Logic-Based Explainability in AI

Glimpse into ANITI's DeepLever Chair

Joao Marques-Silva

IRIT/CNRS & ANITI DeepLever Chair, Toulouse, France

March 2022

Recent & ongoing ML successes

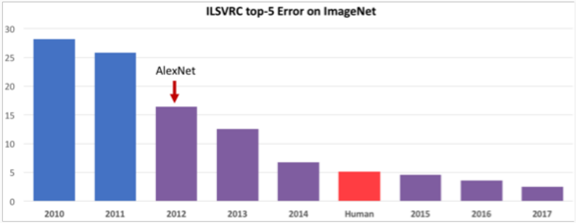
https://en.wikipedia.org/wiki/Waymo

DeepMind

AlphaGo

AlphaGo Zero & **Alpha Zero**

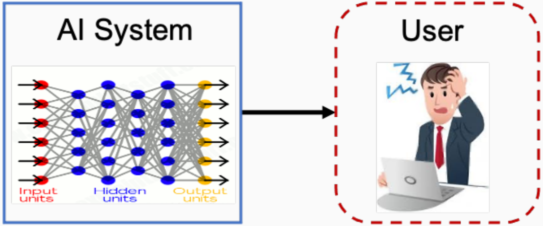
Image & Speech Recognition



http://gradientscience.org/intro_adversarial/

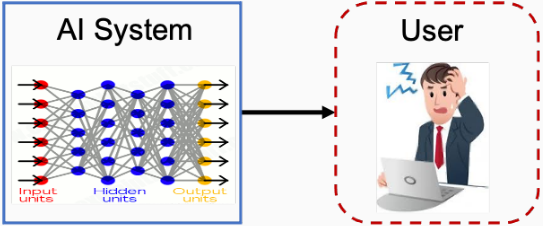
https://fr.wikipedia.org/wiki/Pepper_(robot)

eXplainable AI (XAI)



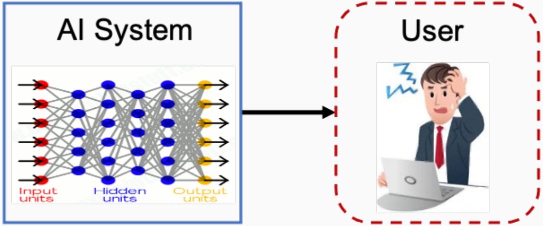
- ML models are most often **opaque**
- Goal of XAI: **to help humans understand ML models**
- Many questions to address:

eXplainable AI (XAI)



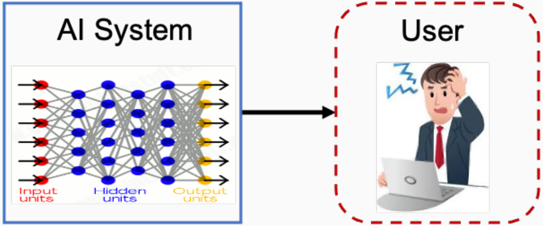
- ML models are most often **opaque**
- Goal of XAI: **to help humans understand ML models**
- Many questions to address:
  - Properties of explanations
    - How to be human understandable?
    - How to answer **Why?** questions? I.e. Why the prediction?
    - How to answer **Why Not?** questions? I.e. Why not some other prediction?
    - Which guarantees of rigor?

eXplainable AI (XAI)



- ML models are most often **opaque**
- Goal of XAI: **to help humans understand ML models**
- Many questions to address:
  - Properties of explanations
    - How to be human understandable?
    - How to answer **Why?** questions? I.e. Why the prediction?
    - How to answer **Why Not?** questions? I.e. Why not some other prediction?
    - Which guarantees of rigor?
  - Other queries: enumeration, membership, preferences, etc.

eXplainable AI (XAI)



- ML models are most often **opaque**
- Goal of XAI: **to help humans understand ML models**
- Many questions to address:
  - Properties of explanations
    - How to be human understandable?
    - How to answer **Why?** questions? I.e. Why the prediction?
    - How to answer **Why Not?** questions? I.e. Why not some other prediction?
    - Which guarantees of rigor?
  - Other queries: enumeration, membership, preferences, etc.
  - Links with robustness, fairness, learning

Importance of XAI

REGULATION (EU) 2016/679 OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL

of 27 April 2016

on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation)

(Text with EEA relevance)

European Union regulations on algorithmic decision-making and a "right to explanation"

Bryce Goodman,[1*] Seth Flaxman,[2]

Proposal for a

REGULATION OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL

LAYING DOWN HARMONISED RULES ON ARTIFICIAL INTELLIGENCE (ARTIFICIAL INTELLIGENCE ACT) AND AMENDING CERTAIN UNION LEGISLATIVE ACTS

■ *We summarize the potential impact that the European Union's new General Data Protection Regulation will have on the routine use of machine-learning algorithms. Slated to take effect as law across the European Union in 2018, it will place restrictions on automated individual decision making (that is, algorithms that make decisions based on user-level predictors) that "significantly affect" users. When put into practice, the law may also effectively create a right to explanation, whereby a user can ask for an explanation of an algorithmic decision that significantly affects them. We argue that while this law may pose large challenges for industry, it highlights opportunities for computer scientists to take the lead in designing algorithms and evaluation frameworks that avoid discrimination and enable explanation.*

Explainable Artificial Intelligence (XAI)

XAI

EXPLAINABLE ARTIFICIAL INTELLIGENCE

FY17 FY18 FY19 FY20 FY21

David Gunning
DARPA/I2O
Program Update November 2017

DARPA

European Commission > Strategy > Digital Single Market > Reports and studies >

Digital Single Market

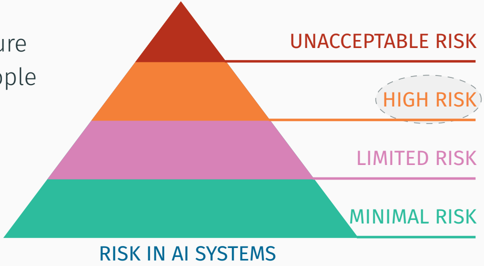
REPORT / STUDY | 8 April 2019

Ethics guidelines for trustworthy AI

XAI for high-risk & safety-critical applications

- **High-risk** (EU regulations):
  - Law enforcement
  - Management and operation of critical infrastructure
  - Biometric identification and categorization of people
  - ...
- **Safety-critical**:
  - Self-driving cars
  - Autonomous vehicles
  - Autonomous aereal devices
  - ...

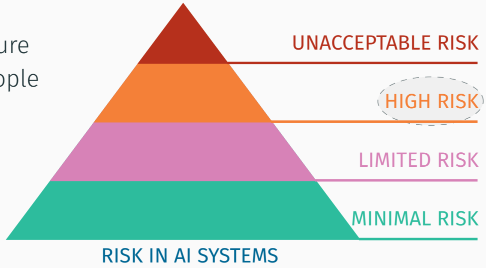
[EU21]



XAI for high-risk & safety-critical applications

- **High-risk** (EU regulations):
  - Law enforcement
  - Management and operation of critical infrastructure
  - Biometric identification and categorization of people
  - ...
- **Safety-critical**:
  - Self-driving cars
  - Autonomous vehicles
  - Autonomous aereal devices
  - ...

[EU21]




PERSPECTIVE
https://doi.org/10.1038/s42256-019-0048-x
nature machine intelligence

Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead

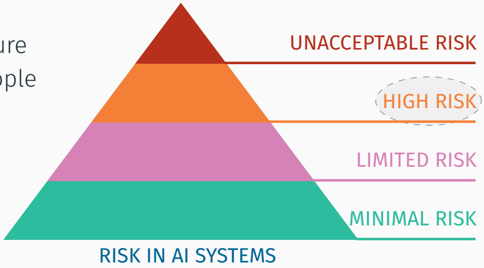
Cynthia Rudin

May 2019

XAI for high-risk & safety-critical applications

- High-risk (EU regulations):
  - Law enforcement
  - Management and operation of critical infrastructure
  - Biometric identification and categorization of people
  - ...
- Safety-critical:
  - Self-driving cars
  - Autonomous vehicles
  - Autonomous aereal devices
  - ...
- Correctness of explanations is paramount!
  - To build trust
  - To help debug AI systems
  - To prevent accidents
  - ...

[EU21]




PERSPECTIVE
https://doi.org/10.1038/s42256-019-0048-x
nature machine intelligence

Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead

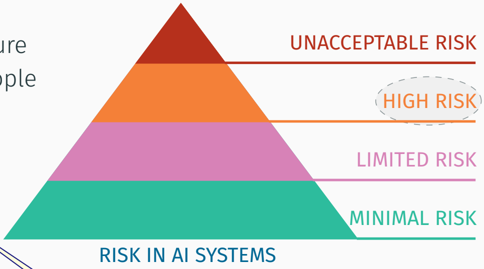
Cynthia Rudin

May 2019

XAI for high-risk & safety-critical applications

- **High-risk** (EU regulations):
  - Law enforcement
  - Management and operation of critical infrastructure
  - Biometric identification and categorization of people
  - ...
- **Safety-critical**:
  - Self-driving cars
  - Autonomous vehicles
  - Autonomous aereal devices
  - ...
- **Correctness of explanations is paramount!**
  - To build trust
  - To help debug AI systems
  - To prevent accidents
  - ...

[EU21]



Main motivation for our work!

Outline

Definitions – classification problems

- Set of features $\mathcal{F} = \{1, 2, \ldots, m\}$, each feature i taking values from domain D_i
  - Features can be categorical, discrete or real-valued
  - Feature space: $\mathbb{F} = \Pi_{i=1}^{m} D_i$
- Set of classes $\mathcal{K} = \{c_1, \ldots, c_K\}$

Definitions – classification problems

- Set of features $\mathcal{F} = \{1, 2, \ldots, m\}$, each feature i taking values from domain D_i
  - Features can be categorical, discrete or real-valued
  - Feature space: $\mathbb{F} = \Pi_{i=1}^{m} D_i$
- Set of classes $\mathcal{K} = \{c_1, \ldots, c_K\}$
- ML model $\mathbb{M}$ computes a (non-constant) classification function $\kappa : \mathbb{F} \to \mathcal{K}$

Definitions – classification problems

- Set of features $\mathcal{F} = \{1, 2, \ldots, m\}$, each feature i taking values from domain D_i
  - Features can be categorical, discrete or real-valued
  - Feature space: $\mathbb{F} = \Pi_{i=1}^{m} D_i$
- Set of classes $\mathcal{K} = \{c_1, \ldots, c_K\}$
- ML model $\mathbb{M}$ computes a (non-constant) classification function $\kappa : \mathbb{F} \rightarrow \mathcal{K}$
- Instance $(\mathbf{v}, c)$ for point $\mathbf{v} = (v_1, \ldots, v_m) \in \mathbb{F}$, with prediction $c = \kappa(\mathbf{v})$, $c \in \mathcal{K}$
  - **Goal:** to compute explanations for $(\mathbf{v}, c)$

Outline

Non-formal XAI approaches – among best known

- Model-agnostic (MA) explainers:
  - LIME & SHAP [RSG16, LL17]
    - Goal: learn a simple interpretable ML model, e.g. linear classifier, decision tree, etc.
    - Approach: train classifier, sample-based vs. game theory
  - Anchor: [RSG18]
    - Goal: learn features deemed more relevant for prediction
    - Anchor is sample-based
  - No formal guarantees of rigor in computed explanations
- Intrinsic interpretability (II) [Rud19, Mol20]
  - (Interpretable) model is the explanation
  - E.g., DTs, DLs, DSs, etc.

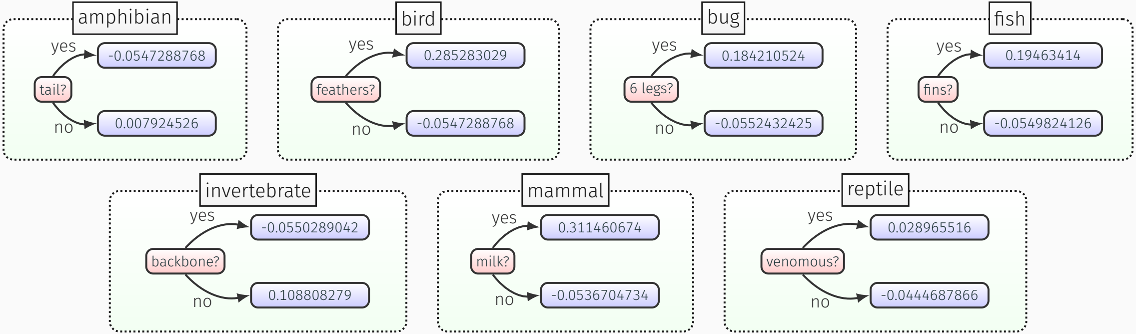
Non-formal XAI approaches – among best known

- Model-agnostic (MA) explainers:
  - LIME & SHAP [RSG16, LL17]
    - Goal: learn a simple interpretable ML model, e.g. linear classifier, decision tree, etc.
    - Approach: train classifier, sample-based vs. game theory
  - Anchor: [RSG18]
    - Goal: learn features deemed more relevant for prediction
    - Anchor is sample-based
  - No formal guarantees of rigor in computed explanations
- Intrinsic interpretability (II) [Rud19, Mol20]
  - (Interpretable) model is the explanation
  - E.g., DTs?, DLs?, DSs?, etc.?

Can MA explainers be trusted ?
Is II indeed *interpretable* ?

