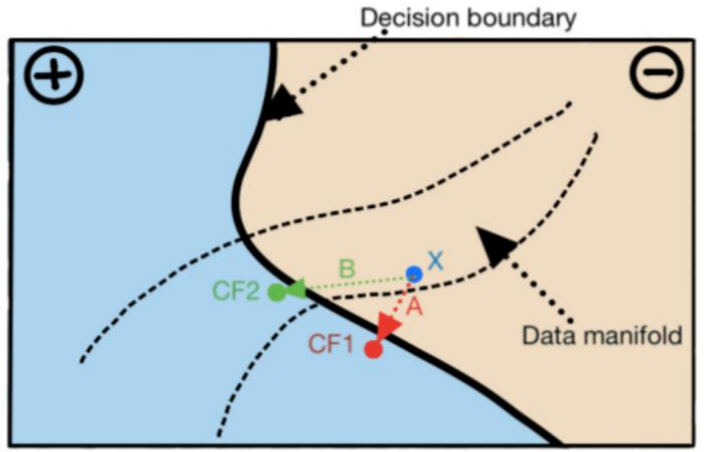
https://slds-lmu.github.io/iml/chapters/

TYPES OF EXPLANATIONS



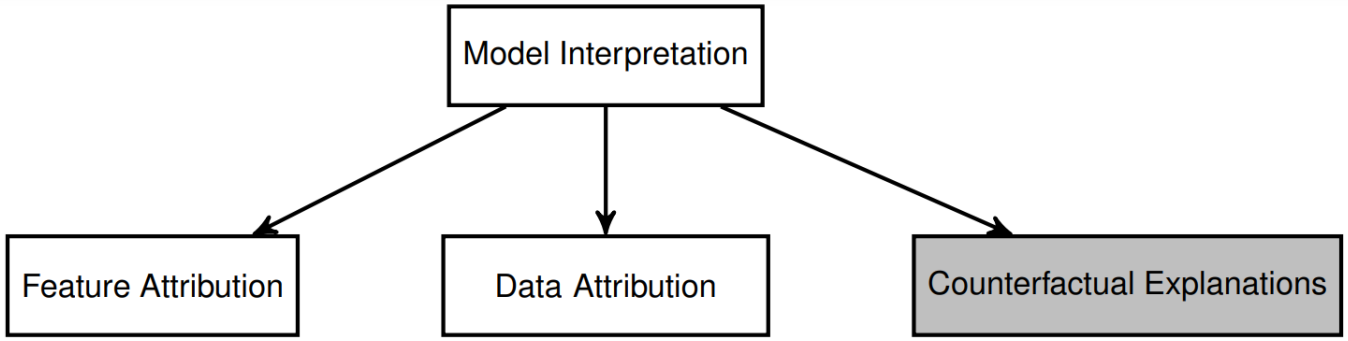
Counterfactual Explanations:

- Identify smallest necessary change in feature values so that a desired outcome is predicted
- Contrastive explanations
- Diverse counterfactuals
- Feasible & actionable explanations

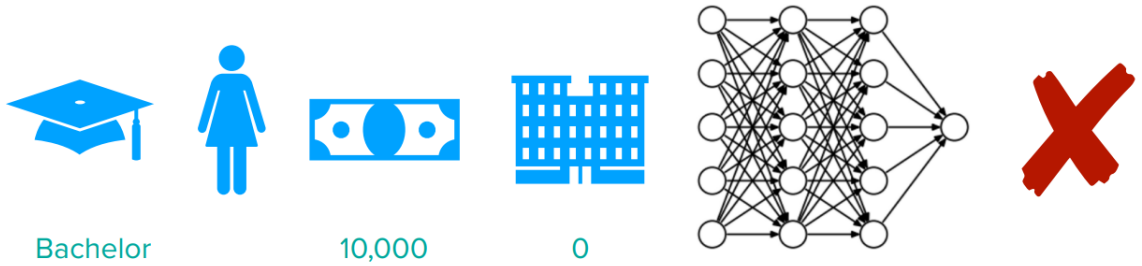


https://slds-lmu.github.io/iml/chapters/

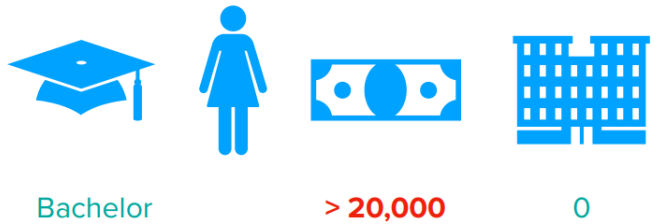
TYPES OF EXPLANATIONS



Example (loan application):



What can a person do to obtain a favorable prediction from a given model ?

