AIAA 5047
Responsible AI
2025 Fall

Sihong Xie, AI Thrust, Information Hub

Lecture 3

W2 201, 9-11:50 AM F

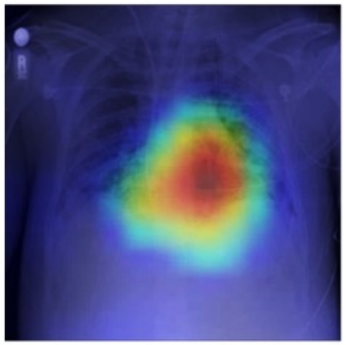
Motivations

- To understand a model's behavior
  - ***how*** and ***why*** a model arrives at a particular conclusion?
- Example
  - A deep model trained to classify chest X-ray images into (a) Pneumonia or (b) COVID-19.
  - Do the model look at **the *right* places** to classify the images? "Right places" means those places where the doctors use for classification.
  - "Explanations" are defined as "regions of the images that are important to the model's decision".
  - By seeing the explanations, doctors can decide if the models are reliable.

(a) Actual (Pneumonia Class)

(b) Predicted (Pneumonia Maps)

(c) Actual (COVID-19 Class)



(d) Predicted (COVID-19 Maps)

https://bmcmedimaging.biomedcentral.com/articles/10.1186/s12880-024-01394-2

Motivations

- To predict the model's behaviors
  - Will the model behave well in novel or unexpected situations?
- Example
  - In autonomous driving (a safety-critical AI application), it is important to predict the car's behavior under unusual driving conditions.
  - It is possible that autopilot has not been trained on any conditions like this, and it did not recognize the all-white truck as an obstacle.
  - Interpreting what autopilot would recognize an overturned truck can be helpful.

In a Tesla accident, a Tesla with autopilot crashed into an overturned truck (edge case).

Training data collected under usual situations cannot cover rare but dangerous conditions.

Autopilot:
- Vision-only
- Data-driven

https://insideevs.com/news/426312/video-tesla-crash-stopped-truck/

Motivations

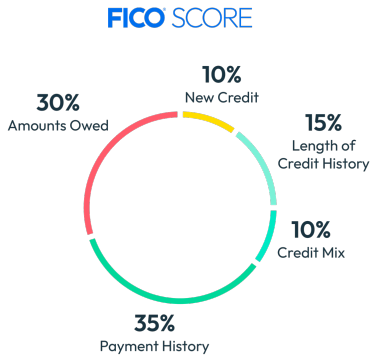
- To intervene: when an AI model makes a decision that impacts a person's life, it's essential to **override** that decision.
  - A clear explanation of why the decision was made helps with the intervention.

A credit card application

Apply again

Interventions:
1) Reduce electricity bill
2) Less dining out
3) No purchase of luxuries.
4) ……

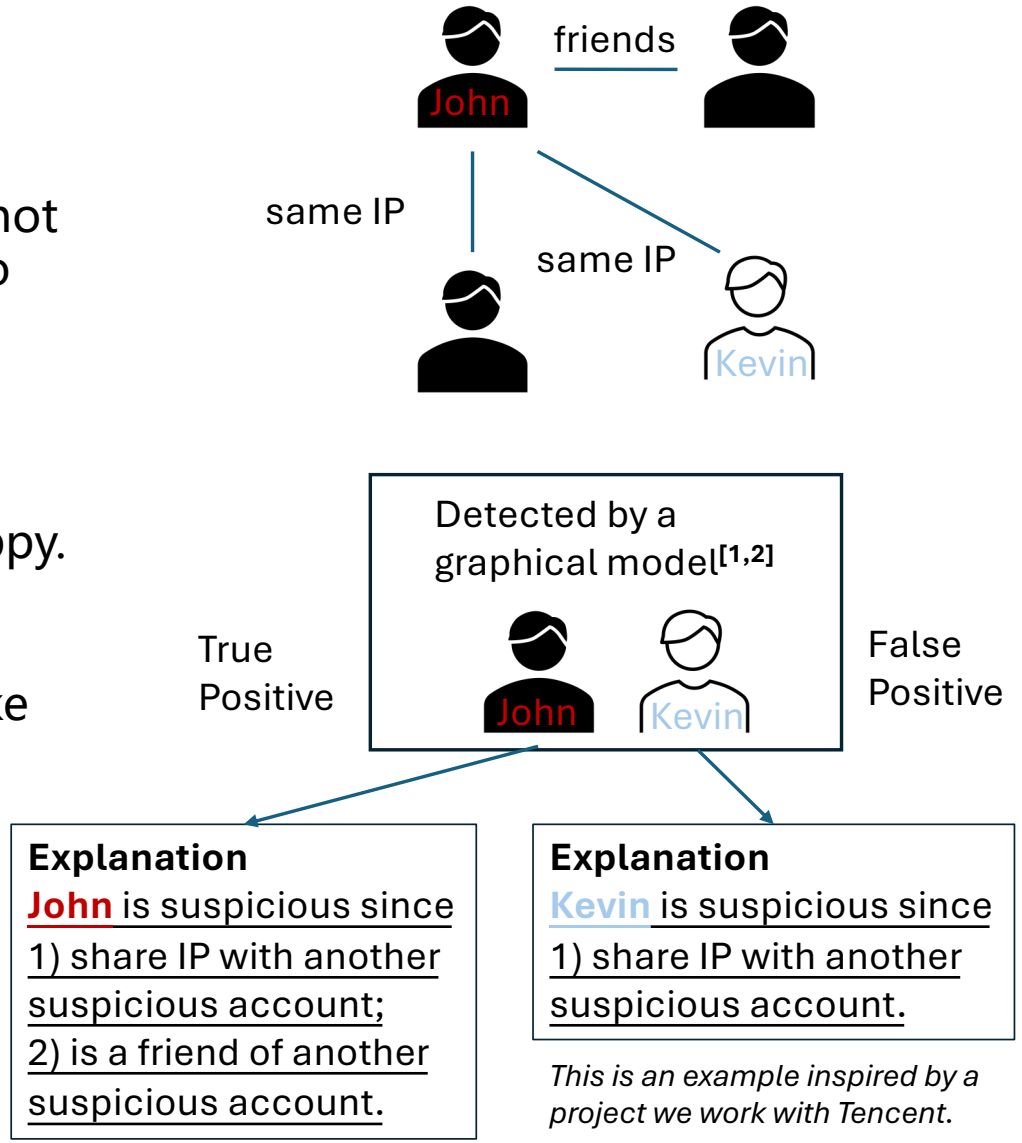
Global evaluation factors



Explanation: *the case is denied because you owe too much, and reducing that by 50% will be helpful.*

Motivations

- Explainable financial fraud intervention
  - Detecting suspicious financial accounts is not enough, but require human intervention to stop financial loss[3].
  - But a mis-classification, in particular, false positives (e.g., Kevin in the example on the right) can make innocent customers unhappy.
  - Explanations that justify why an account is suspicious can be useful for human to make more precise decisions.



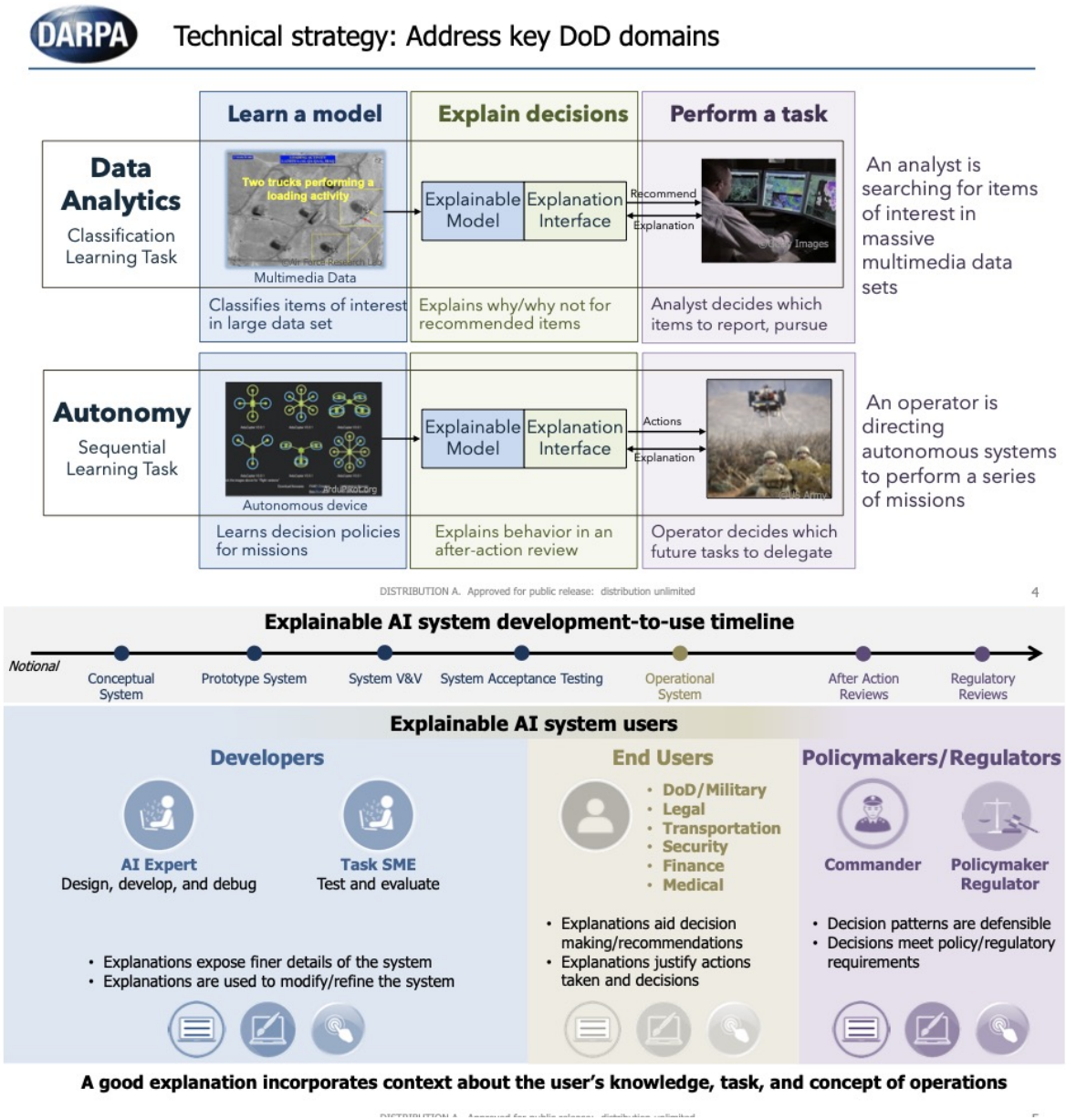
This is an example inspired by a project we work with Tencent.