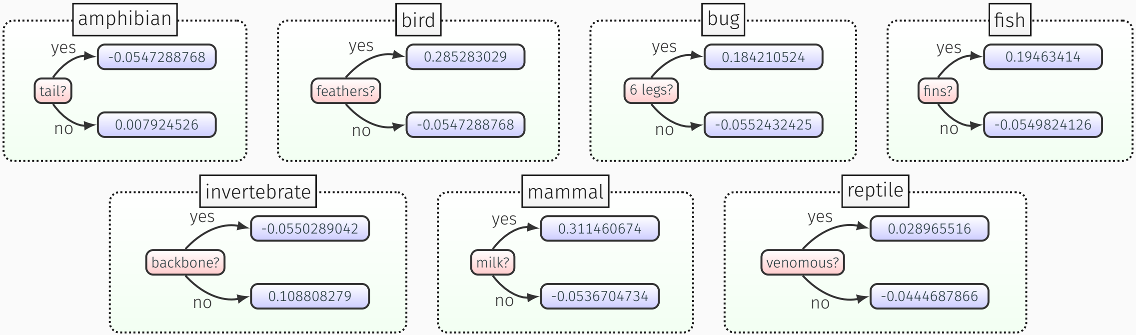
An example – BT for zoo dataset

[INM19c, Ign20]



An example – BT for zoo dataset

[INM19c, Ign20]



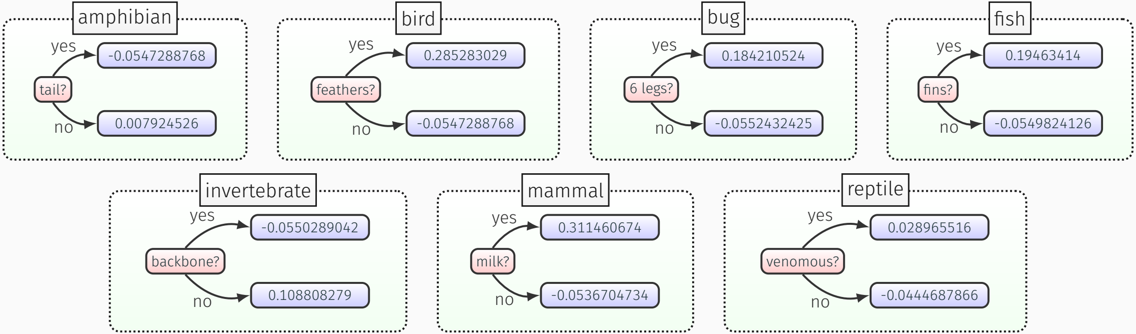
- Example instance:

IF (animal_name = pitviper) ∧ ¬hair ∧ ¬feathers ∧ eggs ∧ ¬milk ∧ ¬airborne ∧ ¬aquatic ∧ predator ∧ ¬toothed ∧ backbone ∧ breathes ∧ venomous ∧ ¬fins ∧ (legs = 0) ∧ tail ∧ ¬domestic ∧ ¬catsize

THEN (class = reptile)

An example – BT for zoo dataset & Anchor

[INM19c, Ign20]



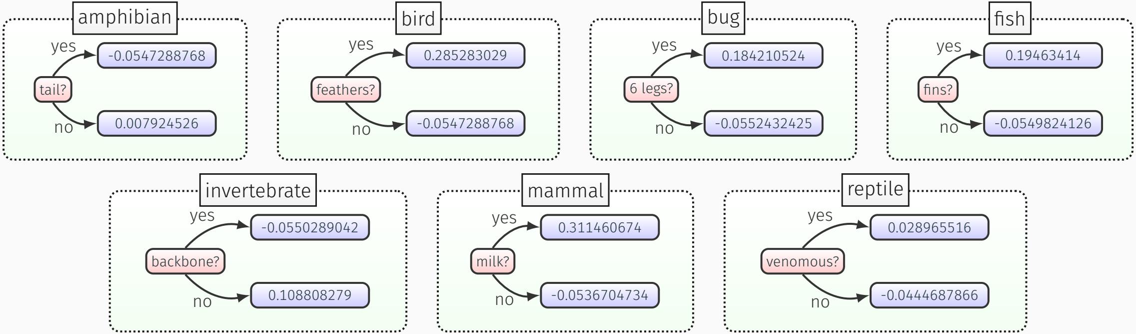
- Example instance (& Anchor picks): [RSG18]

IF (animal_name = pitviper) ∧ ¬***hair*** ∧ ¬feathers ∧ eggs ∧ ¬***milk*** ∧ ¬airborne ∧ ¬aquatic ∧ predator ∧ ¬***toothed*** ∧ backbone ∧ breathes ∧ venomous ∧ ¬***fins*** ∧ (legs = 0) ∧ tail ∧ ¬domestic ∧ ¬catsize

THEN (class = reptile)

An example – BT for zoo dataset & Anchor

[INM19c, Ign20]

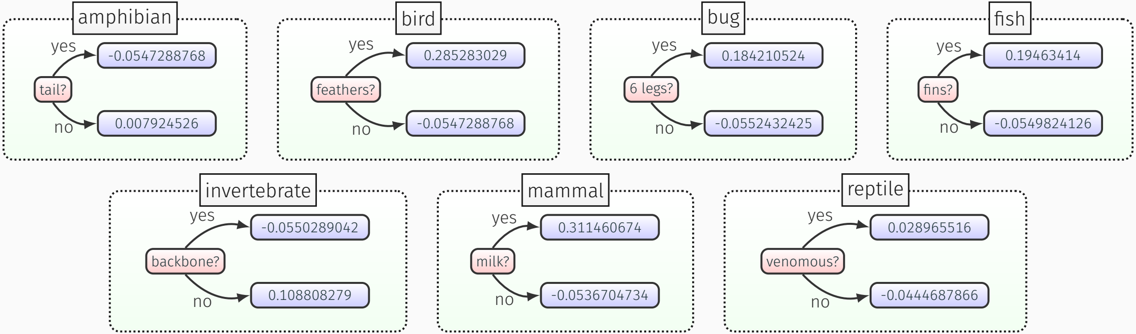


- Explanation obtained with Anchor: [RSG18]

IF	$\neg \textit{hair} \wedge \neg \textit{milk} \wedge \neg \textit{toothed} \wedge \neg \textit{fins}$
THEN	$(\text{class} = \text{reptile})$

An example – BT for zoo dataset & Anchor

[INM19c, Ign20]



- **But**, explanation **incorrectly "explains"** another instance (from training data!)

IF (animal_name = toad) ∧ ¬***hair*** ∧ ¬feathers ∧ eggs ∧ ¬***milk*** ∧ ¬airborne ∧ ¬aquatic ∧ ¬predator ∧ ¬***toothed*** ∧ backbone ∧ breathes ∧ ¬venomous ∧ ¬***fins*** ∧ (legs = 4) ∧ ¬tail ∧ ¬domestic ∧ ¬catsize

THEN (class = amphibian)

Model-agnostic explainers cannot be trusted

[INM19c]

Incorrect explanations:

Example of classifier for deciding bank loans

Model-agnostic explainers cannot be trusted

[INM19c]

Incorrect explanations:

Example of classifier for deciding bank loans

Two samples: Bessie $:= (v_1, \mathsf{Y})$ and Clive $:= (v_2, \mathsf{N})$

Model-agnostic explainers cannot be trusted

[INM19c]

Incorrect explanations:

Example of classifier for deciding bank loans

Two samples: Bessie $:= (v_1, \mathsf{Y})$ and Clive $:= (v_2, \mathsf{N})$

Explanation X: age $= 45$, salary $=$ 50K

Model-agnostic explainers cannot be trusted

[INM19c]

Incorrect explanations:

Example of classifier for deciding bank loans

Two samples: Bessie $:= (v_1, \text{Y})$ and Clive $:= (v_2, \text{N})$

Explanation X: age $= 45$, salary $=$ 50K

And,

X is consistent with Bessie $:= (\mathbf{v}_1, \text{Y})$

X is consistent with Clive $:= (\mathbf{v}_2, \text{N})$

Incorrect explanations:

Example of classifier for deciding bank loans

Two samples: Bessie $\coloneqq (v_1, \text{Y})$ and Clive $\coloneqq (v_2, \text{N})$

Explanation X: age $= 45$, salary $=$ 50K

And,

X is consistent with Bessie $\coloneqq (\mathbf{v}_1, \text{Y})$

X is consistent with Clive $\coloneqq (\mathbf{v}_2, \text{N})$

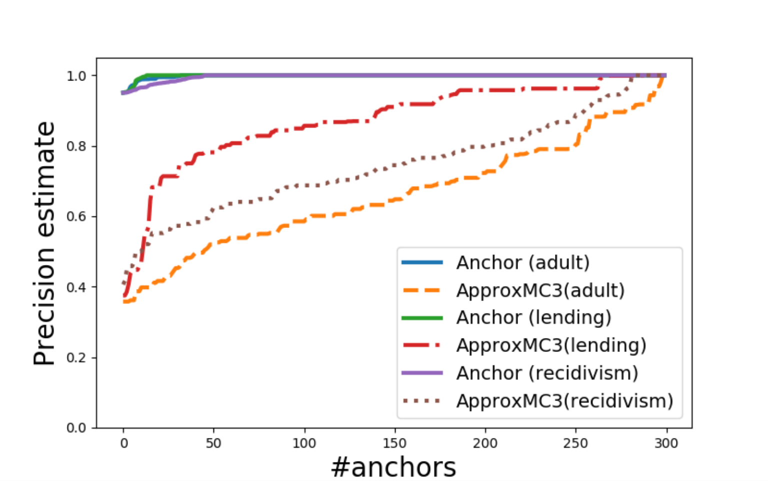
$\therefore$ different outcomes **&** same explanation **!?**

Incorrect explanations are ubiquitous...

[INM19c, Ign20]

Dataset	(# unique)	Explanations								
		incorrect			redundant			correct		
		LIME	Anchor	SHAP	LIME	Anchor	SHAP	LIME	Anchor	SHAP
adult	(5579)	61.3%	**80.5%**	**70.7%**	7.9%	1.6%	10.2%	30.8%	17.9%	19.1 %
lending	(4414)	24.0%	3.0%	17.0%	0.4%	0.0%	2.5%	**75.6%**	**97.0%**	**80.5%**
rcdv	(3696)	**94.1%**	**99.4%**	**85.9%**	4.6%	0.4%	7.9%	1.3%	0.2%	6.2%
compas	(778)	**71.9%**	**84.4%**	60.4%	20.6%	1.7%	27.8%	7.5%	13.9%	11.8%
german	(1000)	**85.3%**	**99.7%**	63.0%	14.6%	0.2%	37.0%	0.1 %	0.1 %	0.0%

Incorrect explanations are ubiquitous & likely... [NSM+19]



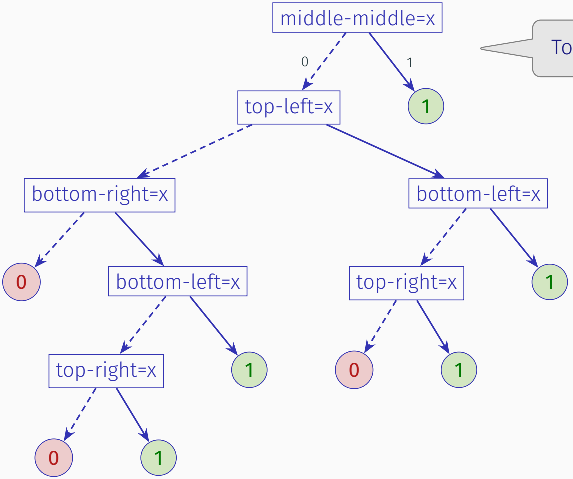
Decision trees are interpretable...

[Bre01, Rud19, Mol20]

"On interpretability, trees rate an A+." [Bre01]

General agreement on interpretability of decision trees [Rud19, Mol20, ANS20, Int21]

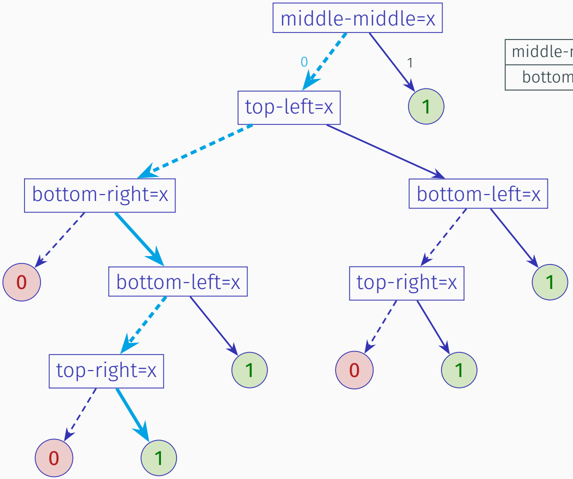
Decision tree explanations can be redundant



Tool: OSDT; train accuracy: 82.881%

Source: Xiyang Hu, Cynthia Rudin, Margo I. Seltzer: **Optimal Sparse Decision Trees**. **NeurIPS 2019**: 7265-7273

Decision tree explanations can be redundant



middle-middle = x iff MM = 1	top-left = x iff TL = 1	
bottom-right = x iff BR = 1	bottom-left = x iff BL = 1	top-right = x iff TR = 1

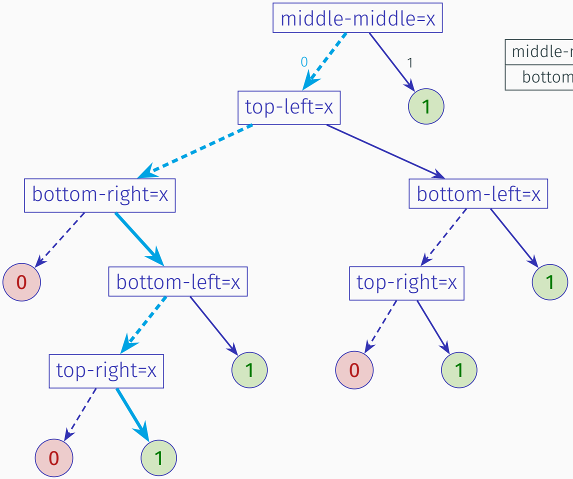
Q: What is explanation for

$(\mathrm{MM} = 0) \wedge (\mathrm{TL} = 0) \wedge (\mathrm{BR} = 1) \wedge$

$(\mathrm{BL} = 0) \wedge (\mathrm{TR} = 1)$?