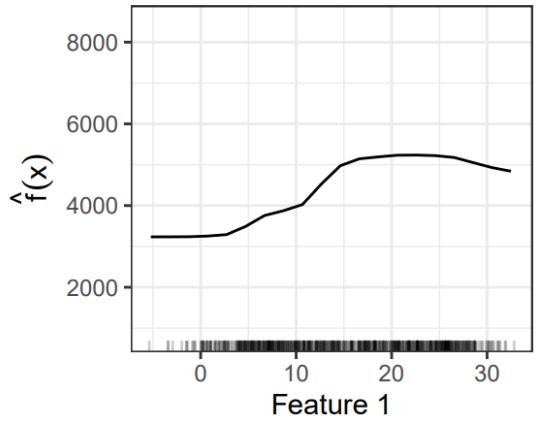


https://slds-lmu.github.io/iml/chapters/

GLOBAL VS. LOCAL

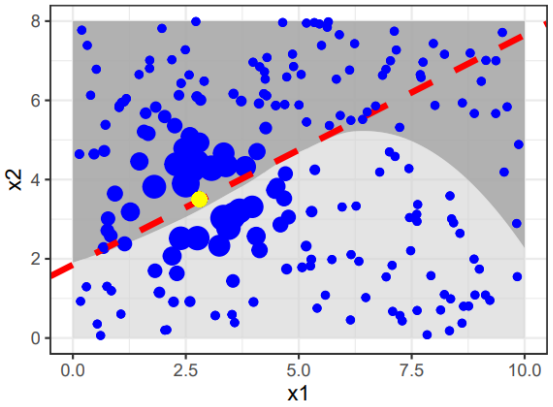
Global interpretation methods explain the model behavior for the entire input space by considering all available observations:

- Permutation Feature Importance (PFI)
- Partial Dependence (PD) plots
- Accumulated Local Effect (ALE) plots
- ...



Local interpretation methods explain the model behavior for single data instances:

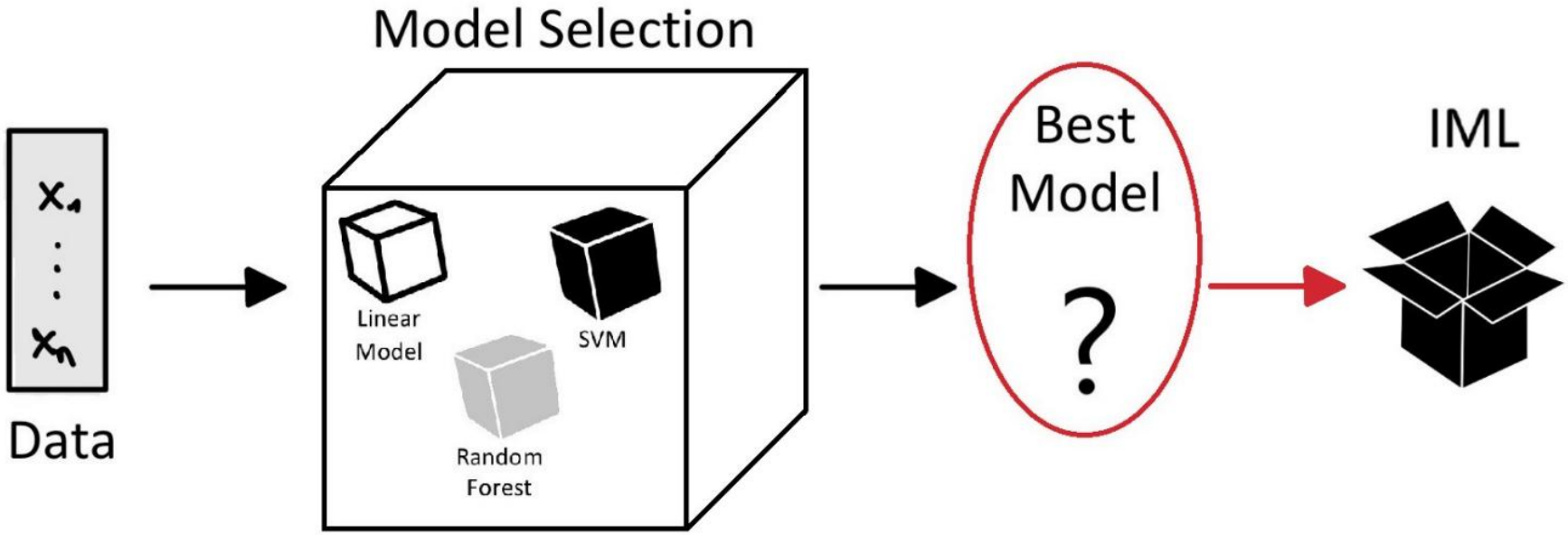
- Individual Conditional Expectation (ICE) curves
- Local Interpretable Model-Agnostic Explanations (LIME)
- Shapley values, SHAP
- ...