Publications from my group

[1] Review graph based online store review spammer detection. ICDM 2011
[2] Inconsistent Matters: A Knowledge-guided Dual-consistency Network for Multi-modal Rumor Detection. TKDE 2023
[3] Enhancing Robustness of Graph Neural Networks on Social Media with Explainable Inverse Reinforcement Learning. NeurIPS 2024

Motivations

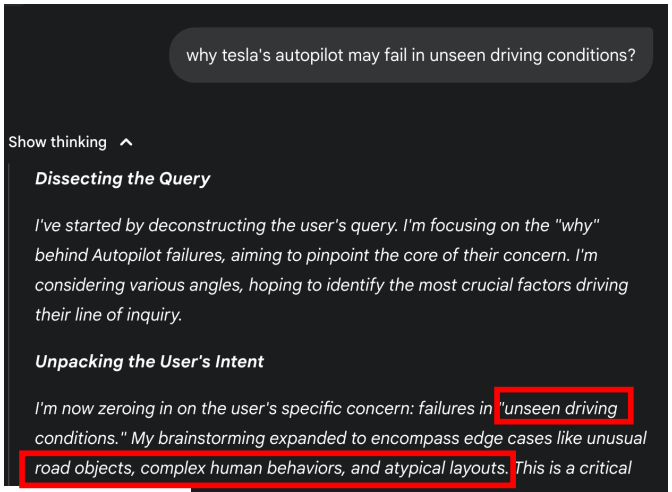
- AI for military missions
  - The AI provides suggestions and humans decide if take actions.
- Examples
  - Satellite images analysis: AI found two trucks loading something, which can be suspicious targets. Cannot check onsite and decisions need to be made using AI explanations.
  - UAV needs to bomb a target, and the AI generates a policy that executes the task. How to ensure the policy is safe and also effective? Explanation help the operator review the rationale of the policy and improve it later.



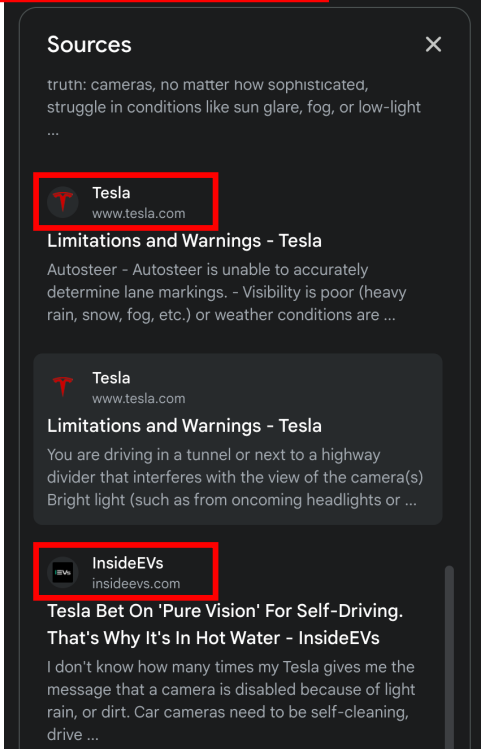
Motivations

- LLM explanations are critical:
  - Blackbox nature of most LLM.
  - Pretrained LLM has outdated knowledge[1]
  - LLM can hallucinate[2].
- Example
  - Retrieval Augmentation Generation (**RAG**) makes the information sources of LLM explicit, tracable, and verifiable (a sort of intervention).
  - Chain-of-Thought (**CoT**) guide LLM to reason in an explicit way, and can reduce hallucination (a sort of intervention).
  - DeepResearch techniques (e.g., Gemini) adopt both CoT and RAG to make the results explainable and improve user trust.
- RAG and CoT are not entirely reliable, and there are other techniques (mechanistic interpretability, knowledge circuit) for explainable LLM.

CoT shows that Gemini understands keywords like "driving conditions".



RAG shows that Gemini relies on external sources (some are more reliable) to find the answer



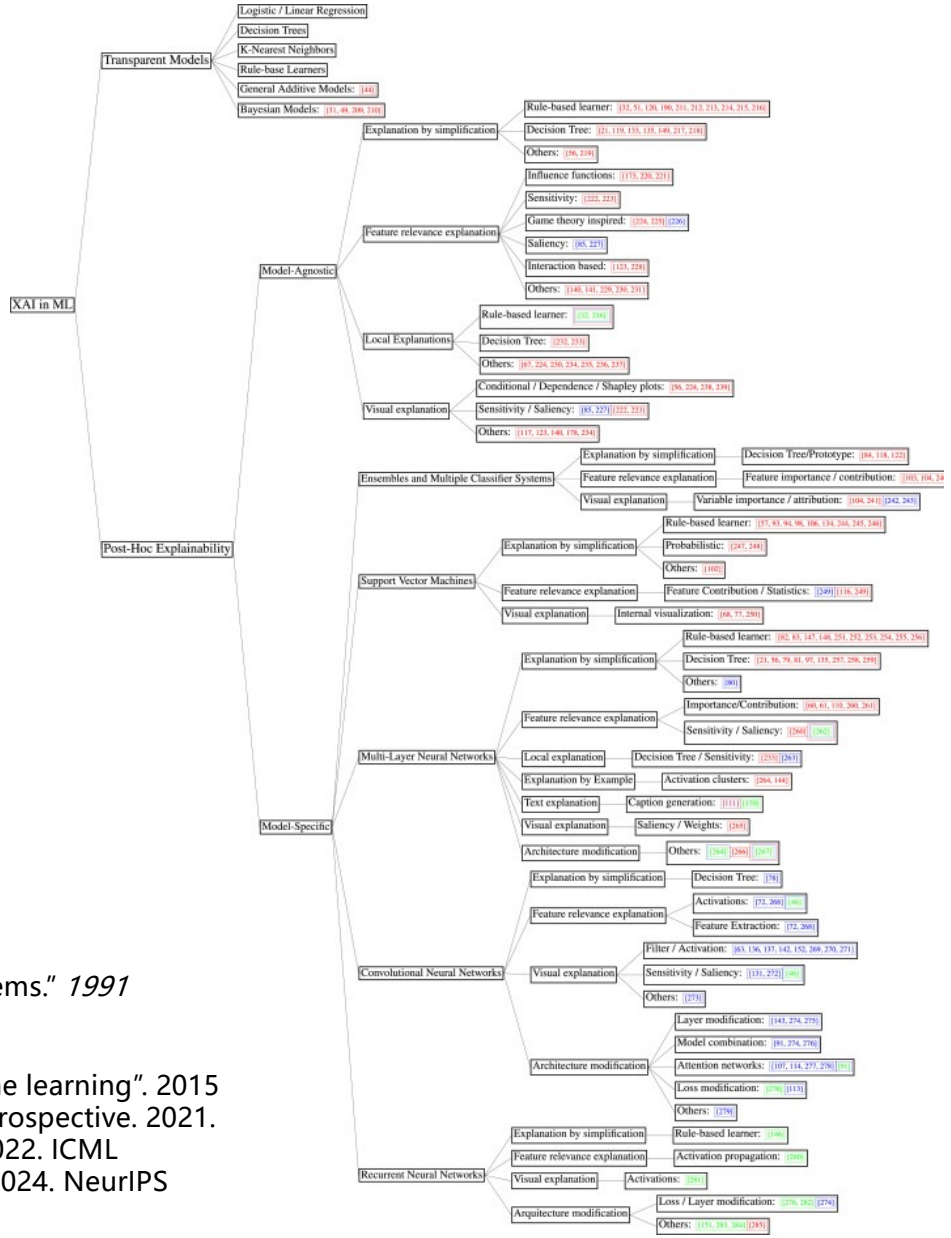
Publications from my group

[1] MBA-RAG: a Bandit Approach for Adaptive Retrieval-Augmented Generation through Question Complexity. AAAI 2025

[2] CogniBench: A Legal-inspired Framework and Dataset for Assessing Cognitive Faithfulness of Large Language Models. ACL 2025.

Explainable AI (XAI)

- 1975: a probabilistic rule-based program for **medical diagnosis**.
- 1991: an "explainable expert system" that helps Lisp **coding**.
- 1994: a "learn to explain" agent for **robotics**.
- 2002: explainable methods for Bayesian networks.
- 2014: explainable CNN for **images classification**.
- 2017: DARPA XAI program starts; **Transformer invented**.
- 2018: visualizing Generative Adversarial Network (GAN).
- 2019: explainable **robot** behaviors.
- 2020: safety and robustness of XAI studied;
  natural language explanations evaluation.
- 2022: explainable Transformer.
- 2023: explainable ViT; **explainable LLM**.
- 2024: explainable Large Multi-modal Model