Source: Xiyang Hu, Cynthia Rudin, Margo I. Seltzer: **Optimal Sparse Decision Trees**. **NeurIPS 2019**: 7265-7273

Decision tree explanations can be redundant



middle-middle = x iff MM = 1	top-left = x iff TL = 1	
bottom-right = x iff BR = 1	bottom-left = x iff BL = 1	top-right = x iff TR = 1

Q: What is explanation for

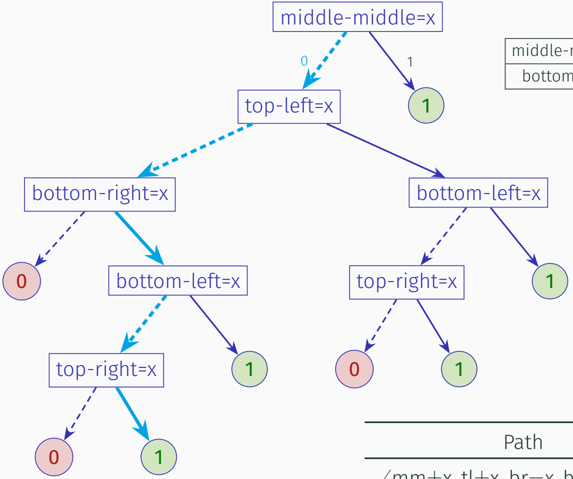
$(\mathrm{MM}=0) \wedge (\mathrm{TL}=0) \wedge (\mathrm{BR}=1) \wedge$

$(\mathrm{BL}=0) \wedge (\mathrm{TR}=1)$?

IF $(\neg\mathrm{MM} \wedge \neg\mathrm{TL} \wedge \mathrm{BR} \wedge \neg\mathrm{BL} \wedge \mathrm{TR})$ THEN $(\kappa(\cdot) = \mathbf{1})$?

Source: Xiyang Hu, Cynthia Rudin, Margo I. Seltzer: **Optimal Sparse Decision Trees**. **NeurIPS 2019**: 7265-7273

Decision tree explanations can be redundant



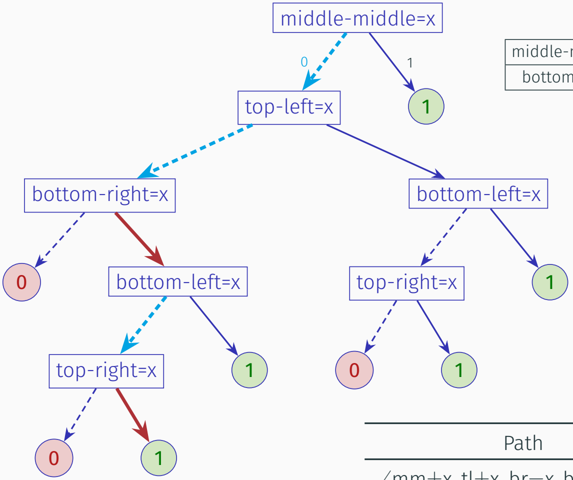
middle-middle = x iff MM = 1	top-left = x iff TL = 1	
bottom-right = x iff BR = 1	bottom-left = x iff BL = 1	top-right = x iff TR = 1

Q: What is explanation for
$(\mathrm{MM} = 0) \land (\mathrm{TL} = 0) \land (\mathrm{BR} = 1) \land$
$(\mathrm{BL} = 0) \land (\mathrm{TR} = 1)$?
IF $(\neg\mathrm{MM} \land \neg\mathrm{TL} \land \mathrm{BR} \land \neg\mathrm{BL} \land \mathrm{TR})$ THEN $(\kappa(\cdot) = \mathbf{1})$?

Path	Reduced XP	Dropped
$\langle$mm$\neq$x, tl$\neq$x, br=x, bl$\neq$x, tr=x$\rangle$		

Source: Xiyang Hu, Cynthia Rudin, Margo I. Seltzer: **Optimal Sparse Decision Trees**. **NeurIPS 2019**: 7265-7273

Decision tree explanations can be arbitrarily redundant



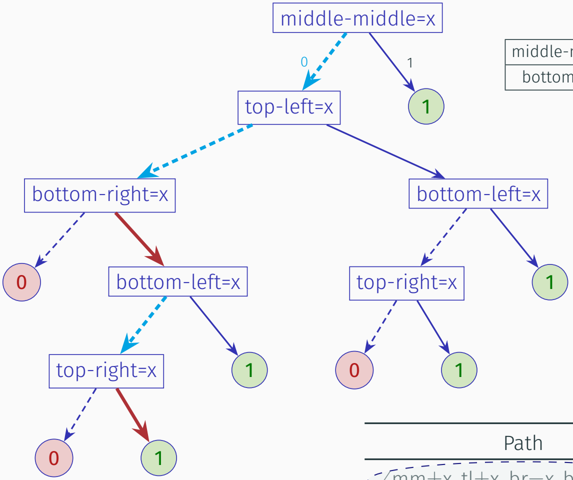
middle-middle $=$ x iff MM $= 1$	top-left $=$ x iff TL $= 1$	
bottom-right $=$ x iff BR $= 1$	bottom-left $=$ x iff BL $= 1$	top-right $=$ x iff TR $= 1$

BR $= 1$, TR $= 1$			
MM	BL	TL	$\kappa(\cdot)$
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

Path	Reduced XP	Dropped
$\langle$mm$\neq$x, tl$\neq$x, br$=$x, bl$\neq$x, tr$=$x$\rangle$	$\langle$br$=$x, tr$=$x$\rangle$	$\{$mm$\neq$x, tl$\neq$x, bl$\neq$x$\}$

Source: Xiyang Hu, Cynthia Rudin, Margo I. Seltzer: **Optimal Sparse Decision Trees**. **NeurIPS 2019**: 7265-7273

Decision tree explanations can be arbitrarily redundant



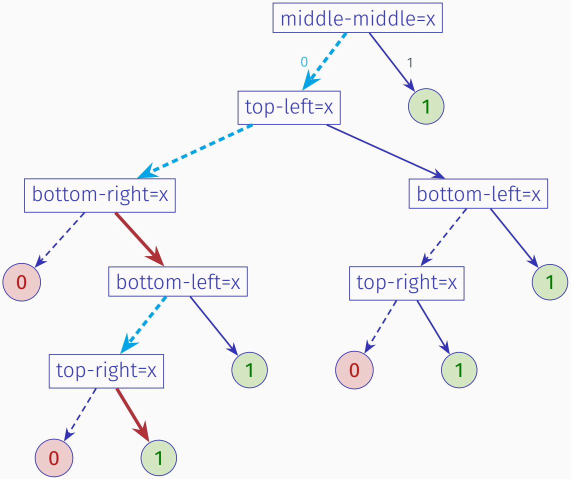
middle-middle = x iff MM = 1	top-left = x iff TL = 1	
bottom-right = x iff BR = 1	bottom-left = x iff BL = 1	top-right = x iff TR = 1

BR = 1, TR = 1			
MM	BL	TL	$\kappa(\cdot)$
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

Path	Reduced XP	Dropped
$\langle$mm$\neq$x, tl$\neq$x, br=x, bl$\neq$x, tr=x$\rangle$	$\langle$br=x, tr=x$\rangle$	{mm$\neq$x, tl$\neq$x, bl$\neq$x}

Source: Xiyang Hu, Cynthia Rudin, Margo I. Seltzer: **Optimal Sparse Decision Trees**. **NeurIPS 2019**: 7265-7273

Decision tree explanations can be arbitrarily redundant



# tree paths	8
# red. paths	5
% red. paths	62.5%

Source: Xiyang Hu, Cynthia Rudin, Margo I. Seltzer: **Optimal Sparse Decision Trees**. **NeurIPS 2019**: 7265-7273

Decision trees are interpretable...

[Bre01, Rud19, Mol20]

- Classifier, with $x_1, \ldots, x_m \in \{0, 1\}$:

$$\kappa(x_1, x_2, \ldots, x_{m-1}, x_m) = \bigvee_{i=1}^{m} x_i$$

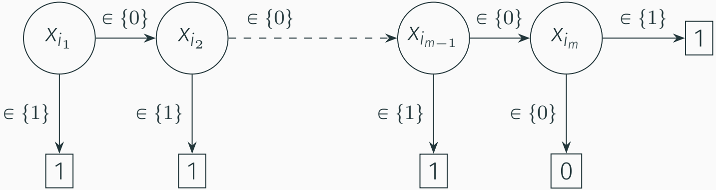
Decision trees are interpretable...

[Bre01, Rud19, Mol20]

- Classifier, with $x_1, \ldots, x_m \in \{0, 1\}$:

$$\kappa(x_1, x_2, \ldots, x_{m-1}, x_m) = \bigvee_{i=1}^{m} x_i$$

- Decision tree (DT), by picking variables in order $\langle i_1, i_2, \ldots, i_m \rangle$, permutation of $\langle 1, 2, \ldots, m \rangle$:



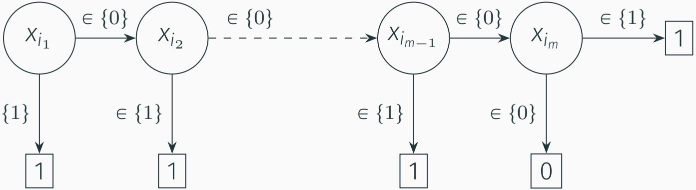
Decision trees are interpretable...

[Bre01, Rud19, Mol20]

- Classifier, with $x_1, \ldots, x_m \in \{0,1\}$:

$$\kappa(x_1, x_2, \ldots, x_{m-1}, x_m) = \bigvee\nolimits_{i=1}^{m} x_i$$

- Decision tree (DT), by picking variables in order $\langle i_1, i_2, \ldots, i_m \rangle$, permutation of $\langle 1, 2, \ldots, m \rangle$:



- Point: $(x_{i_1}, x_{i_2}, \ldots, x_{i_{m-1}}, x_{i_m}) = (0, 0, \ldots, 0, 1)$, and prediction 1

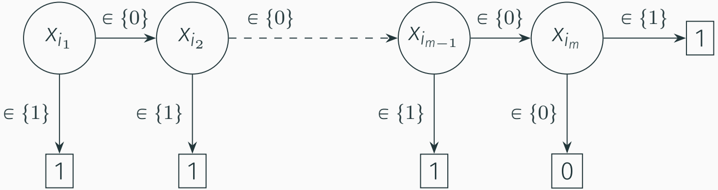
Decision trees are interpretable...

[Bre01, Rud19, Mol20]

- Classifier, with $x_1, \ldots, x_m \in \{0, 1\}$:

$$\kappa(x_1, x_2, \ldots, x_{m-1}, x_m) = \bigvee_{i=1}^{m} x_i$$

- Decision tree (DT), by picking variables in order $\langle i_1, i_2, \ldots, i_m \rangle$, permutation of $\langle 1, 2, \ldots, m \rangle$:



- Point: $(x_{i_1}, x_{i_2}, \ldots, x_{i_{m-1}}, x_{i_m}) = (0, 0, \ldots, 0, 1)$, and prediction 1
- Explanation using path in DT: $\{i_1, i_2, \ldots, i_m\}$, i.e.

$$(x_{i_1} = 0) \land (x_{i_2} = 0) \land \ldots \land (x_{i_{m-1}} = 0) \land (x_{i_m} = 1) \rightarrow \kappa(x_1, \ldots, x_m)$$

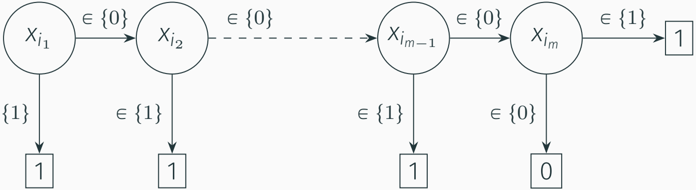
Decision trees are interpretable...

[Bre01, Rud19, Mol20]

- Classifier, with $x_1, \ldots, x_m \in \{0,1\}$:

$$\kappa(x_1, x_2, \ldots, x_{m-1}, x_m) = \bigvee\nolimits_{i=1}^{m} x_i$$

- Decision tree (DT), by picking variables in order $\langle i_1, i_2, \ldots, i_m \rangle$, permutation of $\langle 1, 2, \ldots, m \rangle$:



- Point: $(x_{i_1}, x_{i_2}, \ldots, x_{i_{m-1}}, x_{i_m}) = (0, 0, \ldots, 0, 1)$, and prediction 1
- Explanation using path in DT: $\{i_1, i_2, \ldots, i_m\}$, i.e.

$$(x_{i_1} = 0) \land (x_{i_2} = 0) \land \ldots \land (x_{i_{m-1}} = 0) \land (x_{i_m} = 1) \rightarrow \kappa(x_1, \ldots, x_m)$$

- But $\{i_m\}$ suffices for prediction, i.e. $\forall(\mathbf{x} \in \{0,1\}^m).(x_{i_m}) \rightarrow \kappa(\mathbf{x})$

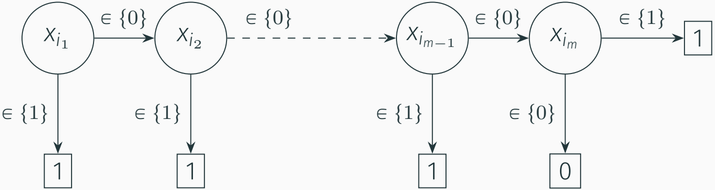
Minimal decision trees are not interpretable!

[IIM20, HIIM21]

- Classifier, with $x_1, \ldots, x_m \in \{0,1\}$:

$$\kappa(x_1, x_2, \ldots, x_{m-1}, x_m) = \bigvee_{i=1}^{m} x_i$$

- Decision tree (DT), by picking variables in order $\langle i_1, i_2, \ldots, i_m \rangle$, permutation of $\langle 1, 2, \ldots, m \rangle$:



- Point: $(x_{i_1}, x_{i_2}, \ldots, x_{i_{m-1}}, x_{i_m}) = (0, 0, \ldots, 0, 1)$, and prediction 1
- Explanation using path in DT: $\{i_1, i_2, \ldots, i_m\}$, i.e.

$$(x_{i_1} = 0) \wedge (x_{i_2} = 0) \wedge \ldots \wedge (x_{i_{m-1}} = 0) \wedge (x_{i_m} = 1) \rightarrow \kappa(x_1, \ldots, x_m)$$

- **But $\{i_m\}$ suffices for prediction, i.e. $\forall(\mathbf{x} \in \{0,1\}^m).(x_{i_m}) \rightarrow \kappa(\mathbf{x})$**

True explanations arbitrarily smaller than paths in DT; **$\therefore$ DTs are not interpretable***

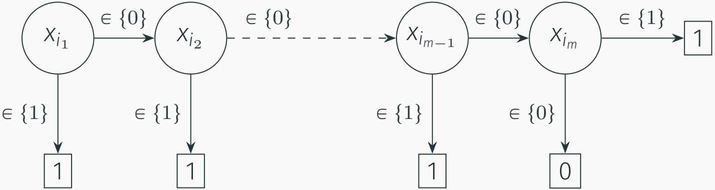
Minimal decision trees are not interpretable!