https://slds-lmu.github.io/iml/chapters/

LEVELS OF INTERPRETABILITY

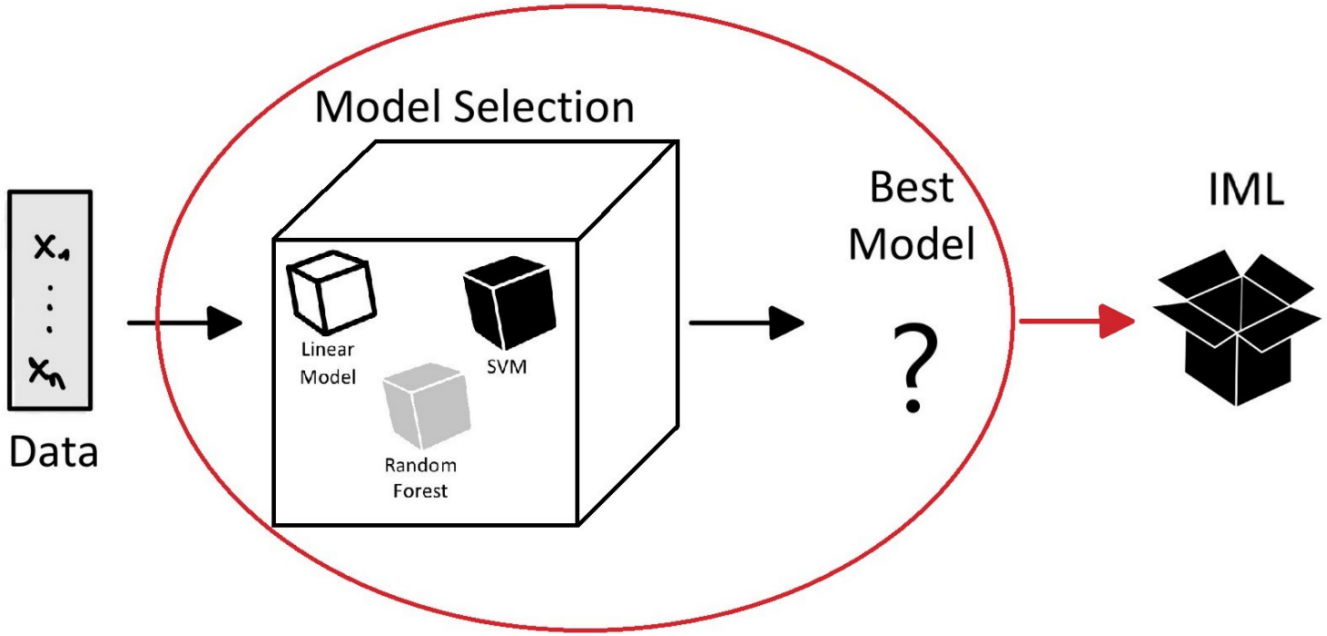
	Research Question	Objects of analysis
1st level view	How to explain a given model fitted on a data set?	(deployed) model $\theta \mapsto \hat{f}(\theta)$