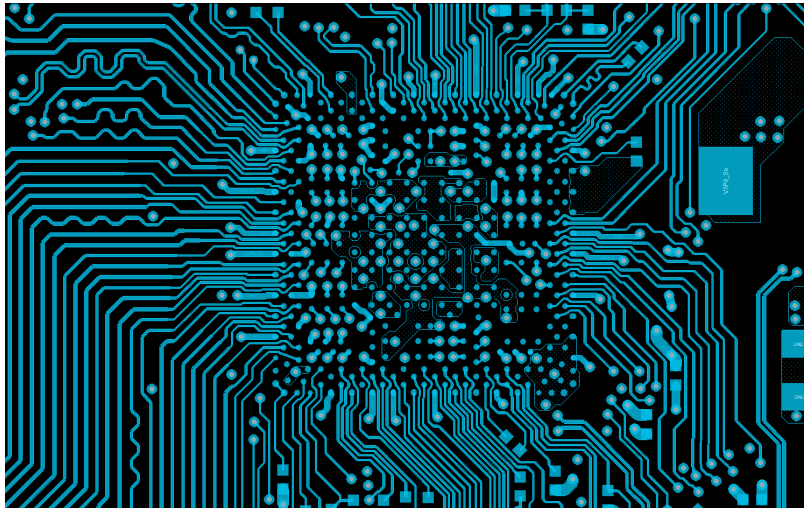


Shortliffe, Edward H. "A model of inexact reasoning in medicine." 1975.
Swartout, William. "Explanations in knowledge systems: Design for explainable expert systems." *1991*
Johnson, W. Lewis. "Agents that Learn to Explain Themselves." 1994.
Zeiler, Matthew D. "Visualizing and understanding convolutional networks." 2014.
Kulesza, Todd, et al. "Principles of explanatory debugging to personalize interactive machine learning". 2015
David Gunning, Eric Vorm, Yunyan Wang, et al. DARPA's Explainable AI (XAI) program: A retrospective. 2021.
Ameen Ali. XAI for Transformers: Better Explanations through Conservative Propagation. 2022. ICML
Jayneel Parekh. A Concept-Based Explainability Framework for Large Multimodal Models. 2024. NeurIPS

My research on XAI

- 2019 ICDM: first paper to explain Probabilistic Graphical Models.
- 2020 CIKM: first paper on Shapley value for graph data.
- 2021 ICDMa: first paper explainig graph topological importance.
- 2024 NeurIPSa: robust explanations via adversarial training.
- 2024 NeurIPSb: explaining reinforcement learning policy on graphs.
- 2025 ICLR: first paper explaining predictions on dynamic graphs.
- 2025 NeurIPS (**just accepted last night!**): theoretical treatment of graph explainability using Graph Geometry.
- On-going: LLM explainability;
- Application: AI for science (medicine, biology, geography)
  AI for EDA (with Huawei)

PUT XAI in action!
How XAI help human designers work faster and better with an LLM copilot.



A taxonomy of traditional XAI techniques

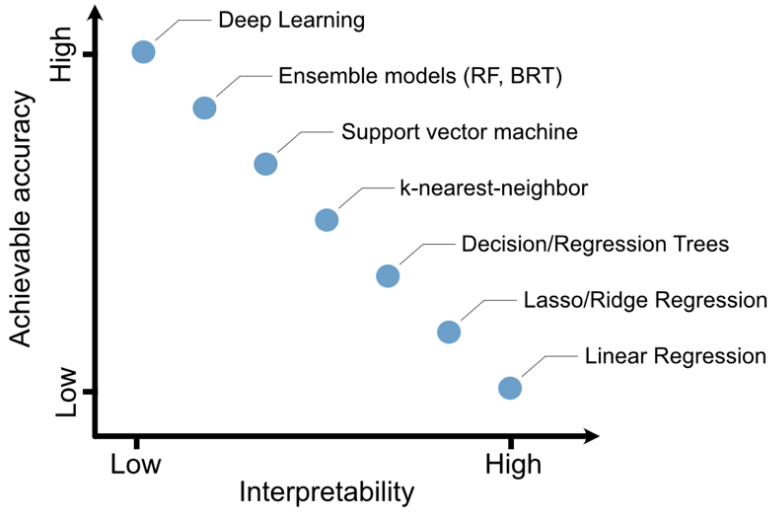
- This taxonomy was first surveyed in 2019[1]
  - Global: regarding the model itself, **regardless of any input**.
  - Local: explanation model's behavior on **a specific input**.
  - Intrinsic: white-box model interpretable **without additional tool**.
  - Post-hod: using **additional tool** to interpret a blackbox model.

Is an explanation about the model or a prediction?

Is the model itself interpretable?

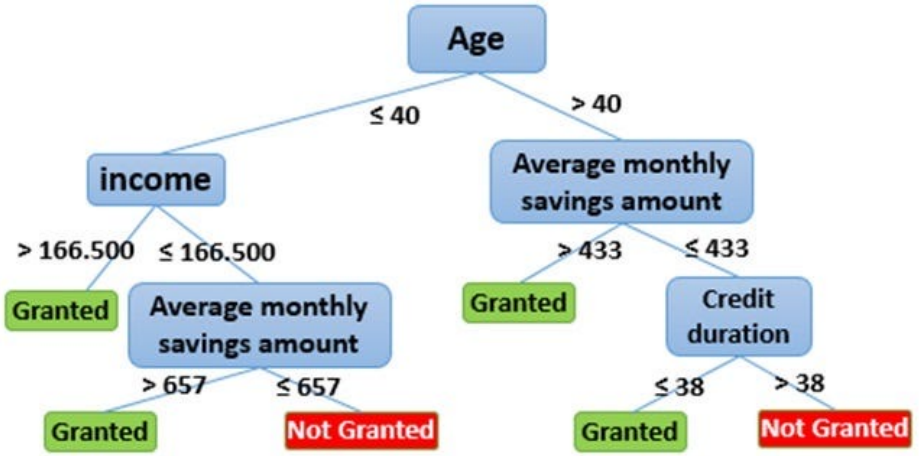
	Global	Local
Intrinsic	• Linear models • sparsity, • explanation-constrained, • capsule networks	• Linear models • attention machenism
Post-hoc	• LIME • Data Shapley • tree-based (coverage), • CNN/RNN (activation maximization)	• LIME • Shapley values (perturbation, counterfactual, contrastive) • Influence scores • Gradient

A trade-off between model accuracy and explainability



Explaining one decision tree

- One decision tree is **globally** and **intrisically** interpretable.
  - Extract rules by enumerating all paths from root to leaves, regardless of any input.



Complexity: depth and number of the leaves

Extracted rules:

- `Age <= 40 & income > 166.5 => Granted`
- `Age <= 40 & income <= 166.5 & AMSA > 657 => Granted`
- `Age <= 40 & income <= 166.5 & AMSA <= 657 => Not Granted`
- `Age > 40 & AMSA > 433 => Granted`
- `Age > 40 & AMSA <= 433 & Credit duration <=38 => Granted`
- `Age > 40 & AMSA <= 433 & Credit duration > 38 => Not Granted`

Why rules are intrinsically interpretable:

Format
1) Logical structure fit human thought process
2) Attributes are grounded and interpretable.

Content
3) Allow a case-by-case reasoning process
4) Allow intervention (counterfactual).

Case-based classifiers

- Use the labels of the nearest neighbors to classify the target instance:
  - Intuition: if an animal has feather like a duck, makes sound like a duck, walks like a duck, it's likely a duck.
  - Need to choose a similarity metric
    - Linear or non-linear
    - Can combined with metric learning or representation learning.
  - Example classifiers: kNN, SVM

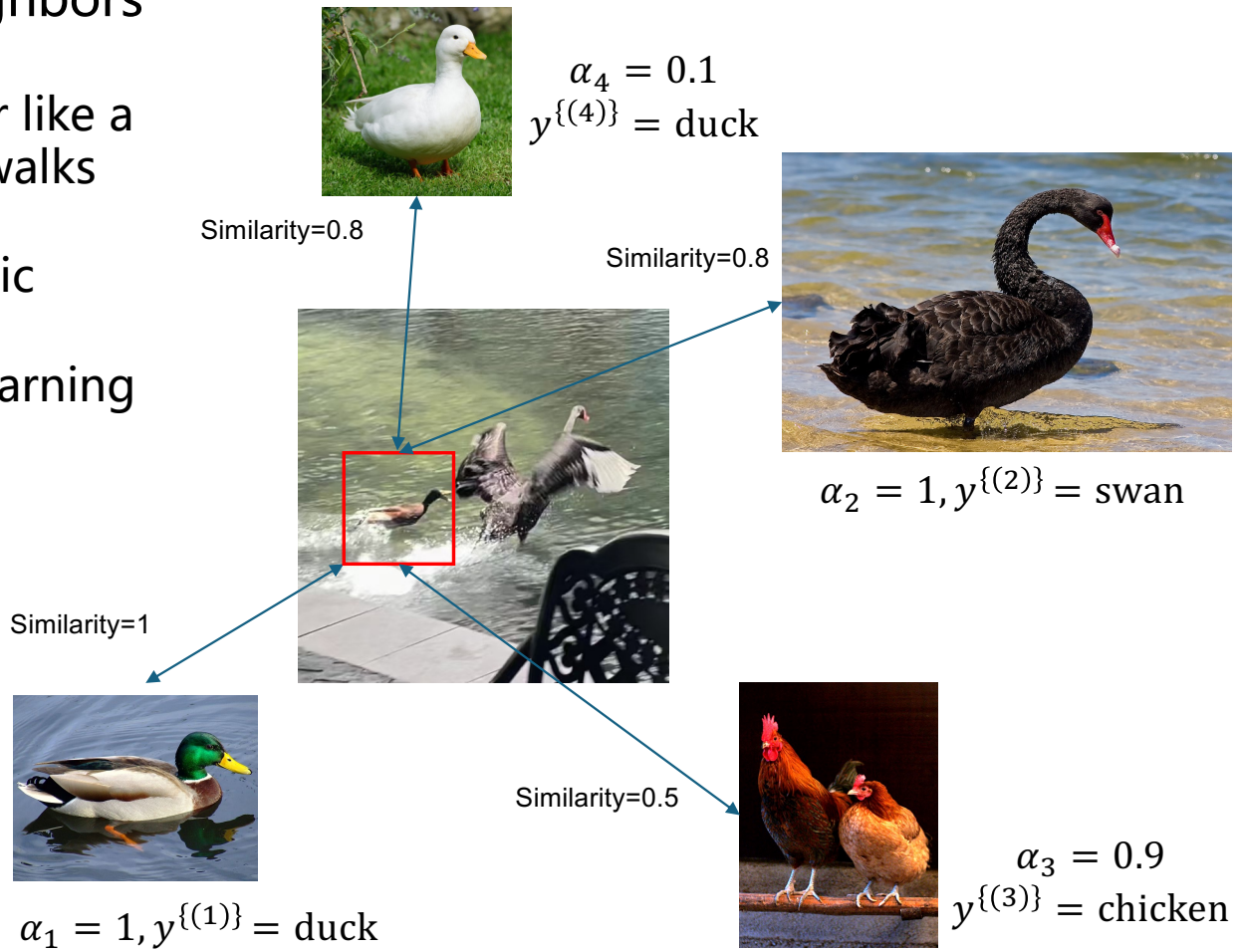
Examples from all possible classes

$$f(\mathbf{x}) = \sum_{i=1}^{n} \alpha_i y^{(i)} k(\mathbf{x}^{(i)}, \mathbf{x}) + b$$

labels

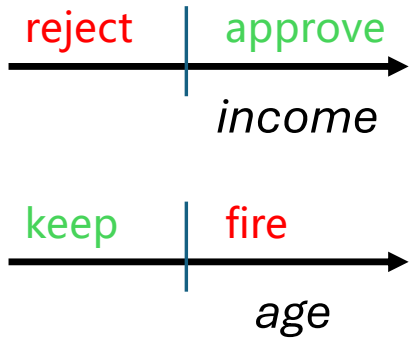
importance

kernel function (similarity)