[IIM20, HIIM21]

- Classifier, with $x_1, \ldots, x_m \in \{0,1\}$:

$$\kappa(x_1, x_2, \ldots, x_{m-1}, x_m) = \bigvee_{i=1}^{m} x_i$$

- Decision tree (DT), by picking variables in order $\langle i_1, i_2, \ldots, i_m \rangle$, permutation of $\langle 1, 2, \ldots, m \rangle$:



- Point: $(x_{i_1}, x_{i_2}, \ldots, x_{i_{m-1}}, x_{i_m}) = (0, 0, \ldots, 0, 1)$, and prediction 1
- Explanation using path in DT: $\{i_1, i_2, \ldots, i_m\}$, i.e.

$$(x_{i_1} = 0) \wedge (x_{i_2} = 0) \wedge \ldots \wedge (x_{i_{m-1}} = 0) \wedge (x_{i_m} = 1) \rightarrow \kappa(x_1, \ldots, x_m)$$

- **But $\{i_m\}$ suffices for prediction, i.e. $\forall(\mathbf{x} \in \{0,1\}^m).(x_{i_m}) \rightarrow \kappa(\mathbf{x})$**

True explanations arbitrarily smaller than paths in DT; **$\therefore$ DTs are not interpretable***

*Pick any definition of interpretability that correlates with succinctness ...

[Lip18]

Outline

Formal explanations

- Instance $(\mathbf{v}, c)$, i.e. $c = \kappa(\mathbf{v})$

Formal explanations

- Instance $(\mathbf{v}, c)$, i.e. $c = \kappa(\mathbf{v})$
- Abductive explanation (AXp, PI-explanation) – answer to **Why?** question: [SCD18, INM19a]
  - **Subset-minimal** set of features $\mathcal{X} \subseteq \mathcal{F}$ sufficient for **ensuring** prediction

$$\forall(\mathbf{x} \in \mathbb{F}). \bigwedge_{j \in \mathcal{X}} (x_j = v_j) \rightarrow (\kappa(\mathbf{x}) = c)$$

Formal explanations

- Instance $(\mathbf{v}, c)$, i.e. $c = \kappa(\mathbf{v})$
- **Abductive explanation** (AXp, PI-explanation) – answer to **Why?** question: [SCD18, INM19a]
  - **Subset-minimal** set of features $\mathcal{X} \subseteq \mathcal{F}$ sufficient for **ensuring** prediction

$$\forall(\mathbf{x} \in \mathbb{F}). \bigwedge_{j \in \mathcal{X}} (x_j = v_j) \rightarrow (\kappa(\mathbf{x}) = c)$$

- **Contrastive explanation** (CXp) – answer to **Why Not?** question: [Mil19, INAM20]
  - **Subset-minimal** set of features $\mathcal{Y} \subseteq \mathcal{F}$ sufficient for **changing** prediction

$$\exists(\mathbf{x} \in \mathbb{F}). \bigwedge_{j \notin \mathcal{Y}} (x_j = v_j) \wedge (\kappa(\mathbf{x}) \neq c)$$

Formal explanations

- Instance $(\mathbf{v}, c)$, i.e. $c = \kappa(\mathbf{v})$
- Abductive explanation (AXp, PI-explanation) – answer to **Why?** question: [SCD18, INM19a]
  - Subset-minimal set of features $\mathcal{X} \subseteq \mathcal{F}$ sufficient for **ensuring** prediction

$$\forall(\mathbf{x} \in \mathbb{F}). \bigwedge_{j\in\mathcal{X}}(x_j = v_j) \rightarrow (\kappa(\mathbf{x}) = c)$$

- Contrastive explanation (CXp) – answer to **Why Not?** question: [Mil19, INAM20]
  - Subset-minimal set of features $\mathcal{Y} \subseteq \mathcal{F}$ sufficient for **changing** prediction

$$\exists(\mathbf{x} \in \mathbb{F}). \bigwedge_{j\notin\mathcal{Y}}(x_j = v_j) \land (\kappa(\mathbf{x}) \neq c)$$

- Subset-minimal, given predicate $\mathbb{P}$: $\mathbb{P}(\mathcal{Z}) \land \forall(\mathcal{Z}' \subsetneq \mathcal{Z}).\neg\mathbb{P}(\mathcal{Z}')$

Formal explanations

- Instance $(\mathbf{v}, c)$, i.e. $c = \kappa(\mathbf{v})$
- Abductive explanation (AXp, PI-explanation) – answer to **Why?** question: [SCD18, INM19a]
  - Subset-minimal set of features $\mathcal{X} \subseteq \mathcal{F}$ sufficient for **ensuring** prediction

$$\mathsf{WeakAXp}(\mathcal{X}) \quad := \quad \forall(\mathbf{x} \in \mathbb{F}). \bigwedge_{j \in \mathcal{X}} (x_j = v_j) \rightarrow (\kappa(\mathbf{x}) = c)$$

- Contrastive explanation (CXp) – answer to **Why Not?** question: [Mil19, INAM20]
  - Subset-minimal set of features $\mathcal{Y} \subseteq \mathcal{F}$ sufficient for **changing** prediction

$$\mathsf{WeakCXp}(\mathcal{Y}) \quad := \quad \exists(\mathbf{x} \in \mathbb{F}). \bigwedge_{j \notin \mathcal{Y}} (x_j = v_j) \land (\kappa(\mathbf{x}) \neq c)$$

- Subset-minimal, given predicate $\mathbb{P}$: $\mathbb{P}(\mathcal{Z}) \land \forall(\mathcal{Z}' \subsetneq \mathcal{Z}).\neg\mathbb{P}(\mathcal{Z}')$

$$\mathsf{AXp}(\mathcal{X}) := \mathsf{WeakAXp}(\mathcal{X}) \land \forall(\mathcal{X}' \subsetneq \mathcal{X}).\neg\mathsf{WeakAXp}(\mathcal{X}')$$
$$\mathsf{CXp}(\mathcal{Y}) := \mathsf{WeakCXp}(\mathcal{Y}) \land \forall(\mathcal{Y}' \subsetneq \mathcal{Y}).\neg\mathsf{WeakCXp}(\mathcal{Y}')$$

Formal explanations

- Instance $(\mathbf{v}, c)$, i.e. $c = \kappa(\mathbf{v})$
- Abductive explanation (AXp, PI-explanation) – answer to **Why?** question: [SCD18, INM19a]
  - Subset-minimal set of features $\mathcal{X} \subseteq \mathcal{F}$ sufficient for **ensuring** prediction

$$\mathsf{WeakAXp}(\mathcal{X}) \quad := \quad \forall(\mathbf{x} \in \mathbb{F}). \bigwedge\nolimits_{j \in \mathcal{X}} (x_j = v_j) \rightarrow (\kappa(\mathbf{x}) = c)$$

- Contrastive explanation (CXp) – answer to **Why Not?** question: [Mil19, INAM20]
  - Subset-minimal set of features $\mathcal{Y} \subseteq \mathcal{F}$ sufficient for **changing** prediction

$$\mathsf{WeakCXp}(\mathcal{Y}) \quad := \quad \exists(\mathbf{x} \in \mathbb{F}). \bigwedge\nolimits_{j \notin \mathcal{Y}} (x_j = v_j) \land (\kappa(\mathbf{x}) \neq c)$$

- Subset-minimal, given predicate $\mathbb{P}$: $\mathbb{P}(\mathcal{Z}) \land \forall(\mathcal{Z}' \subsetneq \mathcal{Z}).\neg\mathbb{P}(\mathcal{Z}')$

$$\mathsf{AXp}(\mathcal{X}) := \mathsf{WeakAXp}(\mathcal{X}) \land \forall(\mathcal{X}' \subsetneq \mathcal{X}).\neg\mathsf{WeakAXp}(\mathcal{X}')$$
$$\mathsf{CXp}(\mathcal{Y}) := \mathsf{WeakCXp}(\mathcal{Y}) \land \forall(\mathcal{Y}' \subsetneq \mathcal{Y}).\neg\mathsf{WeakCXp}(\mathcal{Y}')$$

- **AXp's are minimal hitting sets (MHSes) of CXp's and vice-versa** (& more) [INAM20, INM19b]

General approach

- AXp/CXp conditions are monotone wrt set inclusion

General approach

- AXp/CXp conditions are monotone wrt set inclusion $\Longrightarrow$ suffices to check: $\mathbb{P}(\mathcal{Z}) \wedge \forall(t \in \mathcal{Z}).\neg\mathbb{P}(\mathcal{Z}\setminus\{t\})$
  - E.g. AXp is defined by: $\text{AXp}(\mathcal{X}) := \text{WeakAXp}(\mathcal{X}) \wedge \forall(t \in \mathcal{X}).\neg\text{WeakAXp}(\mathcal{X}\setminus\{t\})$

General approach

- AXp/CXp conditions are monotone wrt set inclusion $\Longrightarrow$ suffices to check: $\mathbb{P}(\mathcal{Z}) \wedge \forall(t \in \mathcal{Z}).\neg\mathbb{P}(\mathcal{Z}\setminus\{t\})$
  - E.g. AXp is defined by: $\text{AXp}(\mathcal{X}) := \text{WeakAXp}(\mathcal{X}) \wedge \forall(t \in \mathcal{X}).\neg\text{WeakAXp}(\mathcal{X}\setminus\{t\})$
- Approach:
  - Encode classifier into suitable logic representation $\mathcal{T}$
  - For AXp: start from $\mathcal{S} = \mathcal{F}$ and drop features from $\mathcal{S}$ while AXp condition holds
  - For CXp: start from $\mathcal{S} = \mathcal{F}$ and drop features from $\mathcal{S}$ while CXp condition holds (*)

General approach

- AXp/CXp conditions are monotone wrt set inclusion $\Longrightarrow$ suffices to check: $\mathbb{P}(\mathcal{Z}) \land \forall(t \in \mathcal{Z}).\neg\mathbb{P}(\mathcal{Z}\setminus\{t\})$
  - E.g. AXp is defined by: $\text{AXp}(\mathcal{X}) := \text{WeakAXp}(\mathcal{X}) \land \forall(t \in \mathcal{X}).\neg\text{WeakAXp}(\mathcal{X}\setminus\{t\})$
- Approach:
  - Encode classifier into suitable logic representation $\mathcal{T}$
  - For **AXp**: start from $\mathcal{S} = \mathcal{F}$ and drop features from $\mathcal{S}$ while AXp condition holds
  - For **CXp**: start from $\mathcal{S} = \mathcal{F}$ and drop features from $\mathcal{S}$ while CXp condition holds (*)
  - **Q:** which reasoner for $\mathcal{T}$ to use?
- Monotone predicates for AXp & CXp:

$$\mathbb{P}_{\text{axp}}(\mathcal{S}) \triangleq \neg\,\mathbf{CO}\left(\left[\!\left[\left(\bigwedge\nolimits_{i\in\mathcal{S}}(x_i = v_i)\right) \land (\kappa(\mathbf{x}) \neq c)\right]\!\right]\right) \qquad \mathbb{P}_{\text{cxp}}(\mathcal{S}) \triangleq \mathbf{CO}\left(\left[\!\left[\left(\bigwedge\nolimits_{i\in\mathcal{F}\setminus\mathcal{S}}(x_i = v_i)\right) \land (\kappa(\mathbf{x}) \neq c)\right]\!\right]\right)$$

General approach

- AXp/CXp conditions are monotone wrt set inclusion $\Longrightarrow$ suffices to check: $\mathbb{P}(\mathcal{Z}) \land \forall(t \in \mathcal{Z}).\neg\mathbb{P}(\mathcal{Z}\setminus\{t\})$
  - E.g. AXp is defined by: $\mathsf{AXp}(\mathcal{X}) := \mathsf{WeakAXp}(\mathcal{X}) \land \forall(t \in \mathcal{X}).\neg\mathsf{WeakAXp}(\mathcal{X}\setminus\{t\})$
- Approach:
  - Encode classifier into suitable logic representation $\mathcal{T}$
  - For **AXp**: start from $\mathcal{S} = \mathcal{F}$ and drop features from $\mathcal{S}$ while AXp condition holds
  - For **CXp**: start from $\mathcal{S} = \mathcal{F}$ and drop features from $\mathcal{S}$ while CXp condition holds (*)
  - **Q:** which reasoner for $\mathcal{T}$ to use?
- Monotone predicates for AXp & CXp:

$$\mathbb{P}_{\mathrm{axp}}(\mathcal{S}) \triangleq \neg\,\mathsf{CO}\left(\left\llbracket\left(\bigwedge\nolimits_{i\in\mathcal{S}}(x_i = v_i)\right) \land (\kappa(\mathbf{x}) \neq c)\right\rrbracket\right) \qquad \mathbb{P}_{\mathrm{cxp}}(\mathcal{S}) \triangleq \mathsf{CO}\left(\left\llbracket\left(\bigwedge\nolimits_{i\in\mathcal{F}\setminus\mathcal{S}}(x_i = v_i)\right) \land (\kappa(\mathbf{x}) \neq c)\right\rrbracket\right)$$

Input: Predicate $\mathbb{P}$, parameterized by $\mathcal{T}$, $\mathcal{F}$, κ, $\mathbf{v}$
Output: One XP $\mathcal{S}$

```
procedure oneXP(ℙ)
    S ← F                              ▷ Initialization: ℙ(S) holds
    for i ∈ F do                       ▷ Loop invariant: ℙ(S) holds
        if ℙ(S\{i}) then
            S ← S\{i}                  ▷ Update S only if ℙ(S\{i}) holds
    return S                           ▷ Returned set S: ℙ(S) holds
```