https://slds-lmu.github.io/iml/chapters/

LEVELS OF INTERPRETABILITY

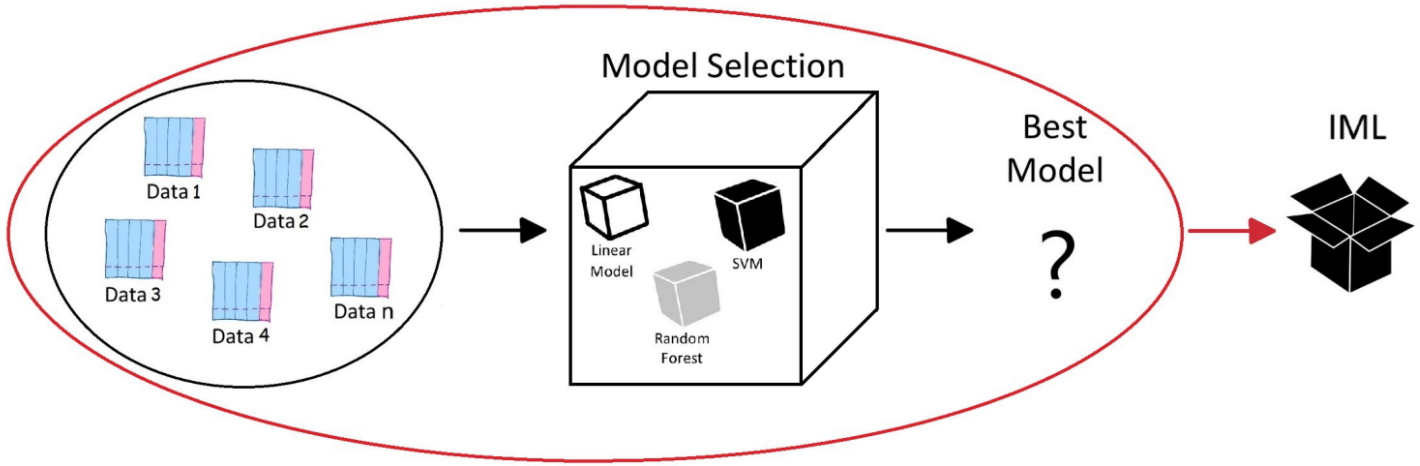
	Research Question	Objects of analysis
1st level view	How to explain a given model fitted on a data set?	(deployed) model $\theta \mapsto \hat{f}(\theta)$
2nd level view	How does an optimizer choose a model based on a data set?	Model selection process (e.g., decisions made by AutoML systems or HPO process)