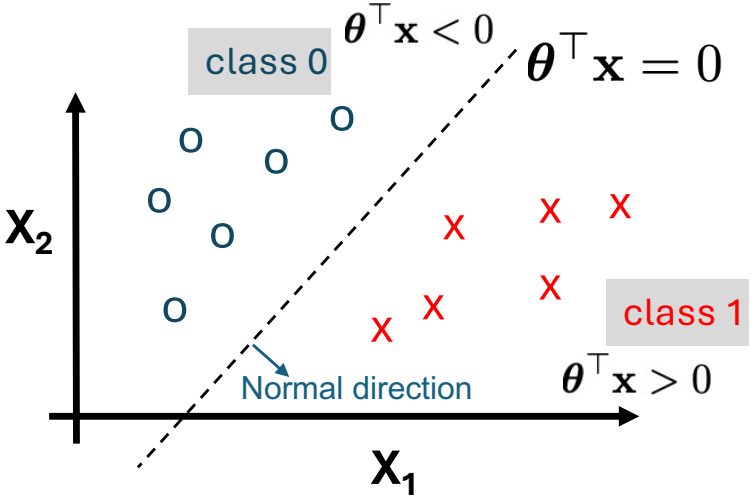
Binary linear classification

- A simple linear classifier
  - $h_{\theta=\{0.75,10000\}}(income) = \begin{cases} \text{reject } if\ 0.75 * income - 10000 < 0 \\ \text{approve } otherwise \end{cases}$
  - $h_{\theta=\{35\}}(age) = \begin{cases} \text{keep } if\ age - 35 < 0 \\ fire\ otherwise \end{cases}$



- General Linear function: $h_{\boldsymbol{\theta}}(\mathbf{x}) = \boldsymbol{\theta}^\top \mathbf{x} = \theta_0 + \theta_1 x_1 + \cdots + \theta_n x_n$ $\quad \boldsymbol{\theta} = [\theta_0, \ldots, \theta_n]^\top$
- $\boldsymbol{\theta}^\top \mathbf{x}$ increases in the normal direction.
- Classification using a linear function:

$$h_{\boldsymbol{\theta}}(\mathbf{x}) = \text{sign}(\boldsymbol{\theta}^\top \mathbf{x}) = \begin{cases} 0, & \text{if } \boldsymbol{\theta}^\top \mathbf{x} < 0 \\ 1, & \text{if } \boldsymbol{\theta}^\top \mathbf{x} > 0 \end{cases}$$

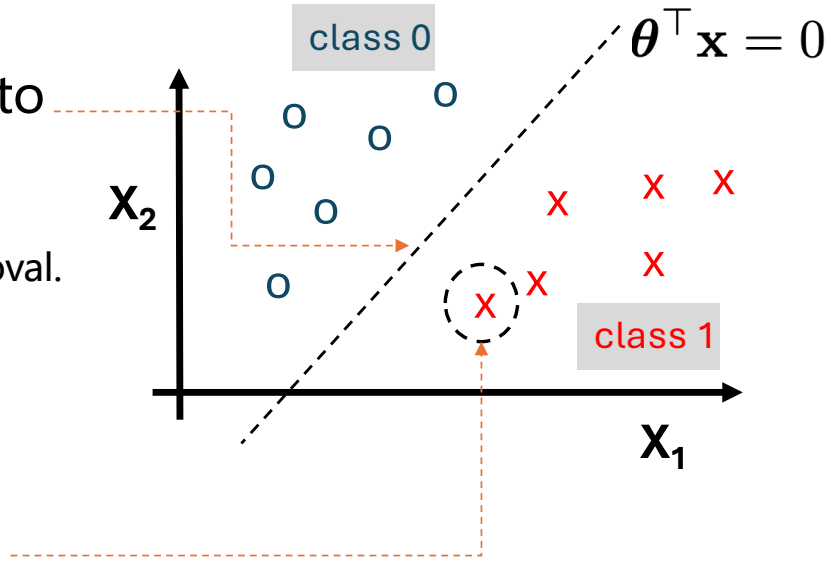


Explaining binary linear models

- Motivation: why simple linear models are still used?
  - In critical domains, such as finance/medicine, linear model provide sufficient accuracy and interpretability is of high importance.
  - Linear model is a building-block of highly complex models include Transformers.

- Global explanation: which variable is important to the model, regardless of the input data **x**.
  - Check if it aligns with domain knowledge
    - Annual income should be important in credit card approval.
  - Help users build trust on the model as a whole.

- Local explanation: which input variable is important to the model at a fixed input data **x**.
  - Input **x** provides more contextual information, more precise to the prediction on the input **x.**

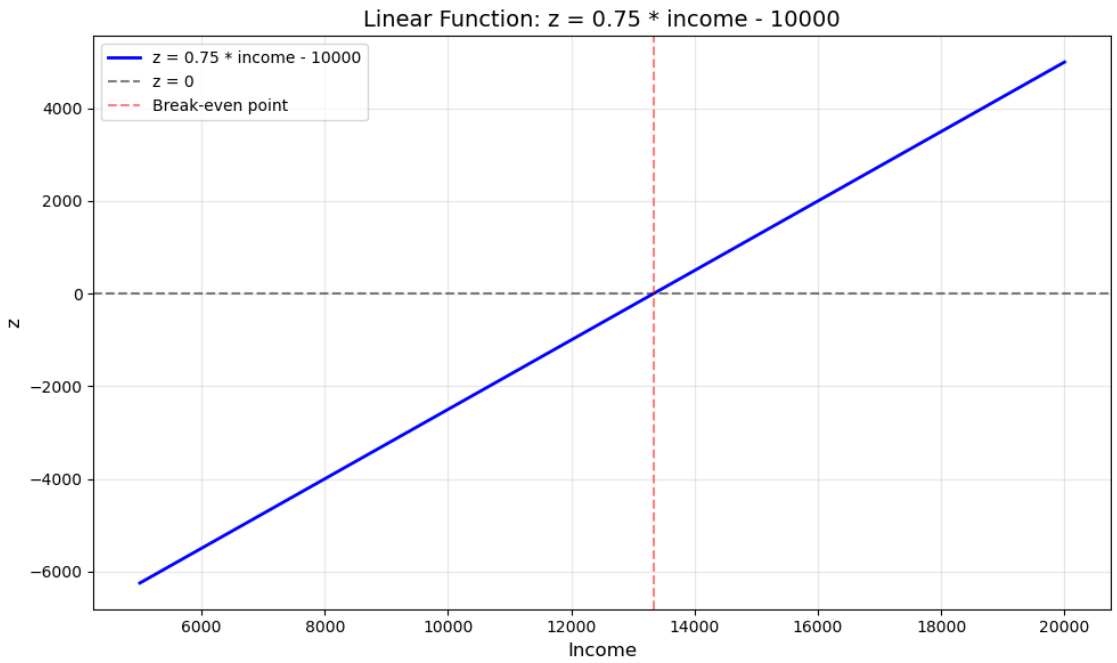


Explaining binary linear models

- Globally explaining $z = \boldsymbol{\theta}^\top \mathbf{x} = \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n$
  - For *any* $\mathbf{x}$, the absolute value $|\theta_i|$ determines the importance of feature x_i .
  - The sign of θ_i determines whether the feature x_i positively or negatively contribute to for *any* values of x_i .
  - Note that global explanation does not depends on x_i .
  - Example
    - x_1 indicates {is a smoker}, x_2 indicates {weight is normal}, z indicates degree of heart attack
    - $\boldsymbol{\theta} = [1, -1]$ are the model parameters
    - Globally, *regardless of individual person*, smoking increases the chance of heart attack, while keeping a normal weight reduces such attack.
    - This does not describe the details of a specific person.

Explaining binary linear models

- Another example
  - Global Explanation: *the higher the income, the better the chance of approval: 0.75 indicates the positive relationship between the two factors.*



  - $h_{\theta=\{0.75,10000\}}(income)$

$$= \begin{cases} \text{reject if } 0.75 * income - 10000 < 0 \\ \text{approve otherwise} \end{cases}$$

- We can make it more formal
  - $\frac{\partial h_\theta(\boldsymbol{x})}{\partial x_i} = \theta_i$: the gradient of the classifier with respect to the input feature x_i is the **directional importance** of the feature.
  - The absolute gradient indicates the **absolute importance**.

Explaining binary linear models

- Locally explaining $z = \boldsymbol{\theta}^\top \mathbf{x} = \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n$
  - For *an* $\mathbf{x}$, the component $\theta_i x_i$ determines contribution of x_i to the output.
  - If x_i and θ_i have the same sign, then $\theta_i x_i$ have positive contribution.
  - Example

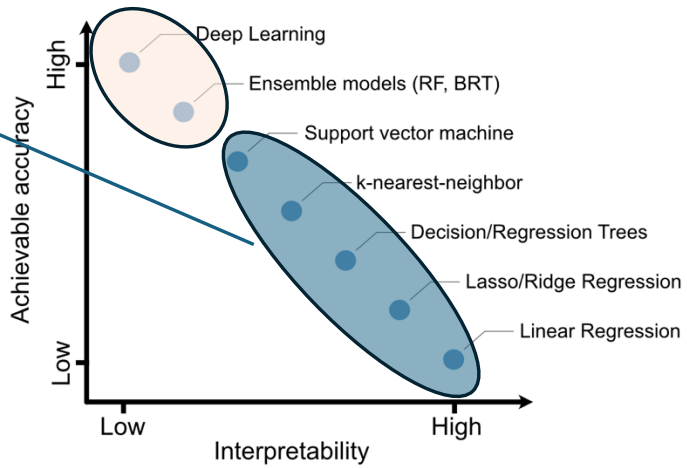
$$h_{\theta=\{0.75,10000\}}(income) = \begin{cases} \text{reject if } 0.75 * income - 10000 < 0 \\ \text{approve otherwise} \end{cases}$$

    - With a positive (negative, resp.) income, the income has positive (negative, resp.) contribution to the approval.