General approach

- AXp/CXp conditions are monotone wrt set inclusion $\Longrightarrow$ suffices to check: $\mathbb{P}(\mathcal{Z}) \land \forall(t \in \mathcal{Z}).\neg\mathbb{P}(\mathcal{Z}\setminus\{t\})$
  - E.g. AXp is defined by: $\mathsf{AXp}(\mathcal{X}) := \mathsf{WeakAXp}(\mathcal{X}) \land \forall(t \in \mathcal{X}).\neg\mathsf{WeakAXp}(\mathcal{X}\setminus\{t\})$
- Approach:
  - Encode classifier into suitable logic representation $\mathcal{T}$
  - For **AXp**: start from $\mathcal{S} = \mathcal{F}$ and drop features from $\mathcal{S}$ while AXp condition holds
  - For **CXp**: start from $\mathcal{S} = \mathcal{F}$ and drop features from $\mathcal{S}$ while CXp condition holds (*)
  - **Q:** which reasoner for $\mathcal{T}$ to use, and how to encode classifiers?
- Monotone predicates for AXp & CXp:

$$\mathbb{P}_{\mathrm{axp}}(\mathcal{S}) \triangleq \neg\,\mathbf{CO}\left(\llbracket(\textstyle\bigwedge_{i\in\mathcal{S}}(x_i = v_i)) \land (\kappa(\mathbf{x}) \neq c)\rrbracket\right) \qquad \mathbb{P}_{\mathrm{cxp}}(\mathcal{S}) \triangleq \mathbf{CO}\left(\left\llbracket\left(\textstyle\bigwedge_{i\in\mathcal{F}\setminus\mathcal{S}}(x_i = v_i)\right) \land (\kappa(\mathbf{x}) \neq c)\right\rrbracket\right)$$

Input: Predicate $\mathbb{P}$, parameterized by $\mathcal{T}$, $\mathcal{F}$, κ, $\mathbf{v}$
Output: One XP $\mathcal{S}$

How to encode
NNs, RFs, BTs, etc. ?

```
procedure oneXP(ℙ)
    S ← F                          ▷ Initialization: ℙ(S) holds
    for i ∈ F do                   ▷ Loop invariant: ℙ(S) holds
        if ℙ(S\{i}) then
            S ← S\{i}              ▷ Update S only if ℙ(S\{i}) holds
    return S                       ▷ Returned set S: ℙ(S) holds
```

A simple example – AXp's

- Classifier:

$$\kappa(x_1, x_2, x_3, x_4) = \bigvee_{i=1}^{4} x_i$$

A simple example – AXp's

- Classifier:

$$\kappa(x_1, x_2, x_3, x_4) = \bigvee_{i=1}^{4} x_i$$

- Point $\mathbf{v} = (0, 0, 0, 1)$ with prediction $\kappa(\mathbf{v}) = 1$. AXp?

A simple example – AXp's

- Classifier:

$$\kappa(x_1, x_2, x_3, x_4) = \bigvee_{i=1}^{4} x_i$$

- Point $\mathbf{v} = (0, 0, 0, 1)$ with prediction $\kappa(\mathbf{v}) = 1$. AXp?
- Define $\mathcal{X} = \{1, 2, 3, 4\} = \mathcal{F}$

A simple example – AXp's

- Classifier:

$$\kappa(x_1, x_2, x_3, x_4) = \bigvee_{i=1}^{4} x_i$$

- Point $\mathbf{v} = (0, 0, 0, 1)$ with prediction $\kappa(\mathbf{v}) = 1$.   AXp?
- Define $\mathcal{X} = \{1, 2, 3, 4\} = \mathcal{F}$
- Can feature 1 be removed, i.e. $\forall(\mathbf{x} \in \{0, 1\}^4).\neg x_2 \land \neg x_3 \land x_4 \rightarrow \kappa(x_1, x_2, x_3, x_4)$?

Recap weak AXp: $\forall(\mathbf{x} \in \mathbb{F}). \bigwedge_{j \in \mathcal{X}}(x_j = v_j) \rightarrow (\kappa(\mathbf{x}) = c)$

A simple example – AXp's

- Classifier:

$$\kappa(x_1, x_2, x_3, x_4) = \bigvee_{i=1}^{4} x_i$$

- Point $\mathbf{v} = (0, 0, 0, 1)$ with prediction $\kappa(\mathbf{v}) = 1$. AXp?
- Define $\mathcal{X} = \{1, 2, 3, 4\} = \mathcal{F}$
- Can feature 1 be removed, i.e. $\forall(\mathbf{x} \in \{0, 1\}^4).\neg x_2 \land \neg x_3 \land x_4 \rightarrow \kappa(x_1, x_2, x_3, x_4)$? Yes

Recap weak AXp: $\forall(\mathbf{x} \in \mathbb{F}). \bigwedge_{j \in \mathcal{X}} (x_j = v_j) \rightarrow (\kappa(\mathbf{x}) = c)$

A simple example – AXp's

- Classifier:

$$\kappa(x_1, x_2, x_3, x_4) = \bigvee_{i=1}^{4} x_i$$

- Point $\mathbf{v} = (0, 0, 0, 1)$ with prediction $\kappa(\mathbf{v}) = 1$. AXp?
- Define $\mathcal{X} = \{1, 2, 3, 4\} = \mathcal{F}$
- Can feature 1 be removed, i.e. $\forall(\mathbf{x} \in \{0, 1\}^4).\neg x_2 \land \neg x_3 \land x_4 \rightarrow \kappa(x_1, x_2, x_3, x_4)$? Yes
- Can feature 2 be removed, i.e. $\forall(\mathbf{x} \in \{0, 1\}^4).\neg x_3 \land x_4 \rightarrow \kappa(x_1, x_2, x_3, x_4)$?

Recap weak AXp: $\forall(\mathbf{x} \in \mathbb{F}). \bigwedge_{j \in \mathcal{X}} (x_j = v_j) \rightarrow (\kappa(\mathbf{x}) = c)$

A simple example – AXp's

- Classifier:

$$\kappa(x_1, x_2, x_3, x_4) = \bigvee_{i=1}^{4} x_i$$

- Point $\mathbf{v} = (0, 0, 0, 1)$ with prediction $\kappa(\mathbf{v}) = 1$. AXp?
- Define $\mathcal{X} = \{1, 2, 3, 4\} = \mathcal{F}$
- Can feature 1 be removed, i.e. $\forall(\mathbf{x} \in \{0, 1\}^4).\neg x_2 \land \neg x_3 \land x_4 \rightarrow \kappa(x_1, x_2, x_3, x_4)$? Yes
- Can feature 2 be removed, i.e. $\forall(\mathbf{x} \in \{0, 1\}^4).\neg x_3 \land x_4 \rightarrow \kappa(x_1, x_2, x_3, x_4)$? Yes

Recap weak AXp: $\forall(\mathbf{x} \in \mathbb{F}). \bigwedge_{j\in\mathcal{X}}(x_j = v_j) \rightarrow (\kappa(\mathbf{x}) = c)$

A simple example – AXp's

- Classifier:

$$\kappa(x_1, x_2, x_3, x_4) = \bigvee_{i=1}^{4} x_i$$

- Point $\mathbf{v} = (0, 0, 0, 1)$ with prediction $\kappa(\mathbf{v}) = 1$. AXp?
- Define $\mathcal{X} = \{1, 2, 3, 4\} = \mathcal{F}$
- Can feature 1 be removed, i.e. $\forall(\mathbf{x} \in \{0, 1\}^4).\neg x_2 \wedge \neg x_3 \wedge x_4 \rightarrow \kappa(x_1, x_2, x_3, x_4)$? Yes
- Can feature 2 be removed, i.e. $\forall(\mathbf{x} \in \{0, 1\}^4).\neg x_3 \wedge x_4 \rightarrow \kappa(x_1, x_2, x_3, x_4)$? Yes
- Can feature 3 be removed, i.e. $\forall(\mathbf{x} \in \{0, 1\}^4).x_4 \rightarrow \kappa(x_1, x_2, x_3, x_4)$?

Recap weak AXp: $\forall(\mathbf{x} \in \mathbb{F}). \bigwedge_{j \in \mathcal{X}}(x_j = v_j) \rightarrow (\kappa(\mathbf{x}) = c)$

A simple example – AXp's

- Classifier:

$$\kappa(x_1, x_2, x_3, x_4) = \bigvee_{i=1}^{4} x_i$$

- Point $\mathbf{v} = (0, 0, 0, 1)$ with prediction $\kappa(\mathbf{v}) = 1$. AXp?
- Define $\mathcal{X} = \{1, 2, 3, 4\} = \mathcal{F}$
- Can feature 1 be removed, i.e. $\forall(\mathbf{x} \in \{0, 1\}^4).\neg x_2 \land \neg x_3 \land x_4 \rightarrow \kappa(x_1, x_2, x_3, x_4)$? Yes
- Can feature 2 be removed, i.e. $\forall(\mathbf{x} \in \{0, 1\}^4).\neg x_3 \land x_4 \rightarrow \kappa(x_1, x_2, x_3, x_4)$? Yes
- Can feature 3 be removed, i.e. $\forall(\mathbf{x} \in \{0, 1\}^4).x_4 \rightarrow \kappa(x_1, x_2, x_3, x_4)$? Yes

Recap weak AXp: $\forall(\mathbf{x} \in \mathbb{F}). \bigwedge_{j\in\mathcal{X}}(x_j = v_j) \rightarrow (\kappa(\mathbf{x}) = c)$

A simple example – AXp's

- Classifier:

$$\kappa(x_1, x_2, x_3, x_4) = \bigvee_{i=1}^{4} x_i$$

- Point $\mathbf{v} = (0, 0, 0, 1)$ with prediction $\kappa(\mathbf{v}) = 1$. AXp?
- Define $\mathcal{X} = \{1, 2, 3, 4\} = \mathcal{F}$
- Can feature 1 be removed, i.e. $\forall(\mathbf{x} \in \{0, 1\}^4).\neg x_2 \land \neg x_3 \land x_4 \rightarrow \kappa(x_1, x_2, x_3, x_4)$? Yes
- Can feature 2 be removed, i.e. $\forall(\mathbf{x} \in \{0, 1\}^4).\neg x_3 \land x_4 \rightarrow \kappa(x_1, x_2, x_3, x_4)$? Yes
- Can feature 3 be removed, i.e. $\forall(\mathbf{x} \in \{0, 1\}^4).x_4 \rightarrow \kappa(x_1, x_2, x_3, x_4)$? Yes
- Can feature 4 be removed, i.e. $\forall(\mathbf{x} \in \{0, 1\}^4).\top \rightarrow \kappa(x_1, x_2, x_3, x_4)$?

Recap weak AXp: $\forall(\mathbf{x} \in \mathbb{F}). \bigwedge_{j \in \mathcal{X}} (x_j = v_j) \rightarrow (\kappa(\mathbf{x}) = c)$

A simple example – AXp's

- Classifier:

$$\kappa(x_1, x_2, x_3, x_4) = \bigvee_{i=1}^{4} x_i$$

- Point $\mathbf{v} = (0, 0, 0, 1)$ with prediction $\kappa(\mathbf{v}) = 1$. AXp?
- Define $\mathcal{X} = \{1, 2, 3, 4\} = \mathcal{F}$
- Can feature 1 be removed, i.e. $\forall(\mathbf{x} \in \{0, 1\}^4).\neg x_2 \land \neg x_3 \land x_4 \rightarrow \kappa(x_1, x_2, x_3, x_4)$? Yes
- Can feature 2 be removed, i.e. $\forall(\mathbf{x} \in \{0, 1\}^4).\neg x_3 \land x_4 \rightarrow \kappa(x_1, x_2, x_3, x_4)$? Yes
- Can feature 3 be removed, i.e. $\forall(\mathbf{x} \in \{0, 1\}^4).x_4 \rightarrow \kappa(x_1, x_2, x_3, x_4)$? Yes
- Can feature 4 be removed, i.e. $\forall(\mathbf{x} \in \{0, 1\}^4).\top \rightarrow \kappa(x_1, x_2, x_3, x_4)$? No

Recap weak AXp: $\forall(\mathbf{x} \in \mathbb{F}). \bigwedge_{j \in \mathcal{X}}(x_j = v_j) \rightarrow (\kappa(\mathbf{x}) = c)$

A simple example – AXp's

- Classifier:

$$\kappa(x_1, x_2, x_3, x_4) = \bigvee_{i=1}^{4} x_i$$

- Point $\mathbf{v} = (0, 0, 0, 1)$ with prediction $\kappa(\mathbf{v}) = 1$. AXp?
- Define $\mathcal{X} = \{1, 2, 3, 4\} = \mathcal{F}$
- Can feature 1 be removed, i.e. $\forall(\mathbf{x} \in \{0,1\}^4).\neg x_2 \land \neg x_3 \land x_4 \rightarrow \kappa(x_1, x_2, x_3, x_4)$? Yes
- Can feature 2 be removed, i.e. $\forall(\mathbf{x} \in \{0,1\}^4).\neg x_3 \land x_4 \rightarrow \kappa(x_1, x_2, x_3, x_4)$? Yes
- Can feature 3 be removed, i.e. $\forall(\mathbf{x} \in \{0,1\}^4).x_4 \rightarrow \kappa(x_1, x_2, x_3, x_4)$? Yes
- Can feature 4 be removed, i.e. $\forall(\mathbf{x} \in \{0,1\}^4).\top \rightarrow \kappa(x_1, x_2, x_3, x_4)$? No
- AXp $\mathcal{X} = \{4\}$

Recap weak AXp: $\forall(\mathbf{x} \in \mathbb{F}). \bigwedge_{j \in \mathcal{X}} (x_j = v_j) \rightarrow (\kappa(\mathbf{x}) = c)$

A simple example – AXp's

- Classifier:

$$\kappa(x_1, x_2, x_3, x_4) = \bigvee\nolimits_{i=1}^{4} x_i$$

- Point $\mathbf{v} = (0, 0, 0, 1)$ with prediction $\kappa(\mathbf{v}) = 1$. AXp?
- Define $\mathcal{X} = \{1, 2, 3, 4\} = \mathcal{F}$
- Can feature 1 be removed, i.e. $\forall(\mathbf{x} \in \{0,1\}^4).\neg x_2 \land \neg x_3 \land x_4 \rightarrow \kappa(x_1, x_2, x_3, x_4)$? Yes
- Can feature 2 be removed, i.e. $\forall(\mathbf{x} \in \{0,1\}^4).\neg x_3 \land x_4 \rightarrow \kappa(x_1, x_2, x_3, x_4)$? Yes
- Can feature 3 be removed, i.e. $\forall(\mathbf{x} \in \{0,1\}^4).x_4 \rightarrow \kappa(x_1, x_2, x_3, x_4)$? Yes
- Can feature 4 be removed, i.e. $\forall(\mathbf{x} \in \{0,1\}^4).\top \rightarrow \kappa(x_1, x_2, x_3, x_4)$? No
- AXp $\mathcal{X} = \{4\}$