https://slds-lmu.github.io/iml/chapters/

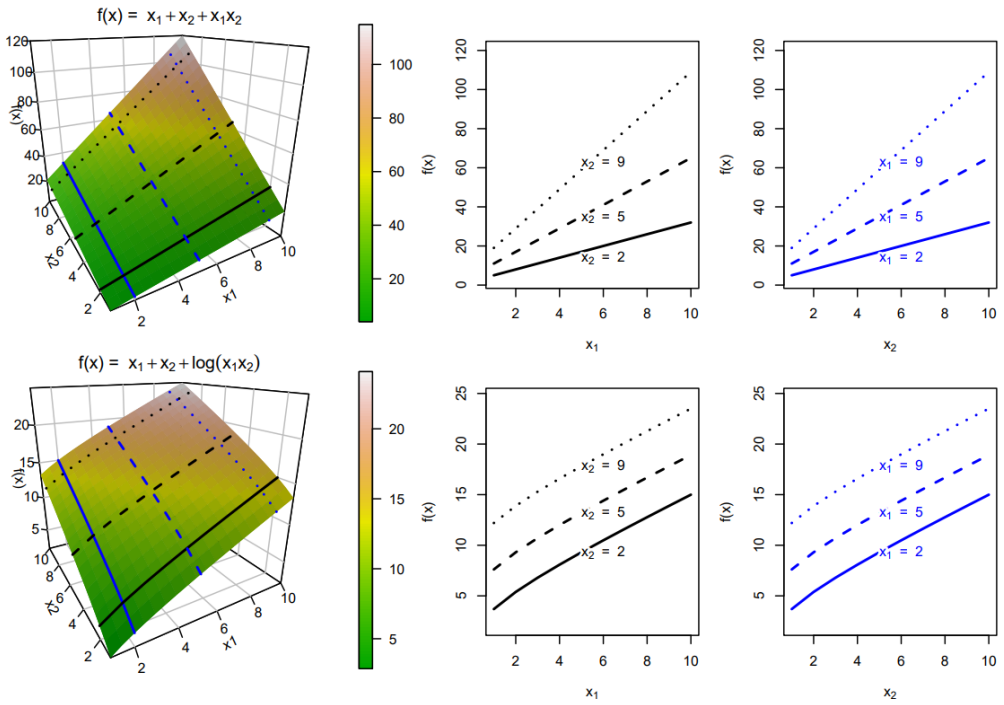
LEVELS OF INTERPRETABILITY

	Research Question	Objects of analysis
1st level view	How to explain a given model fitted on a data set?	(deployed) model $\theta \mapsto \hat{f}(\theta)$
2nd level view	How does an optimizer choose a model based on a data set?	Model selection process (e.g., decisions made by AutoML systems or HPO process)
3rd level view	How do data properties relate to performance of a learner and its hyperparameters?	properties of ML algorithms in general (benchmark)



https://slds-lmu.github.io/iml/chapters/

Feature Interactions



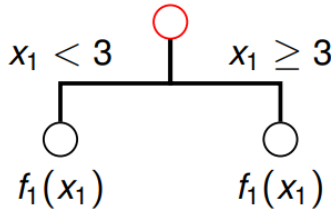
Learning goals

- Feature interactions
- Difference to feature dependencies

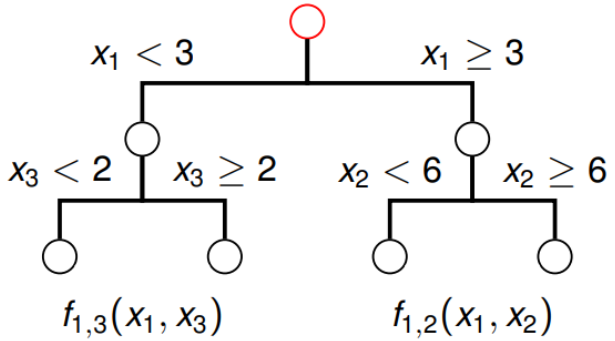
https://slds-lmu.github.io/iml/chapters/

Feature Interactions

- Feature dependencies concern data distribution
- Feature interactions may occur in structure of **both** model or DGP (e.g., functional relationship between X and $\hat{f}(X)$ or X and $Y = f(X)$)
  $\rightsquigarrow$ Feature dependencies may lead to feature interactions in a model
- No. of potential interactions increases exponentially with no. of features
  $\rightsquigarrow$ Difficult to identify interactions, especially when features are dependent
- Interactions: A feature's effect on the prediction depends on other features
  $\rightsquigarrow$ Example: $\hat{f}(\mathbf{x}) = x_1 x_2 \Rightarrow$ Effect of x_1 on $\hat{f}$ depends on x_2 and vice versa



No interaction



Interactions: x_1 and x_3,
x_1 and x_2
No interactions: x_2 and x_3

https://slds-lmu.github.io/iml/chapters/

Feature Interactions

Definition: A function $f(\mathbf{x})$ contains an interaction between x_j and x_k if a difference in $f(\mathbf{x})$-values due to changes in x_j will also depend on x_k, i.e.:

$$\mathbb{E}\left[\frac{\partial^2 f(\mathbf{x})}{\partial x_j \partial x_k}\right]^2 > 0$$

$\Rightarrow$ If x_j and x_k do not interact, $f(\mathbf{x})$ is sum of 2 functions, each independent of x_j, x_k:

$$f(\mathbf{x}) = f_{-j}(x_1, \ldots, x_{j-1}, x_{j+1}, \ldots, x_p) + f_{-k}(x_1, \ldots, x_{k-1}, x_{k+1}, \ldots, x_p)$$