  - Why this is a local explanation: for a different person applicant, the component $\theta_i x_i$ will differ, and thus depends on the input $\mathbf{x}$.
  - Gradient multiplied the input feature $\frac{\partial h_\theta(\boldsymbol{x})}{\partial x_i} x_i = \theta_i x_i$ is a local importance score.

Post-hoc explanation methods

- *Intrinsic* explainability applies to:
  - A single decision tree: rule extraction,
  - kNN and SVM: case-based,
  - Linear models: feature importance.
- Certain models are not *intrinsically* explainable:
  - Tree ensemble, very deep models, AIGC (LLM, diffusion)
- Two post-hoc methods
  - Model approximation
    - Train a simpler model using a larger model's output
    - Trade accuracy for more simplicity
    - Counting: tree ensembles
    - Gradients: deep models, simple transformers.
  - Counterfactual thinking
    - Perturbation-based: Shapley, LLM,



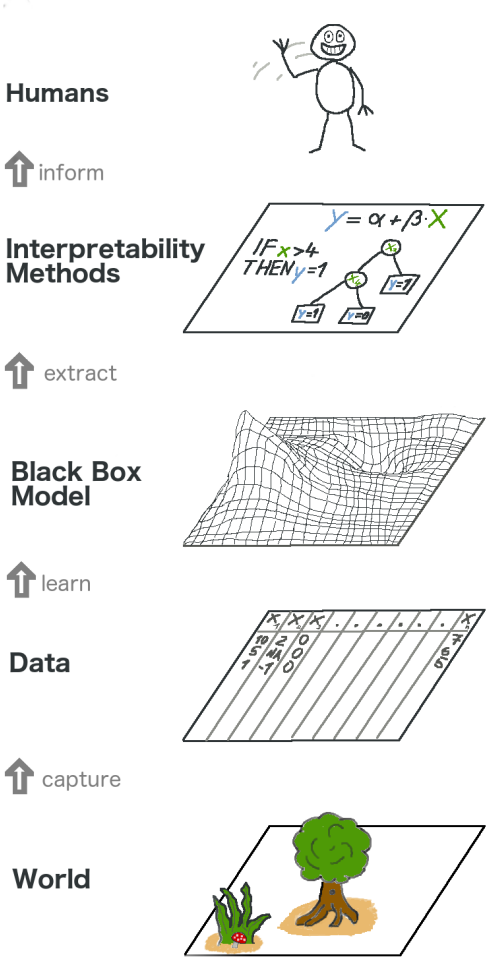
Interpretability via model extraction. 2017.

Model approximation

- **Target model**: used to make accurate predictions in applications, but complex and less interpretable.
- Two sources of complexity:
  - Target model prediction depends on many input features.
  - Features interactions (both low- and high-orders)
  - Example: “stock price” depends on “company size”, “earning”, “market demands”, while “size” and “earning” are correlated.

- To explain is to reduce complexity via approximation:
  - Reveal the relationship between ***a feature*** and model output
  - Example: find the stock price – “company size” relationship.

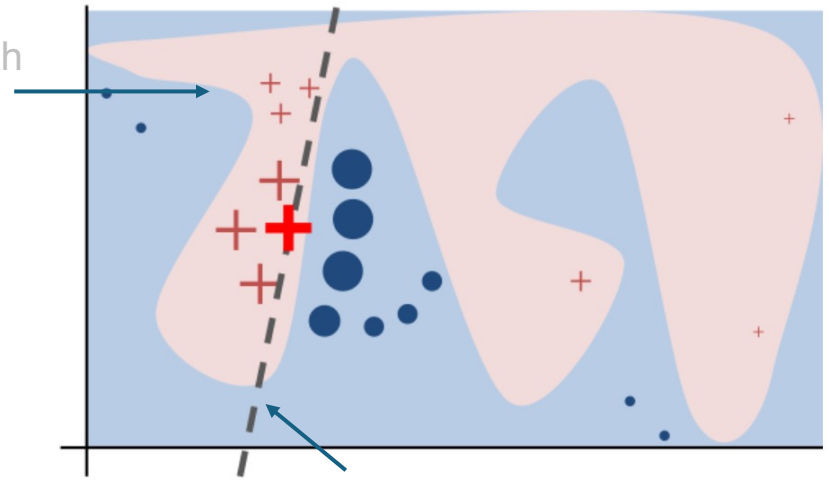
- **Approximating model**: facing human users and must be simple and interpretable.



LIME

- Support local and global explanations:
  - **Local explanation**: train a sparse linear model to approximate the decision boundary of the target model at input $\mathbf{x}$.
  - **Global explanation**: find a subset of *B* training instances separable by important features, using local models' feature weights as importance scores.
- Allow any number of features in the approximation.
- Model-agnostic: **any** target models can be explained.

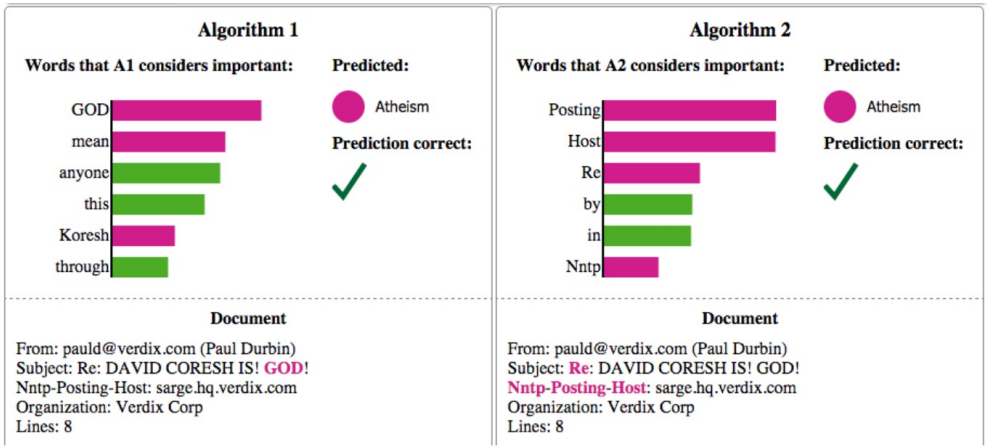
Target model:
Deep nonlinear model with curvy decision boundary



Approximating model:
Linear model with low global accuracy, but accurate locally.

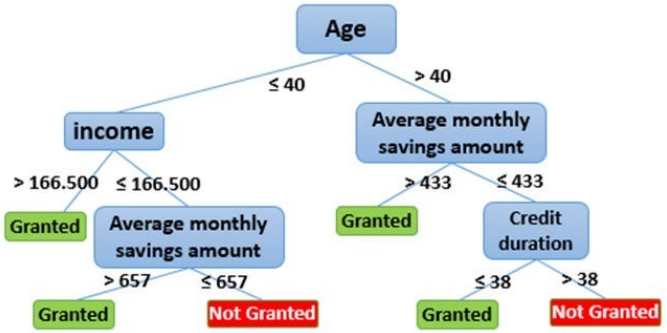
Local explanations for two prediction algorithms

Same input:



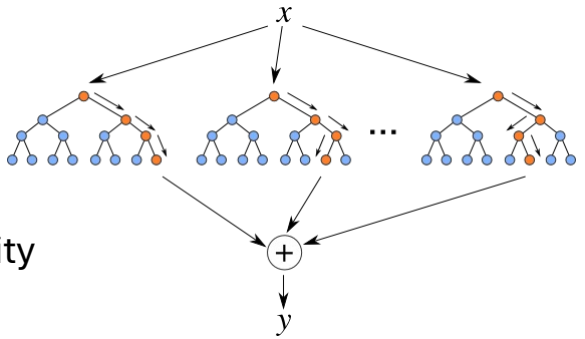
Explaining many trees

- One tree does not offer high accuracy
  - Prediction = averaging many (e.g., 1000) trees' predictions[1]
  - One tree is intrinsically interpretable, but many trees are not

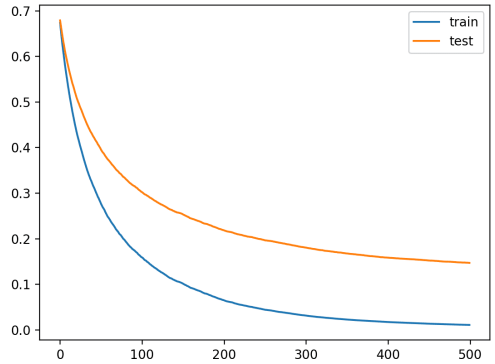


Higher accuracy

Lower interpretability



Accuracy



Number of trees

- Global interpretation using feature importance
  - Example: the most distinguishing feature to separate high- and low-risk loan applicants is how much money in accounts.
  - Count of usage of a feature: how often a feature is used to partition samples.
  - Sum of a feature's information gain: power for partition sample into pure classes (sort of a feature ranking method).

[1] Breiman, Leo. 2001a. "Random Forests." *Machine Learning* 45 (1). Springer: 5–32.

Explaining many trees

- Permutation Feature Importance (PFI)[1]

  Given an ensemble of trees $f(\mathbf{x})$, a dataset X and ground truth labels y

  1. Perturb a feature randomly on a set of data;
  2. Make predictions on the perturbed data;
  3. Repeat 1-2 and evaluate the *average* change in Mean-Absolute-Error (MAE).
  4. The **higher** averge MAE, the more important the feature.

- Pros:
  - No need of model re-training to save time
- Cons:
  - Don't know how to perturb the values
  - May lead to unrealistic cases.
    - For example, age = 5 and income > 50000.