- **Validity/consistency checked with SAT/SMT/MILP/CP reasoners**

Recap weak AXp: $\forall(\mathbf{x} \in \mathbb{F}). \bigwedge_{j \in \mathcal{X}}(x_j = v_j) \rightarrow (\kappa(\mathbf{x}) = c)$

A simple example – AXp's

- Classifier:

$$\kappa(x_1, x_2, x_3, x_4) = \bigvee_{i=1}^{4} x_i$$

- Point $\mathbf{v} = (0, 0, 0, 1)$ with prediction $\kappa(\mathbf{v}) = 1$. AXp?
- Define $\mathcal{X} = \{1, 2, 3, 4\} = \mathcal{F}$
- Can feature 1 be removed, i.e. $\forall(\mathbf{x} \in \{0,1\}^4).\neg x_2 \land \neg x_3 \land x_4 \rightarrow \kappa(x_1, x_2, x_3, x_4)$? Yes
- Can feature 2 be removed, i.e. $\forall(\mathbf{x} \in \{0,1\}^4).\neg x_3 \land x_4 \rightarrow \kappa(x_1, x_2, x_3, x_4)$? Yes
- Can feature 3 be removed, i.e. $\forall(\mathbf{x} \in \{0,1\}^4).x_4 \rightarrow \kappa(x_1, x_2, x_3, x_4)$? Yes
- Can feature 4 be removed, i.e. $\forall(\mathbf{x} \in \{0,1\}^4).\top \rightarrow \kappa(x_1, x_2, x_3, x_4)$? No
- AXp $\mathcal{X} = \{4\}$

- Validity/consistency checked with SAT/SMT/MILP/CP reasoners
  - **Obs:** for some classes of classifiers, poly-time algorithms exist

Recap weak AXp: $\forall(\mathbf{x} \in \mathbb{F}). \bigwedge_{j \in \mathcal{X}} (x_j = v_j) \rightarrow (\kappa(\mathbf{x}) = c)$

A simple example – CXp's

- Classifier:

$$\kappa(x_1, x_2, x_3, x_4) = \bigvee_{i=1}^{4} x_i$$

- Point $\mathbf{v} = (0, 0, 0, 1)$ with prediction $\kappa(\mathbf{v}) = 1$
- Define $\mathcal{Y} = \{1, 2, 3, 4\} = \mathcal{F}$

A simple example – CXp's

- Classifier:

$$\kappa(x_1, x_2, x_3, x_4) = \bigvee_{i=1}^{4} x_i$$

- Point $\mathbf{v} = (0, 0, 0, 1)$ with prediction $\kappa(\mathbf{v}) = 1$
- Define $\mathcal{Y} = \{1, 2, 3, 4\} = \mathcal{F}$
- Can feature 1 be removed, i.e. $\exists(\mathbf{x} \in \{0, 1\}^4).\neg x_1 \land \neg\kappa(x_1, x_2, x_3, x_4)$?

Recap weak CXp: $\exists(\mathbf{x} \in \mathbb{F}). \bigwedge_{j \notin \mathcal{Y}}(x_j = v_j) \land (\kappa(\mathbf{x}) \neq c)$

A simple example – CXp's

- Classifier:

$$\kappa(x_1, x_2, x_3, x_4) = \bigvee_{i=1}^{4} x_i$$

- Point $\mathbf{v} = (0, 0, 0, 1)$ with prediction $\kappa(\mathbf{v}) = 1$
- Define $\mathcal{Y} = \{1, 2, 3, 4\} = \mathcal{F}$
- Can feature 1 be removed, i.e. $\exists(\mathbf{x} \in \{0, 1\}^4).\neg x_1 \wedge \neg\kappa(x_1, x_2, x_3, x_4)$? Yes

Recap weak CXp: $\exists(\mathbf{x} \in \mathbb{F}). \bigwedge_{j \notin \mathcal{Y}}(x_j = v_j) \wedge (\kappa(\mathbf{x}) \neq c)$

A simple example – CXp's

- Classifier:

$$\kappa(x_1, x_2, x_3, x_4) = \bigvee_{i=1}^{4} x_i$$

- Point $\mathbf{v} = (0, 0, 0, 1)$ with prediction $\kappa(\mathbf{v}) = 1$
- Define $\mathcal{Y} = \{1, 2, 3, 4\} = \mathcal{F}$
- Can feature 1 be removed, i.e. $\exists(\mathbf{x} \in \{0,1\}^4).\neg x_1 \wedge \neg\kappa(x_1, x_2, x_3, x_4)$? Yes
- Can feature 2 be removed, i.e. $\exists(\mathbf{x} \in \{0,1\}^4).\neg x_1 \wedge \neg x_2 \wedge \neg\kappa(x_1, x_2, x_3, x_4)$?

Recap weak CXp: $\exists(\mathbf{x} \in \mathbb{F}). \bigwedge_{j \notin \mathcal{Y}}(x_j = v_j) \wedge (\kappa(\mathbf{x}) \neq c)$

A simple example – CXp's

- Classifier:

$$\kappa(x_1, x_2, x_3, x_4) = \bigvee_{i=1}^{4} x_i$$

- Point $\mathbf{v} = (0, 0, 0, 1)$ with prediction $\kappa(\mathbf{v}) = 1$
- Define $\mathcal{Y} = \{1, 2, 3, 4\} = \mathcal{F}$
- Can feature 1 be removed, i.e. $\exists(\mathbf{x} \in \{0,1\}^4).\neg x_1 \land \neg\kappa(x_1, x_2, x_3, x_4)$? Yes
- Can feature 2 be removed, i.e. $\exists(\mathbf{x} \in \{0,1\}^4).\neg x_1 \land \neg x_2 \land \neg\kappa(x_1, x_2, x_3, x_4)$? Yes

Recap weak CXp: $\exists(\mathbf{x} \in \mathbb{F}). \bigwedge_{j \notin \mathcal{Y}} (x_j = v_j) \land (\kappa(\mathbf{x}) \neq c)$

A simple example – CXp's

- Classifier:

$$\kappa(x_1, x_2, x_3, x_4) = \bigvee_{i=1}^{4} x_i$$

- Point $\mathbf{v} = (0, 0, 0, 1)$ with prediction $\kappa(\mathbf{v}) = 1$
- Define $\mathcal{Y} = \{1, 2, 3, 4\} = \mathcal{F}$
- Can feature 1 be removed, i.e. $\exists(\mathbf{x} \in \{0, 1\}^4).\neg x_1 \land \neg\kappa(x_1, x_2, x_3, x_4)$? Yes
- Can feature 2 be removed, i.e. $\exists(\mathbf{x} \in \{0, 1\}^4).\neg x_1 \land \neg x_2 \land \neg\kappa(x_1, x_2, x_3, x_4)$? Yes
- Can feature 3 be removed, i.e. $\exists(\mathbf{x} \in \{0, 1\}^4).\neg x_1 \land \neg x_2 \land \neg x_3 \land \neg\kappa(x_1, x_2, x_3, x_4)$?

Recap weak CXp: $\exists(\mathbf{x} \in \mathbb{F}).\bigwedge_{j \notin \mathcal{Y}}(x_j = v_j) \land (\kappa(\mathbf{x}) \neq c)$

A simple example – CXp's

- Classifier:

$$\kappa(x_1, x_2, x_3, x_4) = \bigvee_{i=1}^{4} x_i$$

- Point $\mathbf{v} = (0, 0, 0, 1)$ with prediction $\kappa(\mathbf{v}) = 1$
- Define $\mathcal{Y} = \{1, 2, 3, 4\} = \mathcal{F}$
- Can feature 1 be removed, i.e. $\exists(\mathbf{x} \in \{0, 1\}^4).\neg x_1 \land \neg\kappa(x_1, x_2, x_3, x_4)$? Yes
- Can feature 2 be removed, i.e. $\exists(\mathbf{x} \in \{0, 1\}^4).\neg x_1 \land \neg x_2 \land \neg\kappa(x_1, x_2, x_3, x_4)$? Yes
- Can feature 3 be removed, i.e. $\exists(\mathbf{x} \in \{0, 1\}^4).\neg x_1 \land \neg x_2 \land \neg x_3 \land \neg\kappa(x_1, x_2, x_3, x_4)$? Yes

Recap weak CXp: $\exists(\mathbf{x} \in \mathbb{F}). \bigwedge_{j \notin \mathcal{Y}} (x_j = v_j) \land (\kappa(\mathbf{x}) \neq c)$

A simple example – CXp's

- Classifier:

$$\kappa(x_1, x_2, x_3, x_4) = \bigvee_{i=1}^{4} x_i$$

- Point $\mathbf{v} = (0, 0, 0, 1)$ with prediction $\kappa(\mathbf{v}) = 1$
- Define $\mathcal{Y} = \{1, 2, 3, 4\} = \mathcal{F}$
- Can feature 1 be removed, i.e. $\exists(\mathbf{x} \in \{0, 1\}^4).\neg x_1 \land \neg\kappa(x_1, x_2, x_3, x_4)$? Yes
- Can feature 2 be removed, i.e. $\exists(\mathbf{x} \in \{0, 1\}^4).\neg x_1 \land \neg x_2 \land \neg\kappa(x_1, x_2, x_3, x_4)$? Yes
- Can feature 3 be removed, i.e. $\exists(\mathbf{x} \in \{0, 1\}^4).\neg x_1 \land \neg x_2 \land \neg x_3 \land \neg\kappa(x_1, x_2, x_3, x_4)$? Yes
- Can feature 4 be removed, i.e. $\exists(\mathbf{x} \in \{0, 1\}^4).\neg x_1 \land \neg x_2 \land \neg x_3 \land x_4 \land \neg\kappa(x_1, x_2, x_3, x_4)$?

Recap weak CXp: $\exists(\mathbf{x} \in \mathbb{F}). \bigwedge_{j \notin \mathcal{Y}}(x_j = v_j) \land (\kappa(\mathbf{x}) \neq c)$

A simple example – CXp's

- Classifier:

$$\kappa(x_1, x_2, x_3, x_4) = \bigvee_{i=1}^{4} x_i$$

- Point $\mathbf{v} = (0, 0, 0, 1)$ with prediction $\kappa(\mathbf{v}) = 1$
- Define $\mathcal{Y} = \{1, 2, 3, 4\} = \mathcal{F}$
- Can feature 1 be removed, i.e. $\exists(\mathbf{x} \in \{0, 1\}^4).\neg x_1 \wedge \neg\kappa(x_1, x_2, x_3, x_4)$? Yes
- Can feature 2 be removed, i.e. $\exists(\mathbf{x} \in \{0, 1\}^4).\neg x_1 \wedge \neg x_2 \wedge \neg\kappa(x_1, x_2, x_3, x_4)$? Yes
- Can feature 3 be removed, i.e. $\exists(\mathbf{x} \in \{0, 1\}^4).\neg x_1 \wedge \neg x_2 \wedge \neg x_3 \wedge \neg\kappa(x_1, x_2, x_3, x_4)$? Yes
- Can feature 4 be removed, i.e. $\exists(\mathbf{x} \in \{0, 1\}^4).\neg x_1 \wedge \neg x_2 \wedge \neg x_3 \wedge x_4 \wedge \neg\kappa(x_1, x_2, x_3, x_4)$? No

Recap weak CXp: $\exists(\mathbf{x} \in \mathbb{F}). \bigwedge_{j \notin \mathcal{Y}} (x_j = v_j) \wedge (\kappa(\mathbf{x}) \neq c)$

A simple example – CXp's

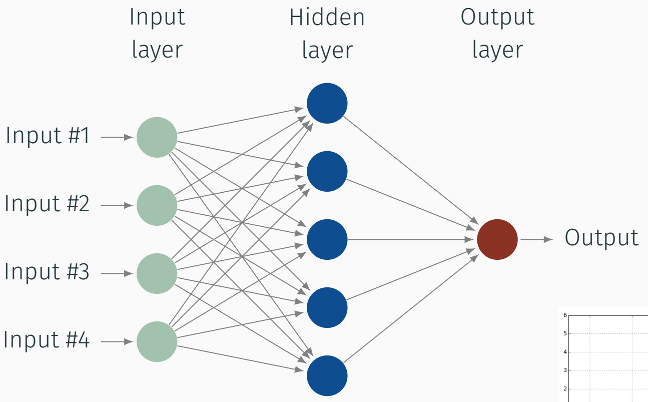
- Classifier:

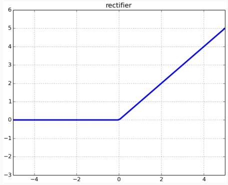
$$\kappa(x_1, x_2, x_3, x_4) = \bigvee_{i=1}^{4} x_i$$

- Point $\mathbf{v} = (0, 0, 0, 1)$ with prediction $\kappa(\mathbf{v}) = 1$
- Define $\mathcal{Y} = \{1, 2, 3, 4\} = \mathcal{F}$
- Can feature 1 be removed, i.e. $\exists(\mathbf{x} \in \{0,1\}^4).\neg x_1 \land \neg\kappa(x_1, x_2, x_3, x_4)$? Yes
- Can feature 2 be removed, i.e. $\exists(\mathbf{x} \in \{0,1\}^4).\neg x_1 \land \neg x_2 \land \neg\kappa(x_1, x_2, x_3, x_4)$? Yes
- Can feature 3 be removed, i.e. $\exists(\mathbf{x} \in \{0,1\}^4).\neg x_1 \land \neg x_2 \land \neg x_3 \land \neg\kappa(x_1, x_2, x_3, x_4)$? Yes
- Can feature 4 be removed, i.e. $\exists(\mathbf{x} \in \{0,1\}^4).\neg x_1 \land \neg x_2 \land \neg x_3 \land x_4 \land \neg\kappa(x_1, x_2, x_3, x_4)$? No
- CXp $\mathcal{Y} = \{4\}$
- **Obs:** AXp is MHS of CXp and vice-versa...