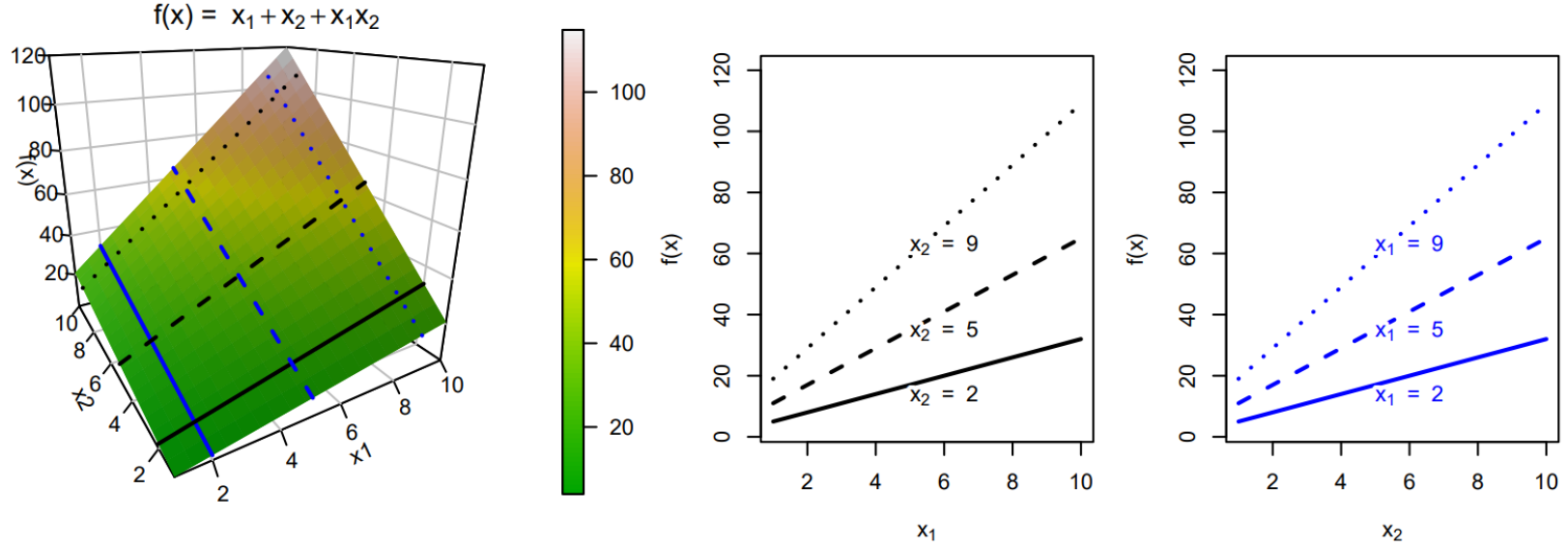
▸ Friedman and Popescu (2008)

Feature Interactions

Example: $f(\mathbf{x}) = x_1 + x_2 + x_1 \cdot x_2$ (not separable)

$$\mathbb{E}\left[\frac{\partial^2(x_1+x_2+x_1\cdot x_2)}{\partial x_1 \partial x_2}\right]^2 = \mathbb{E}\left[\frac{\partial(1+x_2)}{\partial x_2}\right]^2 = 1 > 0$$

$\Rightarrow$ interaction between x_1 and x_2



- Effect of x_1 on $f(\mathbf{x})$ varies with x_2 (and vice versa)