Evaluate the importance of the feature x_1

Original data set

	x_1	x_2	...	x_p	y	$\hat{y}$
1	2	0.2		female	1	1
2	8	0.6		male	0	0
3	7	0.5		male	1	0
4	3	1.1		female	0	0
5	14	0.8		female	1	1
...						
n	11	0.4		male	1	1

Data set with permuted covariate x_1

	x_1	x_2	...	x_p	y	$\hat{y}$
1	14	0.2		female	1	0
2	11	0.6		male	0	1
3	2	0.5		male	1	1
4	7	1.1		female	0	1
5	3	0.8		female	1	0
...						
n	8	0.4		male	1	1

$$PFI_1 = L(y, f(X_{\text{perm},1})) - L(y, f(X))$$

Evaluate the importance of the feature x_p='sex'

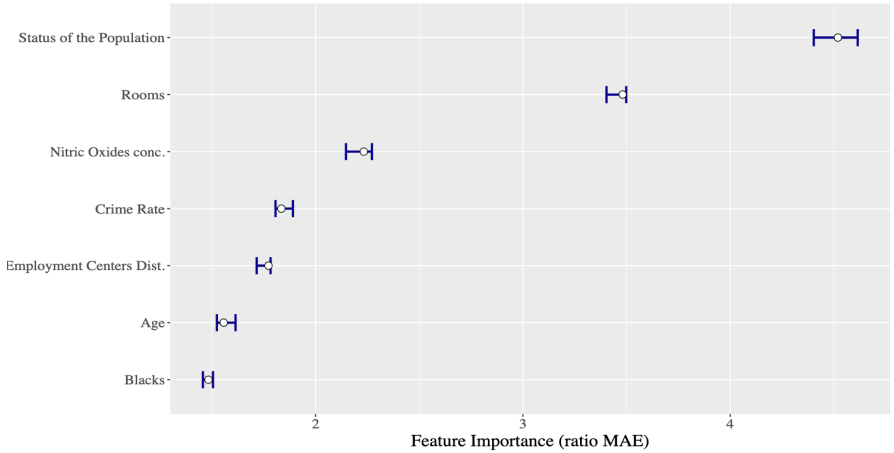
Original data set

	x_1	x_2	...	x_p	y	$\hat{y}$
1	2	0.2		female	1	1
2	8	0.6		male	0	0
3	7	0.5		male	1	0
4	3	1.1		female	0	0
5	14	0.8		female	1	1
...						
n	11	0.4		male	1	1

Data set with permuted covariate x_p

	x_1	x_2	...	x_p	y	$\hat{y}$
1	2	0.2		male	1	1
2	8	0.6		female	0	0
3	7	0.5		male	1	0
4	3	1.1		male	0	1
5	14	0.8		female	1	1
...						
n	11	0.4		female	1	0

$$PFI_p = L(y, f(X_{\text{perm,p}})) - L(y, f(X))$$



[1] Breiman, Leo. 2001a. "Random Forests." *Machine Learning* 45 (1). Springer: 5–32.

Leave-one-out score

- Permutation Feature Importance (PFI)[1]

  Given an ensemble of trees, denoted by $f(\mathbf{x})$, a dataset X and ground truth labels y

  1. Remove each feature from X and re-fit the model (why?).
  2. Evalute the change in the training error.
  3. The higher the error change, the more important the feature.

- Cons
  - Need time-consuming model retraining.
  - Not explaining the original set of trees.
  - Ignore feature interactions (*smoke+overweight=>heart attack*)
  - What if two features are exactly the same?

- Alternatively, just set a feature to zero or its mean value to remove information about the feature to avoid model re-training.

[1] Ideas on Interpreting Machine Learning. 2017

Original data set

	x_1	x_2	...	x_p	y	$\hat{y}$
1	2	0.2		female	1	1
2	8	0.6		male	0	0
3	7	0.5		male	1	0
4	3	1.1		female	0	0
5	14	0.8		female	1	1
...						
n	11	0.4		male	1	1

Data set without covariate x_1

	x_2	...	x_p	y	$\hat{y}_{-1}$
1	0.2		female	1	1
2	0.6		male	0	1
3	0.5		male	1	1
4	1.1		female	0	1
5	0.8		female	1	1
...					
n	0.4		male	1	0

$$L(y, f_{-1}(X_{-1})) - L(y, f(X)) = FI_1$$

Original data set

	x_1	x_2	...	x_p	y	$\hat{y}$
1	2	0.2		female	1	1
2	8	0.6		male	0	0
3	7	0.5		male	1	0
4	3	1.1		female	0	0
5	14	0.8		female	1	1
...						
n	11	0.4		male	1	1

Data set without covariate x_p

	x_1	x_2	...	x_{p-1}	y	$\hat{y}_{-p}$
1	2	0.2			1	0
2	8	0.6			0	0
3	7	0.5			1	0
4	3	1.1			0	0
5	14	0.8			1	1
...						
n	11	0.4			1	0

$$L(y, f_{-p}(X_{-p})) - L(y, f(X)) = FI_p$$

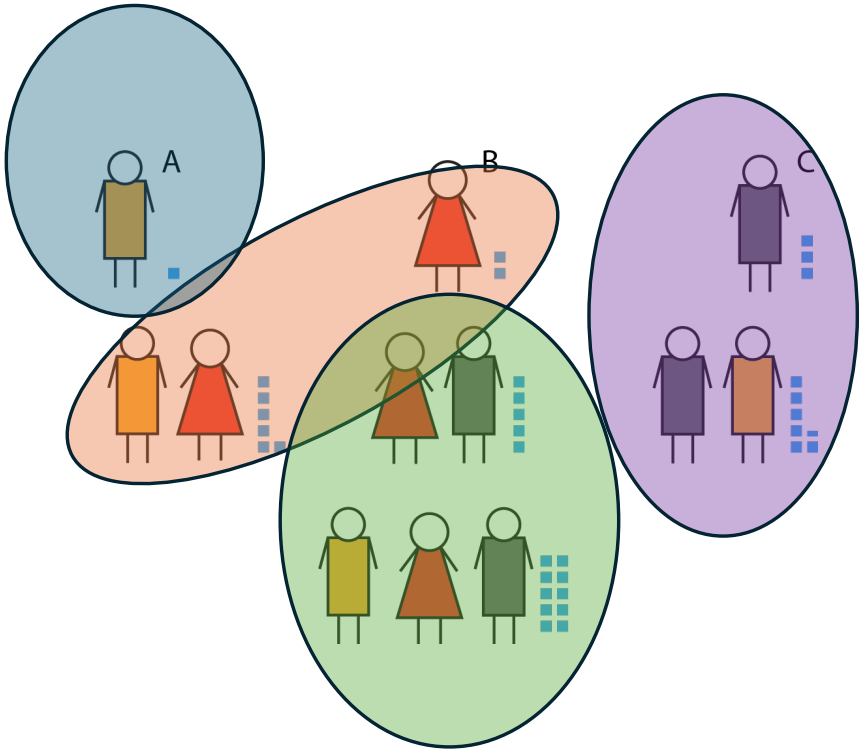
Shapley values

- It was originally invented in 1953[1] to **fairly** (no more, no less) evaluate the importance of team members.
- In a game (e.g., final project), there are 3 players (students). How to evaluate each student's contribution?
- Each student can work alone or with any others on the project. Any team configuration can generate some value.
  - Differentiate the cases where A presents or absents.
  - V({A}) – V({}) = 1-0 = 1
  - V({B,A}) – V({B}) = 6-2 = 4
  - V({C,A}) – V({C}) = 6.5-3 = 3.5
  - V({B,C, A}) – V({B,C}) = 10-5 = 5

Any one of these differences won't fairly evaluate A's contribution.

- The contribution of A is (1+4+3.5+5)/4
- If A is important (e.g., A is the only one who can code), without him/her leads to a large difference.
- If A is unimportant (e.g., replaceable), without him/her leads to a small difference.

A game of 3 players and all possible team configuation and the outcome (indicated by the number of little blue squares). How much a player values?

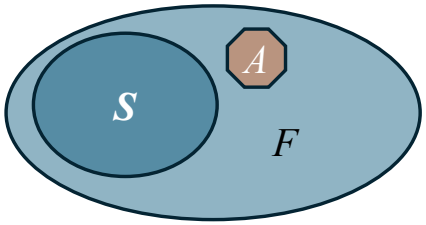


[1] Shapley, L. S. A value for n-person games. *Contributions to the Theory of Games*, 2(28):307–317, 1953.

Shapley values for feature importance

- Given any feature subset S, Model $f(\mathbf{x}_S)$ is trained on feature subset S and makes predict using the feature subset S .
- Shapley value of feature A (a player) is defined as

$$\phi_A = \sum_{S \subset F \setminus \{A\}} \frac{|S|!(|F| - |S| - 1)!}{|F|!} [f(\mathbf{x}_{A \cup \{A\}}) - f(\mathbf{x}_S)]$$



Normalizing by how likely a specific configuration will happen: the more frequent, downweight the **difference**.

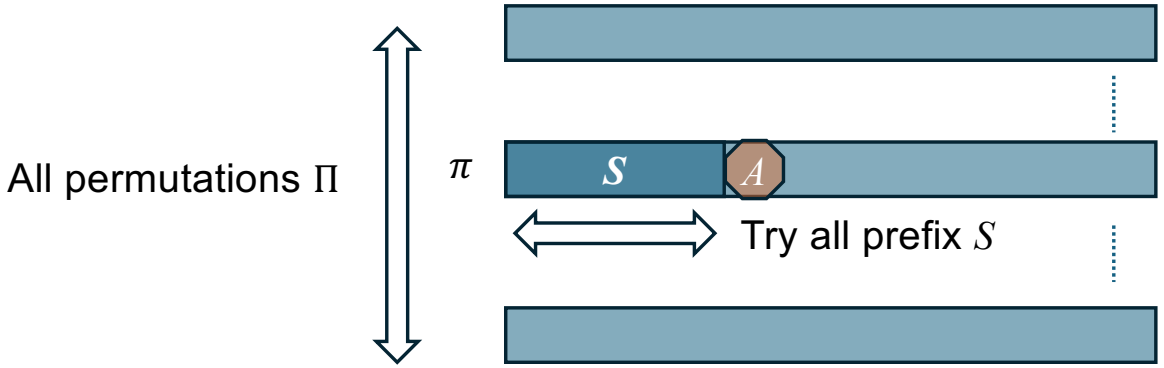
Kononenko, I. et al. An efficient explanation of individual classifications using game theory. Journal of Machine Learning Research
Fatima, S. S., Wooldridge, M., and Jennings, N. R. A linear approximation method for the shapley value. Artificial Intelligence, 172(14):1673–1699, 2008.