Recap weak CXp: $\exists(\mathbf{x} \in \mathbb{F}). \bigwedge_{j \notin \mathcal{Y}}(x_j = v_j) \land (\kappa(\mathbf{x}) \neq c)$

Encoding NNs





- Each layer (except first) viewed as a **block**, and
  - Compute $\mathbf{x}'$ given input $\mathbf{x}$, weights matrix $\mathbf{A}$, and bias vector $\mathbf{b}$
  - Compute output $\mathbf{y}$ given $\mathbf{x}'$ and activation function
- Each unit uses a **ReLU** activation function

[NH10]

Encoding NNs (using MILP)

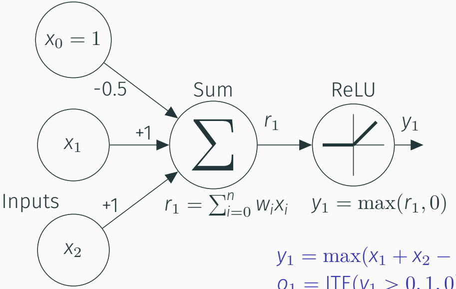
Computation for a NN ReLU **block**, in two steps:

$$\mathbf{x}' = \mathbf{A} \cdot \mathbf{x} + \mathbf{b}$$
$$\mathbf{y} = \max(\mathbf{x}', \mathbf{0})$$

Encoding each **block**: [FJ18]

$$\sum_{j=1}^{n} a_{i,j} x_j + b_i = y_i - s_i$$
$$z_i = 1 \rightarrow y_i \leqslant 0$$
$$z_i = 0 \rightarrow s_i \leqslant 0$$
$$y_i \geqslant 0, s_i \geqslant 0, z_i \in \{0, 1\}$$

Encoding a simple NN in MILP



x_1	x_2	r_1	y_1	o_1
0	0	-0.5	0	0
1	0	0.5	0.5	1
0	1	0.5	0.5	1
1	1	1.5	1.5	1

$$y_1 = \max(x_1 + x_2 - 0.5, 0)$$
$$o_1 = \mathrm{ITE}(y_1 > 0, 1, 0)$$

MILP encoding:

$$x_1 + x_2 - 0.5 = y_1 - s_1$$
$$z_1 = 1 \rightarrow y_1 \leqslant 0$$
$$z_1 = 0 \rightarrow s_1 \leqslant 0$$
$$o_1 = (y_1 > 0)$$
$$x_1, x_2, z_1, o_1 \in \{0, 1\}$$
$$y_1, s_1 \geqslant 0$$

Instance: $(\mathbf{x}, c) = ((1, 0), 1)$

$$1 + 0 - 0.5 = 0.5 - 0$$
$$1 \vee 0.5 \leqslant 0$$
$$0 \vee 0 \leqslant 0$$
$$1 = (0.5 > 0)$$
$$x_1 = 1, x_2 = 0, z_1 = 0, o_1 = 1$$
$$y_1 = 0.5, s_1 = 0$$

Checking: $\mathbf{x} = (0, 0)$

$$0 + 0 - 0.5 = 0 - 0.5$$
$$0 \vee 0 \leqslant 0$$
$$1 \vee 0.5 \leqslant 0$$
$$0 = (0 > 0)$$
$$x_1 = 0, x_2 = 0, z_1 = 1, o_1 = 0$$
$$y_1 = 0, s_1 = 0.5$$

Initial results for NNs (with SMT/MILP)

[INM19a]

Dataset			Minimal explanation			Minimum explanation		
			size	SMT (s)	MILP (s)	size	SMT (s)	MILP (s)
australian	(14)	m	1	0.03	0.05	—	—	—
		a	8.79	1.38	0.33	—	—	—
		M	14	17.00	1.43	—	—	—
backache	(32)	m	13	0.13	0.14	—	—	—
		a	19.28	5.08	0.85	—	—	—
		M	26	22.21	2.75	—	—	—
breast-cancer	(9)	m	3	0.02	0.04	3	0.02	0.03
		a	5.15	0.65	0.20	4.86	2.18	0.41
		M	9	6.11	0.41	9	24.80	1.81
cleve	(13)	m	4	0.05	0.07	4	—	0.07
		a	8.62	3.32	0.32	7.89	—	5.14
		M	13	60.74	0.60	13	—	39.06
hepatitis	(19)	m	6	0.02	0.04	4	0.01	0.04
		a	11.42	0.07	0.06	9.39	4.07	2.89
		M	19	0.26	0.20	19	27.05	22.23
voting	(16)	m	3	0.01	0.02	3	0.01	0.02
		a	4.56	0.04	0.13	3.46	0.3	0.25
		M	11	0.10	0.37	11	1.25	1.77
spect	(22)	m	3	0.02	0.02	3	0.02	0.04
		a	7.31	0.13	0.07	6.44	1.61	0.67
		M	20	0.88	0.29	20	8.97	10.73

Initial results for NNs (with SMT/MILP)

[INM19a]

First rigorous approach
for **explaining** NNs !

			Minimal explanation			Minimum explanation		
			size	SMT (s)	MILP (s)	size	SMT (s)	MILP (s)
australian	(14)	m	1	0.03	0.05	—	—	—
		a	8.79	1.38	0.33	—	—	—
		M	14	17.00	1.43	—	—	—
backache	(32)	m	13	0.13	0.14	—	—	—
		a	19.28	5.08	0.85	—	—	—
		M	26	22.21	2.75	—	—	—
breast-cancer	(9)	m	3	0.02	0.04	3	0.02	0.03
		a	5.15	0.65	0.20	4.86	2.18	0.41
		M	9	6.11	0.41	9	24.80	1.81
cleve	(13)	m	4	0.05	0.07	4	—	0.07
		a	8.62	3.32	0.32	7.89	—	5.14
		M	13	60.74	0.60	13	—	39.06
hepatitis	(19)	m	6	0.02	0.04	4	0.01	0.04
		a	11.42	0.07	0.06	9.39	4.07	2.89
		M	19	0.26	0.20	19	27.05	22.23
voting	(16)	m	3	0.01	0.02	3	0.01	0.02
		a	4.56	0.04	0.13	3.46	0.3	0.25
		M	11	0.10	0.37	11	1.25	1.77
spect	(22)	m	3	0.02	0.02	3	0.02	0.04
		a	7.31	0.13	0.07	6.44	1.61	0.67
		M	20	0.88	0.29	20	8.97	10.73

Initial results for NNs (with SMT/MILP)

[INM19a]

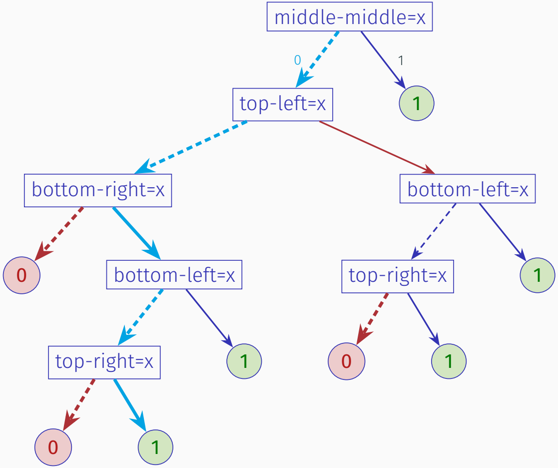
First rigorous approach for **explaining** NNs !

			Minimal explanation			Minimum explanation		
			size	SMT (s)	MILP (s)	size	SMT (s)	MILP (s)
australian	(14)	m	1	0.03	0.05	—	—	—
		a	8.79	1.38	0.33	—	—	—
		M	14	17.00	1.43	—	—	—
backache	(32)	m	13	0.13	0.14	—	—	—
		a	19.28	5.08	0.85	—	—	—
		M	26	22.21	2.75	—	—	—
breast-cancer	(9)	m	3	0.02	0.04	3	0.02	0.03
		a	5.15	0.65	0.20	4.86	2.18	0.41
		M	9	6.11	0.41	9	24.80	1.81
cleve	(13)	m	4	0.05	0.07	4	—	0.07
		a	8.62	3.32	0.32	7.89	—	5.14
		M	13	60.74	0.60	13	—	39.06
hepatitis	(19)	m	6	0.02	0.04	4	0.01	0.04
		a	11.42	0.07	0.06	9.39	4.07	2.89
		M	19	0.26	0.20	19	27.05	22.23
voting	(16)	m	3	0.01	0.02	3	0.01	0.02
		a	4.56	0.04	0.13	3.46	0.3	0.25
		M	11	0.10	0.37	11	1.25	1.77
spect	(22)	m	3	0.02	0.02	3	0.02	0.04
		a	7.31	0.13	0.07	6.44	1.61	0.67
		M	20	0.88	0.29	20	8.97	10.73

Scales to (a few) tens of neurons...

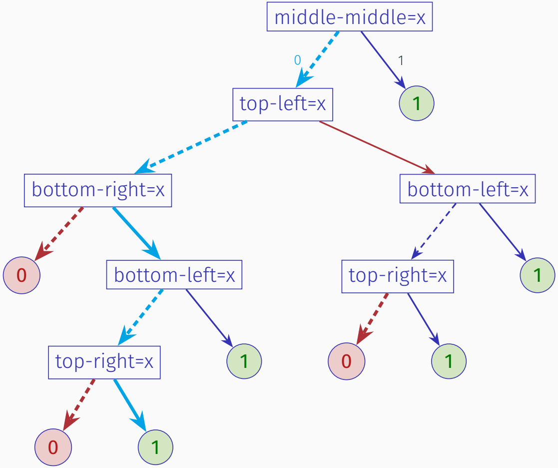
DT explanations

[IIM20]



DT explanations

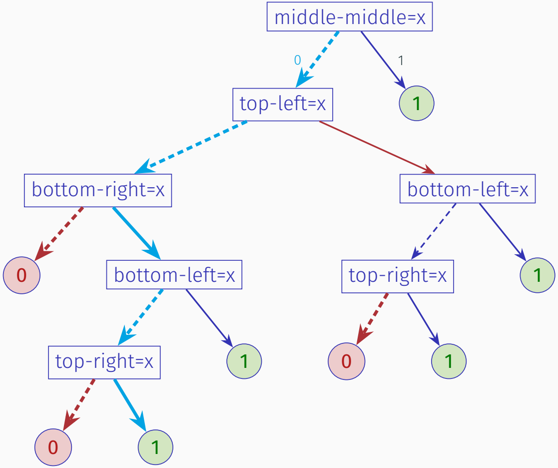
[IIM20]



- Run PI-explanation algorithm based on NP-oracles
    - Worst-case exponential time

DT explanations

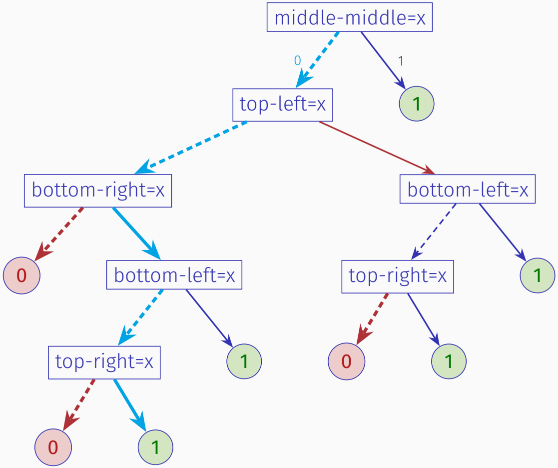
[IIM20]



- Run PI-explanation algorithm based on NP-oracles
  - Worst-case exponential time
- For prediction 1, it suffices to ensure all paths with prediction 0 remain inconsistent

DT explanations in polynomial time

[IIM20]



- Run PI-explanation algorithm based on NP-oracles
  - Worst-case exponential time
- For prediction 1, it suffices to ensure all paths with prediction 0 remain inconsistent
  - I.e. find a subset-minimal hitting set of all 0 paths; these are the features to keep
    - E.g. BR and TR suffice for prediction
  - Well-known to be solvable in polynomial time [EG95]

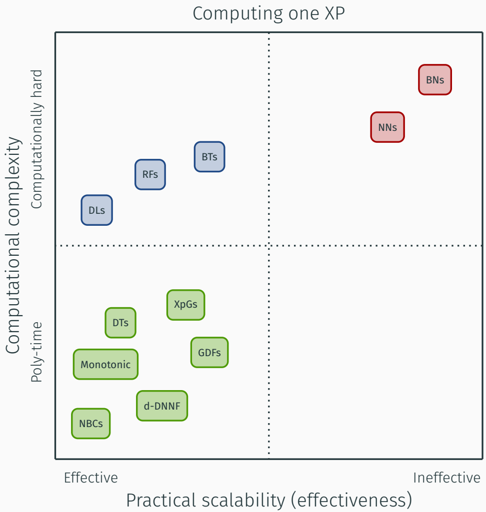
Preliminary results for DTs

[IIM20, HIIM21]