$\Rightarrow$ Different slopes

https://slds-lmu.github.io/iml/chapters/

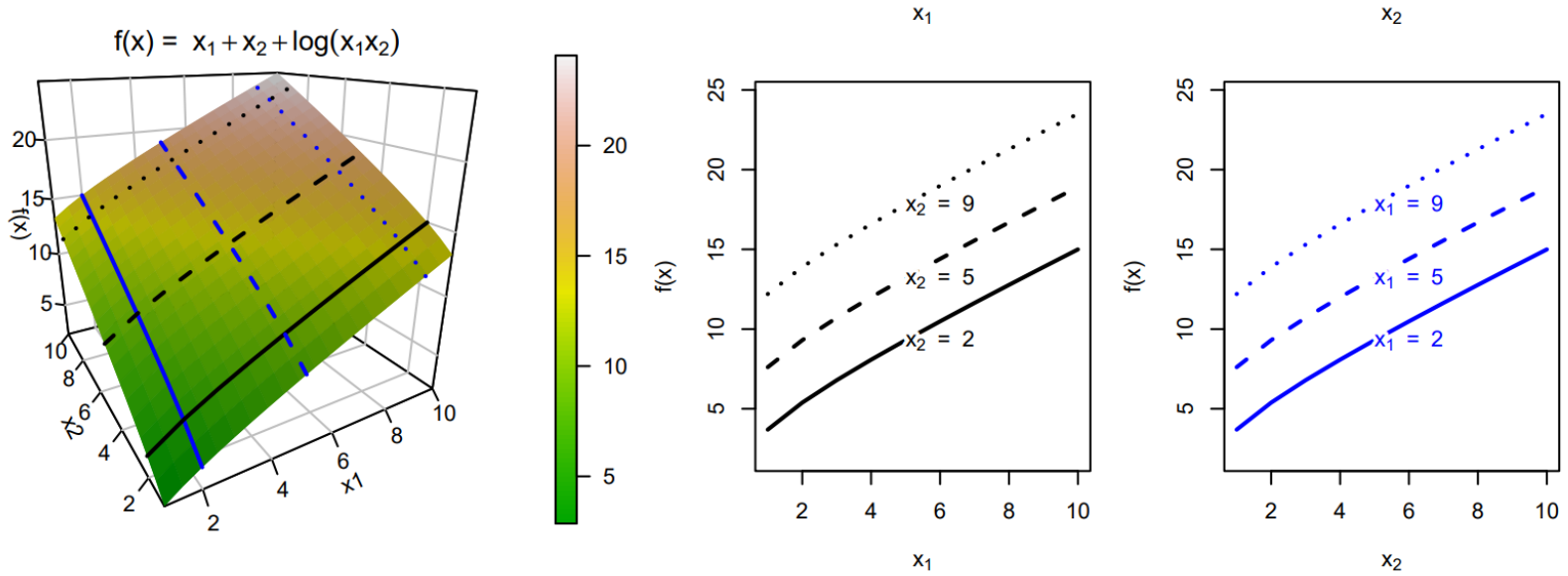
Feature Interactions

Example of separable function:

$$f(\mathbf{x}) = x_1 + x_2 + \log(x_1 \cdot x_2) = x_1 + x_2 + \log(x_1) + \log(x_2)$$

$$\Rightarrow f(\mathbf{x}) = f_1(x_1) + f_2(x_2) \text{ with } f_1(x_1) = x_1 + \log(x_1) \text{ and } f_2(x_2) = x_2 + \log(x_2)$$