Shapley values for feature importance

- It is usually infeasible to find the exact Shapley values and sampling is needed.
- It can also be calculated as $\phi_A = \frac{1}{n!} \sum_{\pi \in \Pi} [f(\mathbf{x}_{S_A(\pi) \cup \{A\}}) - f(\mathbf{x}_{S_A(\pi)})]$

All features upto A (**inclusive**) in the permutation π

All features upto A (**exclusive**) in the permutation π



1. Can randomly shuffle the players to sample one $\pi \in \Pi$
2. Sequentially scan the items in π to find S and A

Kononenko, I. et al. An efficient explanation of individual classifications using game theory. Journal of Machine Learning Research
Fatima, S. S., Wooldridge, M., and Jennings, N. R. A linear approximation method for the shapley value. Artificial Intelligence, 172(14):1673–1699, 2008.

Data Shapley

- The same idea can be used to evaluate data point importance, thus generating case-based explanations.
  - A data point $\mathbf{x}^{(i)}$ can contribute to the final model test accuracy along with ***any*** data subset S and its overall contribution is

$$\phi_i = C \sum_{S \subset D \setminus \{\mathbf{x}^{(i)}\}} [V(S \cup \mathbf{x}^{(i)}) - V(S)]$$

V is a contribution function, such as model accuracy, to measure reduction in accuracy due to missing a data point.

- It can also be calculated by enumerating all permutations of the data points:

$$\phi_i = \mathbb{E}_{\pi \sim \Pi}[V(S_\pi^i \cup \{i\}) - V(S_\pi^i)]$$

- The above re-formulation allows an online Monte-Carlo algorithm:
  1. Randomly shuffle the whole dataset and then scan it sequentially,
  2. For the next data point $\mathbf{x}^{(i)}$, update the model parameter using information in $\mathbf{x}^{(i)}$,
  3. Then evalute the change in test accuracy.

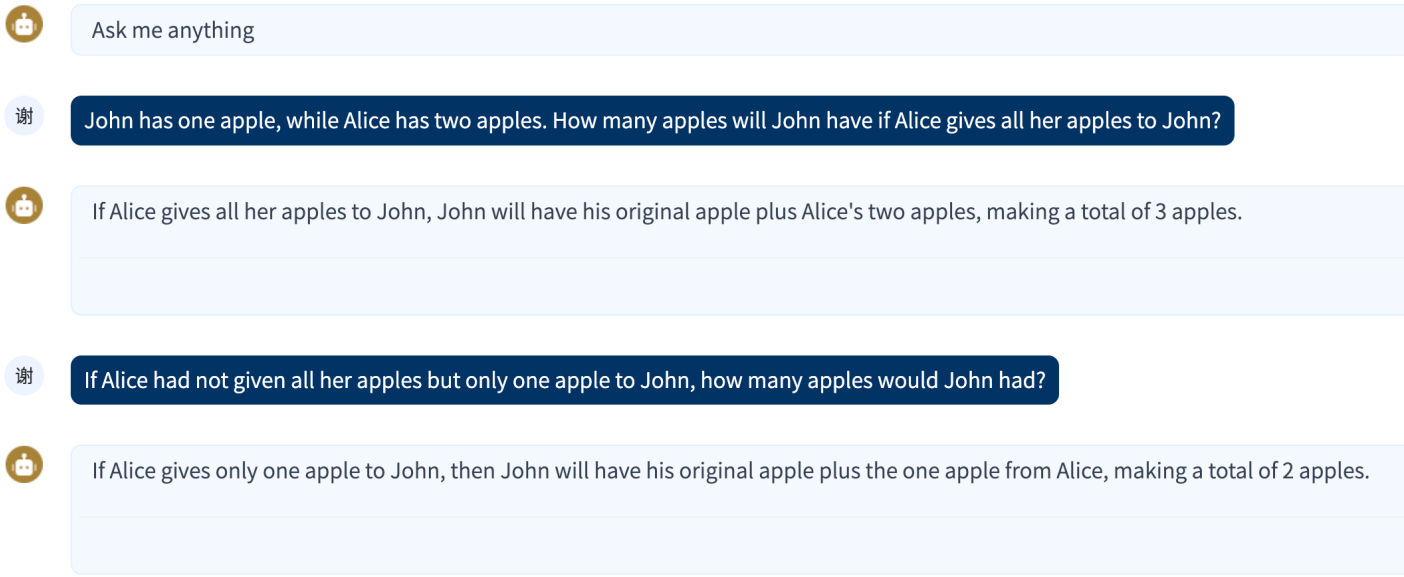
Data Shapley: Equitable Valuation of Data for Machine Learning. ICML 2019

Counterfactual explanation

- A counterfactual explanation describes a causal situation.

> If X had not occurred, Y would not have occurred

- It is a way of thinking, and can be local or global explainability.
  - Example:



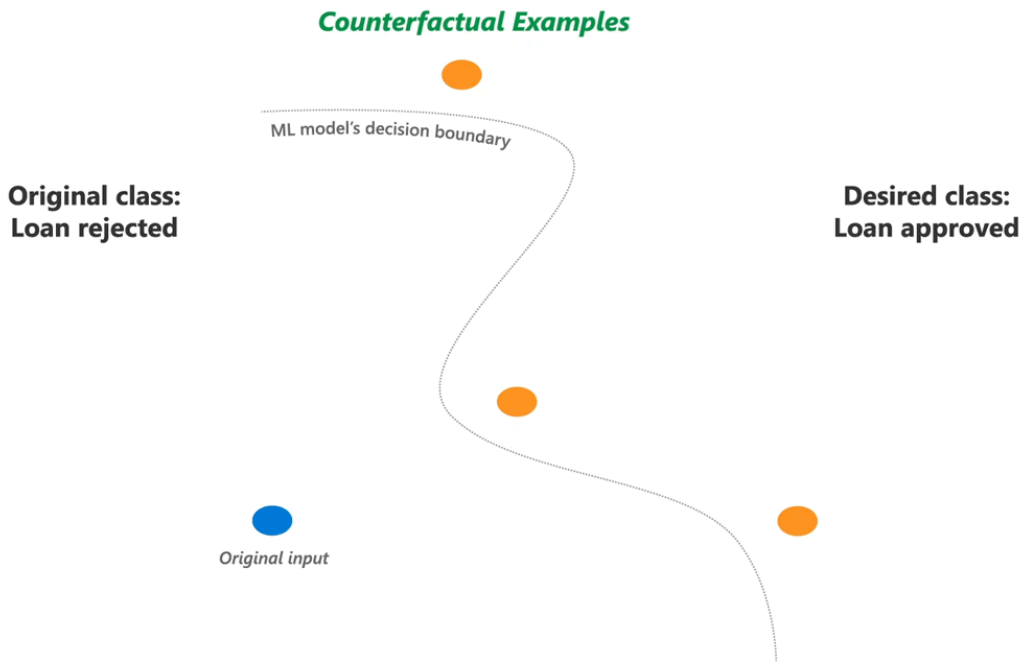
Counterfactual explanation

- Counterfactual explanation: find a counterfactual example whose predicted class flips to another class
  - **Example**: Peter applies for a loan and gets rejected by AI. He change his spending behaviors to improve his chances.

- Ask for minimal changes to the original input to be relevant
  - A very different case will naturally be predicted to a different class, thus not probing the model's local behavior near the original input.
  - In practice, minimal change give users actionable suggestions: increase your annual income by 10000 RMB, rather than 1000000 RMB.
- Multiple diverse counterfactual explanations (Rashomon effect)
  - Example: change spending, address, saving amount, income, etc.

- Reasonable values: change size from 100 to 10 m^2 while keeping 3 bedrooms.

Counterfactual explanation: visualization



Source: https://medium.com/@bijil.subhash/explainable-ai-diverse-counterfactual-explanations-dice-315f058c0364

Counterfactual explanation: optimization

- Method by Wachter et al. [1]
  - Minimize $L(x, x', y', \lambda) = \lambda \cdot (\hat{f}(x') - y')^2 + d(x, x')$
  - Algorithm:
    1) Initialize a counterfactual example x' and some λ
    2) Optimize x' by minimizing L
    3) While not satisfied $|\hat{f}(x') - y'| > \epsilon$, increase λ (so to focus more on reducing the first term)
  - Does not consider reasonable data values and number of changed features
- Method by Dandl et al. [2]
  - Minimizing multiple objectives $L(x, x', y', X^{obs}) = \left(o_1(\hat{f}(x'), y'), o_2(x, x'), o_3(x, x'), o_4(x', X^{obs})\right)$

$$o_3(x, x') = ||x - x'||_0 = \sum_{j=1}^{p} \mathbb{I}_{x'_j \neq x_j}.$$

Change as less number of features as possible, so that the change is feasible.

$$o_4(x', \mathbf{X}^{obs}) = \frac{1}{p} \sum_{j=1}^{p} \delta_G(x'_j, x_j^{[1]})$$

Close to the most similar observed data, making counterfactual realistic.

[1] Wachter, Sandra, Brent Mittelstadt, and Chris Russell. “Counterfactual explanations without opening the black box: Automated decisions and the GDPR.” (2017).↩
[2] Dandl, Susanne, Christoph Molnar, Martin Binder, Bernd Bischl. “Multi-objective counterfactual explanations”. 2020.↩

Counterfactual explanation: example

Input x

age	sex	job	housing	savings	amount	duration	purpose
58	f	unskilled	free	little	6143	48	car

Approval probability 24.2 %

Diverse multiple Counterfactuals x'

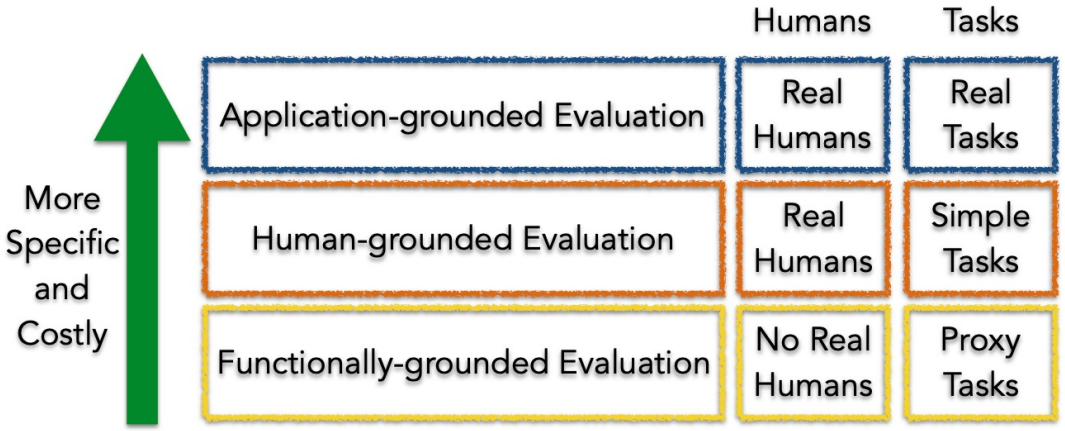
Not realistic changes

Too many changes

age	sex	job	amount	duration	o_2	o_3	o_4	Approval probability $\hat{f}(x')$
		skilled		-20	0.108	2	0.036	0.501
		skilled		-24	0.114	2	0.029	0.525
		skilled		-22	0.111	2	0.033	0.513
-6		skilled		-24	0.126	3	0.018	0.505
-3		skilled		-24	0.120	3	0.024	0.515
-1		skilled		-24	0.116	3	0.027	0.522
-3	m			-24	0.195	3	0.012	0.501
-6	m			-25	0.202	3	0.011	0.501
-30	m	skilled		-24	0.285	4	0.005	0.590
-4	m		-1254	-24	0.204	4	0.002	0.506

Explanation evaluation metrics

Taxonomy of evaluation approaches for interpretability[1]



Examples:

Doctors + deployed AI in a hospital with a patient

Doctors judge explanations in office without patients.

Machines calculate metrics for explanations.

A research paper usually have a mixture of the last two approaches, while the application-grounded evaluation is too costly to perform on a regular basis.

[1] Towards A Rigorous Science of Interpretable Machine Learning. 2017

Real human evaluation on simple tasks

- Evaluating decision sets [1] and lists [2], which can be learned

If Respiratory-Illness=Yes **and** Smoker=Yes **and** Age≥ 50 **then** Lung Cancer

If Risk-LungCancer=Yes **and** Blood-Pressure≥ 0.3 **then** Lung Cancer

If Risk-Depression=Yes **and** Past-Depression=Yes **then** Depression

If BMI≥ 0.3 **and** Insurance=None **and** Blood-Pressure≥ 0.2 **then** Depression

If Smoker=Yes **and** BMI≥ 0.2 **and** Age≥ 60 **then** Diabetes

If Risk-Diabetes=Yes **and** BMI≥ 0.4 **and** Prob-Infections≥ 0.2 **then** Diabetes

If Doctor-Visits ≥ 0.4 **and** Childhood-Obesity=Yes **then** Diabetes

If Respiratory-Illness=Yes **and** Smoker=Yes **and** Age≥ 50 **then** Lung Cancer

Else if Risk-Depression=Yes **then** Depression

Else if BMI ≥ 0.2 **and** Age≥ 60 **then** Diabetes

Else if Headaches=Yes **and** Dizziness=Yes, **then** Depression

Else if Doctor-Visits≥ 0.3 **then** Diabetes

Else if Disposition-Tiredness=Yes **then** Depression

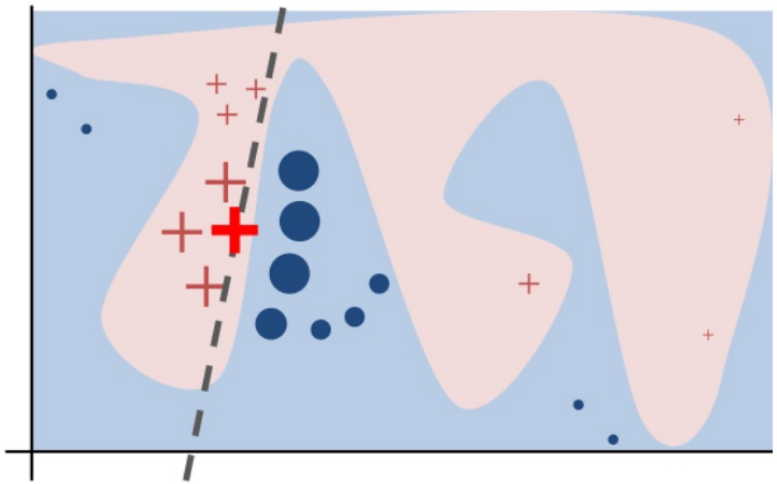
Else Diabetes

- Human evaluation of ***global*** rules: ask humans to write a description of the classifier based on the rules, then experts evaluate the description to get an correct/incorrect answer. Shorter time, shorter descriptions, and higher accuracy are better.
- Human evaluation of ***local*** rules: ask humans to use the rules to decide if a simulated patient has a disease given some feature values. Shorter time and higher accuracy are better.
- **Conclusions: decision sets lead to better human evaluation performance.**

[1] Interpretable Decision Sets: A Joint Framework for Description and Prediction. KDD 2016
[2] Learning decision lists. 2017

Quantitative evaluation

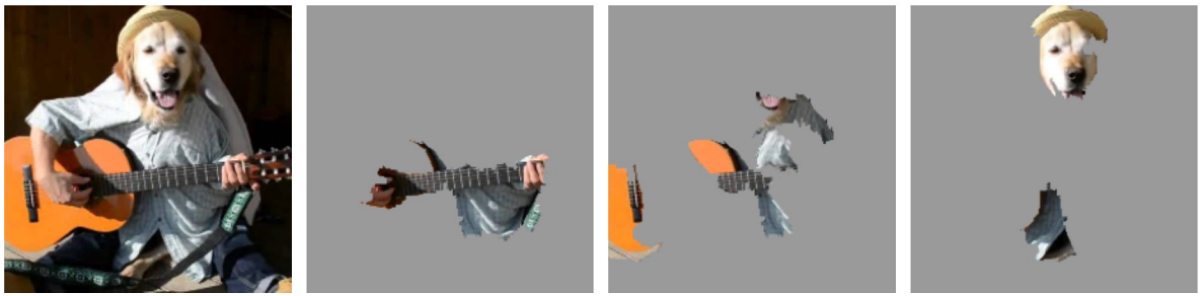
- Human-grounded evaluation cannot be reproduced and cannot be scaled.
- Need functionally-grounded evaluation, using quantitative metrics:
  - **Faithfulness**: how close the explanation can approximate the predictions of the target model.
  - **Complexity**: how many explaining elements are used and how they are fit together.



[1] Towards A Rigorous Science of Interpretable Machine Learning. 2017
[2] "Why Should I Trust You?"- Explaining the Predictions of Any Classifier. KDD 2016

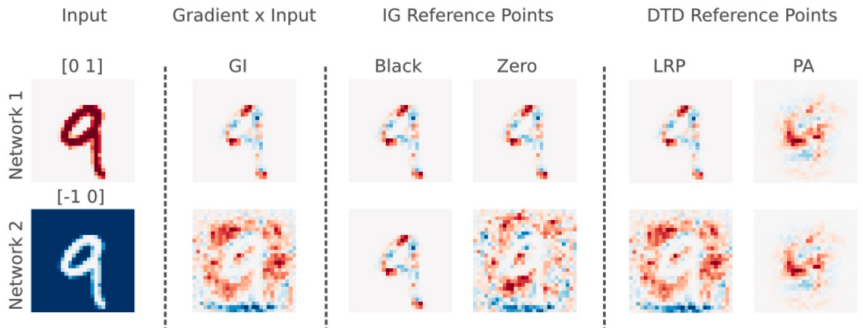
Quantitative evaluation

- Need functionally-grounded evaluation, using quantitative metrics:
  - Counterfactual: change the identified elements in the explanation to change the model output. Help establish a causal-outcome relationship.



(a) Original Image (b) Explaining *Electric guitar* (c) Explaining *Acoustic guitar* (d) Explaining *Labrador*

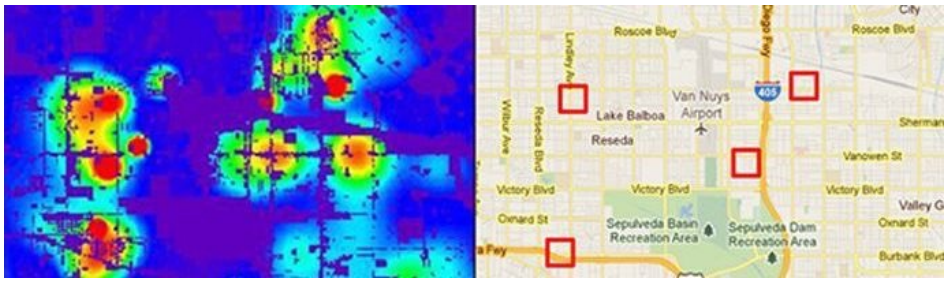
  - Sanity check[1]: is an explanation invariant to irrelevant input perturbations?
    - Later used by many LLM explanation evaluation



[1] The (Un)reliability of saliency methods. 2017

Looking forward

- Methods for Deep Neural Networks and reinforcement learning.There are many interesting methods to discussed.



- Relationship between all these methods [1].
- Interactions with other responsibility areas (privacy, fairness, robustness)
- Interpreting Transformer and large foundation models.
- Surveys:
  - Techniques for Interpretable Machine Learning. CACM. 2020
  - A Multidisciplinary Survey and Framework for Design and Evaluation of Explainable AI Systems. ACM TIST 2020
  - Methods for interpreting and understanding deep neural networks. 2017
  - Towards A Rigorous Science of Interpretable Machine Learning. 2017

[1] Lundberg, S. M. and Lee, S.-I. A unified approach to interpreting model predictions. 2017.