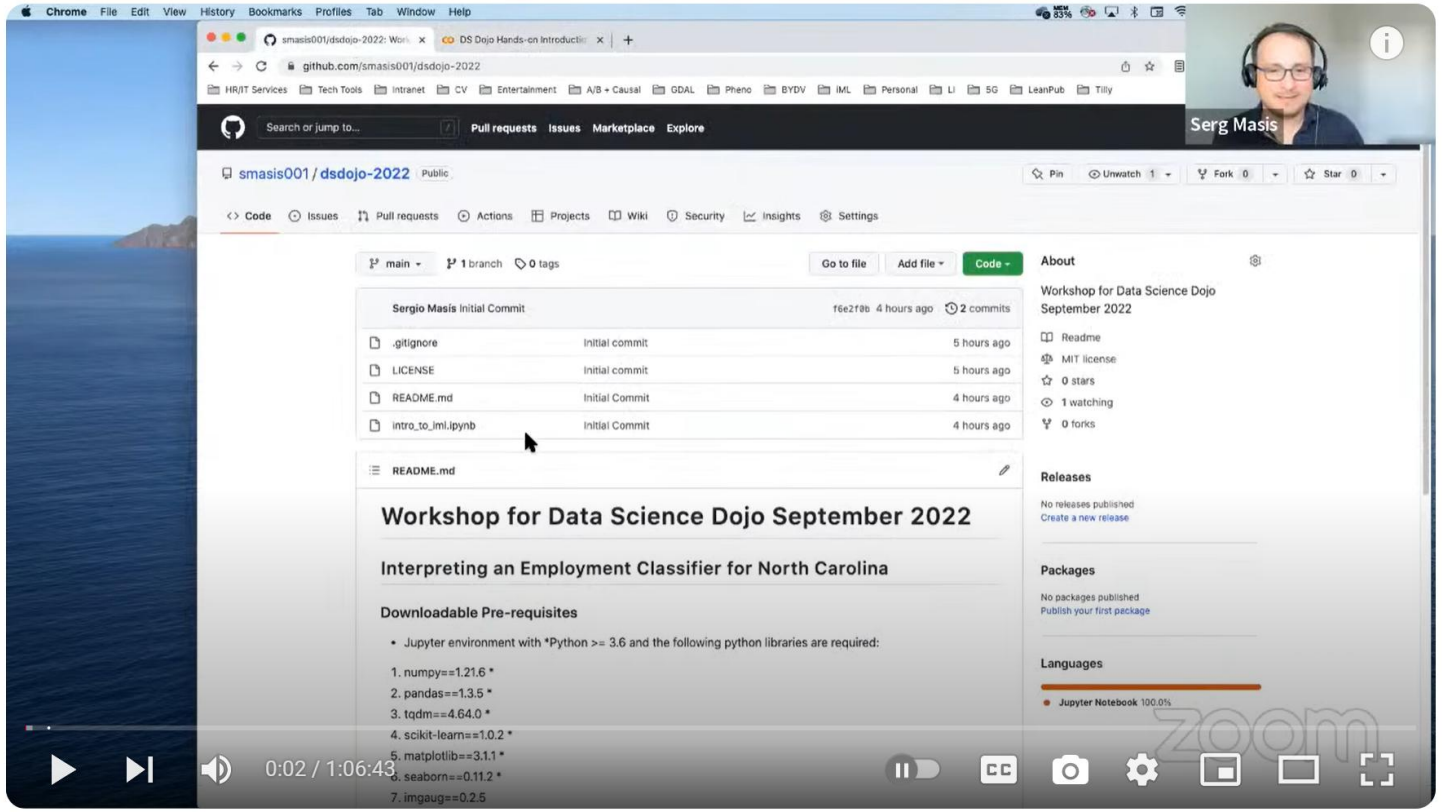
$$\Rightarrow \text{no interactions due to separability, also } \mathbb{E}\left[\frac{\partial^2 f(\mathbf{x})}{\partial x_1 \partial x_2}\right]^2 = 0$$



- Effect of x_1 on $f(\mathbf{x})$ stays the same for different x_2 values (and vice versa)

$\Rightarrow$ Parallel lines at different horizontal (blue) or vertical (black) slices

Hands-on Exercises (1)



https://colab.research.google.com/drive/107rimlhRJvsQF7nDv4cO3asdU1I2irak?usp=sharing

https://github.com/smasis001/dsdojo-2022

https://www.youtube.com/watch?v=cRwB2W1oTvI

Hands-on Exercises (2)

interpretable ML

In [1]:
```
## we start again by reading our data
import pandas as pd
from sklearn.model_selection import train_test_split

eye_movements = pd.read_csv("../data/eye_movements_aggregated.csv")
eye_features = eye_movements.loc[:,['fixcount', 'firstPassCnt', 'P1stFixation', 'P2stFixation',
       'prevFixDur', 'firstfixDur', 'firstPassFixDur', 'nextFixDur',
       'firstSaccLen', 'lastSaccLen', 'prevFixPos', 'landingPos', 'leavingPos',
       'totalFixDur', 'meanFixDur', 'nRegressFrom', 'regressLen',
       'nextWordRegress', 'regressDur', 'pupilDiamMax', 'pupilDiamLag',
       'timePrtctg']]
labels = eye_movements['target'].astype(int)
```

In [2]:
```
## multi-class with SHAP adds a layer of complexity, we will keep it simple here and
## focus on a model separating the category 2 (relevant and correct) from the rest
labels = labels == 2
labels.value_counts()
```

Out[2]:
```
target
False    2328
True      336
Name: count, dtype: int64
```

In [3]:
```
X_train, X_valid, y_train, y_valid = train_test_split(eye_features, labels,
                                                      stratify=labels,
                                                      test_size=0.2, random_state=123)
```

https://github.com/sib-swiss/intermediate-machine-learning-training/blob/master/notebooks/04_model_explanation.ipynb

Thanks for your attention!

Giảng viên: TS. Lưu Phúc Lợi

Luu.p.loi@googlemail.com

Zalo: 0901802182