Dataset	(#F	#S)	IAI									ITI								
			D	#N	%A	#P	%R	%C	%m	%M	%avg	D	#N	%A	#P	%R	%C	%m	%M	%avg
adult	(12	6061)	6	83	78	42	33	25	20	40	25	17	509	73	255	75	91	10	66	22
anneal	(38	886)	6	29	99	15	26	16	16	33	21	9	31	100	16	25	4	12	20	16
backache	(32	180)	4	17	72	9	33	39	25	33	30	3	9	91	5	80	87	50	66	54
bank	(19	36 293)	6	113	88	57	5	12	16	20	18	19	1467	86	734	69	64	7	63	27
biodegradation	(41	1052)	5	19	65	10	30	1	25	50	33	8	71	76	36	50	8	14	40	21
cancer	(9	449)	6	37	87	19	36	9	20	25	21	5	21	84	11	54	10	25	50	37
car	(6	1728)	6	43	96	22	86	89	20	80	45	11	57	98	29	65	41	16	50	30
colic	(22	357)	6	55	81	28	46	6	16	33	20	4	17	80	9	33	27	25	25	25
compas	(11	1155)	6	77	34	39	17	8	16	20	17	15	183	37	92	66	43	12	60	27
contraceptive	(9	1425)	6	99	49	50	8	2	20	60	37	17	385	48	193	27	32	12	66	21
dermatology	(34	366)	6	33	90	17	23	3	16	33	21	7	17	95	9	22	0	14	20	17
divorce	(54	150)	5	15	90	8	50	19	20	33	24	2	5	96	3	33	16	50	50	50
german	(21	1000)	6	25	61	13	38	10	20	40	29	10	99	72	50	46	13	12	40	22
heart-c	(13	302)	6	43	65	22	36	18	20	33	22	4	15	75	8	87	81	25	50	34
heart-h	(13	293)	6	37	59	19	31	4	20	40	24	8	25	77	13	61	60	20	50	32
kr-vs-kp	(36	3196)	6	49	96	25	80	75	16	60	33	13	67	99	34	79	43	7	70	35
lending	(9	5082)	6	45	73	23	73	80	16	50	25	14	507	65	254	69	80	12	75	25
letter	(16	18 668)	6	127	58	64	1	0	20	20	20	46	4857	68	2429	6	7	6	25	9
lymphography	(18	148)	6	61	76	31	35	25	16	33	21	6	21	86	11	9	0	16	16	16
mortality	(118	13 442)	6	111	74	56	8	14	16	20	17	26	865	76	433	61	61	7	54	19
mushroom	(22	8124)	6	39	100	20	80	44	16	33	24	5	23	100	12	50	31	20	40	25
pendigits	(16	10 992)	6	121	88	61	0	0	—	—	—	38	937	85	469	25	86	6	25	11
promoters	(58	106)	1	3	90	2	0	0	—	—	—	3	9	81	5	20	14	33	33	33
recidivism	(15	3998)	6	105	61	53	28	22	16	33	18	15	611	51	306	53	38	9	44	16
seismic_bumps	(18	2578)	6	37	89	19	42	19	20	33	24	8	39	93	20	60	79	20	60	42
shuttle	(9	58 000)	6	63	99	32	28	7	20	33	23	23	159	99	80	33	9	14	50	30
soybean	(35	623)	6	63	88	32	9	5	25	25	25	16	71	89	36	22	1	9	12	10
spambase	(57	4210)	6	63	75	32	37	12	16	33	19	15	143	91	72	76	98	7	58	25
spect	(22	228)	6	45	82	23	60	51	20	50	35	6	15	86	8	87	98	50	83	65
splice	(2	3178)	3	7	50	4	0	0	—	—	—	88	177	55	89	0	0	—	—	—

Outline

Efficacy map – current status



[INM19c, Ign20, IIM20, MGC+20, MGC+21, HIIM21, IMS21, IM21, CM21, HII+22, IISMS22]

- **Formal explanations efficient for several families of classifiers**
  - Polynomial-time:
    - Naive-Bayes classifiers (NBCs) [MGC+20]
    - Decision trees (DTs) [IIM20, HIIM21]
    - XpG's: DTs, OBDDs, OMDDs, etc. [HIIM21]
    - Monotonic classifiers [MGC+21]
    - Propositional languages (e.g. d-DNNF, ...) [HII+22]
    - Additional results [CM21, HII+22]
  - Comp. hard, but effective (efficient in practice):
    - Random forests (RFs) [IMS21]
    - Decision lists (DLs) [IM21]
    - Boosted trees (BTs) [INM19c, Ign20, IISMS22]
  - Comp. hard, and ineffective (hard in practice):
    - Neural networks (NNs) [INM19a]
    - Bayesian networks (BNs) [SCD18]

Some recent results – RFs (with SAT)

[IMS21]

Dataset	(#F	#C	#I)	RF			CNF		SAT oracle				PI-expl (**`RFxpl`**)				Anchor	
				D	#N	%A	#var	#cl	MxS	MxU	#S	#U	Mx	m	avg	%w	avg	%w
ann-thyroid	(21	3	718)	4	2192	98	17 854	29 230	0.12	0.15	2	18	0.36	0.05	0.13	96	0.32	4
appendicitis	(7	2	43)	6	1920	90	5181	10 085	0.02	0.02	4	3	0.05	0.01	0.03	100	0.48	0
banknote	(4	2	138)	5	2772	97	8068	16 776	0.01	0.01	2	2	0.03	0.02	0.02	100	0.19	0
biodegradation	(41	2	106)	5	4420	88	11 007	23 842	0.31	1.05	17	22	2.27	0.04	0.29	97	4.07	3
heart-c	(13	2	61)	5	3910	85	5594	11 963	0.04	0.02	6	7	0.07	0.01	0.04	100	0.85	0
ionosphere	(34	2	71)	5	2096	87	7174	14 406	0.02	0.02	22	11	0.11	0.02	0.03	100	12.43	0
karhunen	(64	10	200)	5	6198	91	36 708	70 224	1.06	1.41	35	29	14.64	0.65	2.78	100	28.15	0
letter	(16	26	398)	8	44 304	82	28 991	68 148	1.97	3.31	8	8	6.91	0.24	1.61	70	2.48	30
magic	(10	2	381)	6	9840	84	29 530	66 776	0.51	1.84	6	4	2.13	0.07	0.14	99	0.91	1
new-thyroid	(5	3	43)	5	1766	100	17 443	28 134	0.03	0.01	3	2	0.08	0.03	0.05	100	0.36	0
pendigits	(16	10	220)	6	12 004	95	30 522	59 922	2.40	1.32	10	6	4.11	0.14	0.94	96	3.68	4
ring	(20	2	740)	6	6188	89	19 114	42 362	0.27	0.44	11	9	1.25	0.05	0.25	92	7.25	8
segmentation	(19	7	42)	4	1966	90	21 288	35 381	0.11	0.17	8	10	0.53	0.11	0.31	100	4.13	0
shuttle	(9	7	1160)	3	1460	99	18 669	29 478	0.11	0.08	2	7	0.34	0.05	0.14	99	0.42	1
sonar	(60	2	42)	5	2614	88	9938	20 537	0.04	0.06	36	24	0.43	0.04	0.09	100	23.02	0
spectf	(44	2	54)	5	2306	88	6707	13 449	0.07	0.06	20	24	0.34	0.02	0.07	100	8.12	0
texture	(40	11	550)	5	5724	87	34 293	64 187	0.79	0.63	23	17	3.24	0.19	0.93	100	28.13	0
twonorm	(20	2	740)	5	6266	94	21 198	46 901	0.08	0.08	12	8	0.28	0.06	0.10	100	5.73	0
vowel	(13	11	198)	6	10 176	90	44 523	88 696	1.66	2.11	8	5	4.52	0.15	1.15	66	1.67	34
waveform-40	(40	3	500)	5	6232	83	30 438	58 380	0.50	0.86	15	25	7.07	0.11	0.88	100	11.93	0
wpbc	(33	2	78)	5	2432	76	9078	18 675	1.00	1.53	20	13	5.33	0.03	0.65	79	3.91	21

Outline

Additional computational problems – queries

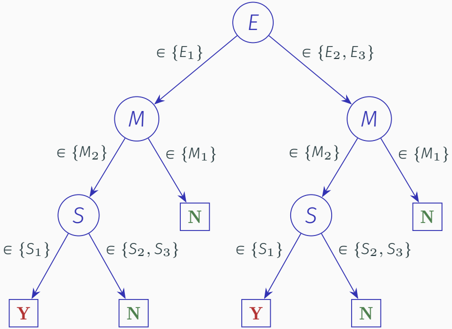
- For a given instance $(\mathbf{v}, c)$:
  - **Enumeration:** list all/some/preferred explanations (AXp's/CXp's)
    - Exploit duality between AXp's & CXp's [INAM20]
    - For classifiers with explanations in P: one SAT oracle call per computed explanation [MGC+21, HIIM21]
    - For NBCs: enumeration with polynomial delay [MGC+20]
  - **Membership:** decide whether there exists explanation that includes target feature
    - Σ_2^{P}-hard for DNF classifiers [HIIM21]
    - In P for DTs [HIIM21]
    - In NP if finding one AXp/CXp in P [HM22]
  - **Probabilistic explanations:** compute set of features which, if fixed, the probability of predicting the target class is sufficiently large
    - NP^{PP}-hard for boolean circuit classifiers [WMHK21]
- Membership & enumeration for a prediction c, independently of point in feature space
- ...

Another DT example

Bank loan default, with $\mathbf{x} = (E, M, S, A)$ and $\mathcal{K} = \{\mathbf{Y}, \mathbf{N}\}$				
Feature	ID	Var.	Domain	Coded Domain
Ethnic Group	1	E	{European, African, Asian}	$\{E_1, E_2, E_3\}$
Value of Mortgage	2	M	$\{\leqslant 100\text{K}, > 100\text{K}\}$	$\{M_1, M_2\}$
Salary of Applicant	3	S	$\{\leqslant 20\text{K}, > 20\text{K} \wedge \leqslant 50\text{K}, > 50\text{K}\}$	$\{S_1, S_2, S_3\}$
Age of Applicant	4	A	$\{\leqslant 40, > 40 \wedge \leqslant 55, > 55\}$	$\{A_1, A_2, A_3\}$

Another DT example

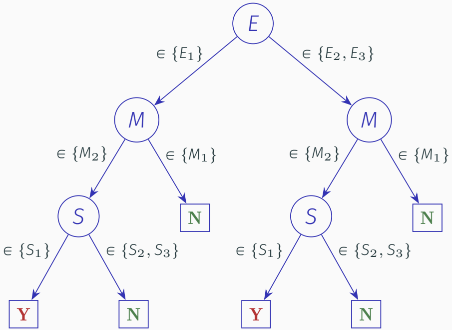
Bank loan default, with $\mathbf{x} = (E, M, S, A)$ and $\mathcal{K} = \{\mathbf{Y}, \mathbf{N}\}$				
Feature	ID	Var.	Domain	Coded Domain
Ethnic Group	1	E	{European, African, Asian}	$\{E_1, E_2, E_3\}$
Value of Mortgage	2	M	$\{\leqslant 100\text{K}, > 100\text{K}\}$	$\{M_1, M_2\}$
Salary of Applicant	3	S	$\{\leqslant 20\text{K}, > 20\text{K} \wedge \leqslant 50\text{K}, > 50\text{K}\}$	$\{S_1, S_2, S_3\}$
Age of Applicant	4	A	$\{\leqslant 40, > 40 \wedge \leqslant 55, > 55\}$	$\{A_1, A_2, A_3\}$



Another DT example

Bank loan default, with $\mathbf{x} = (E, M, S, A)$ and $\mathcal{K} = \{\mathbf{Y}, \mathbf{N}\}$				
Feature	ID	Var.	Domain	Coded Domain
Ethnic Group	1	E	{European, African, Asian}	$\{E_1, E_2, E_3\}$
Value of Mortgage	2	M	$\{\leqslant 100\text{K}, > 100\text{K}\}$	$\{M_1, M_2\}$
Salary of Applicant	3	S	$\{\leqslant 20\text{K}, > 20\text{K} \wedge \leqslant 50\text{K}, > 50\text{K}\}$	$\{S_1, S_2, S_3\}$
Age of Applicant	4	A	$\{\leqslant 40, > 40 \wedge \leqslant 55, > 55\}$	$\{A_1, A_2, A_3\}$

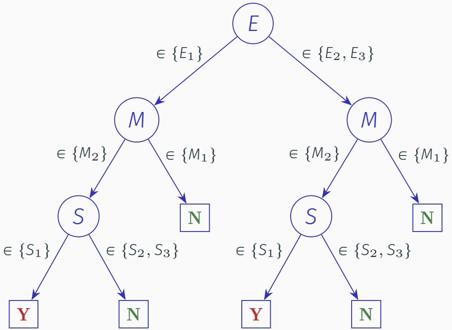
- Instance: $((E_2, M_2, S_1, A_3), \mathbf{Y})$



Another DT example

Bank loan default, with $\mathbf{x} = (E, M, S, A)$ and $\mathcal{K} = \{\mathbf{Y}, \mathbf{N}\}$				
Feature	ID	Var.	Domain	Coded Domain
Ethnic Group	1	E	{European, African, Asian}	$\{E_1, E_2, E_3\}$
Value of Mortgage	2	M	$\{\leqslant 100\text{K}, > 100\text{K}\}$	$\{M_1, M_2\}$
Salary of Applicant	3	S	$\{\leqslant 20\text{K}, > 20\text{K} \wedge \leqslant 50\text{K}, > 50\text{K}\}$	$\{S_1, S_2, S_3\}$
Age of Applicant	4	A	$\{\leqslant 40, > 40 \wedge \leqslant 55, > 55\}$	$\{A_1, A_2, A_3\}$

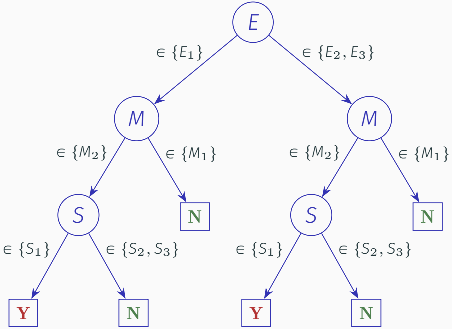
- Instance: $((E_2, M_2, S_1, A_3), \mathbf{Y})$
- Tree path: $(E = E_2, M = M_2, S = S_1)$



Another DT example

Bank loan default, with $\mathbf{x} = (E, M, S, A)$ and $\mathcal{K} = \{\mathbf{Y}, \mathbf{N}\}$				
Feature	ID	Var.	Domain	Coded Domain
Ethnic Group	1	E	{European, African, Asian}	$\{E_1, E_2, E_3\}$
Value of Mortgage	2	M	$\{\leqslant 100\text{K}, > 100\text{K}\}$	$\{M_1, M_2\}$
Salary of Applicant	3	S	$\{\leqslant 20\text{K}, > 20\text{K} \wedge \leqslant 50\text{K}, > 50\text{K}\}$	$\{S_1, S_2, S_3\}$
Age of Applicant	4	A	$\{\leqslant 40, > 40 \wedge \leqslant 55, > 55\}$	$\{A_1, A_2, A_3\}$

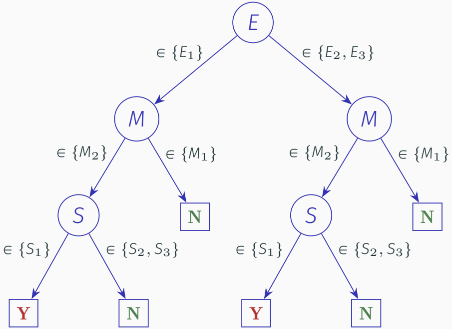
- Instance:$((E_2, M_2, S_1, A_3), \mathbf{Y})$
- Tree path: $(E = E_2, M = M_2, S = S_1)$, i.e. ethnic group matters...



Another DT example

Bank loan default, with $\mathbf{x} = (E, M, S, A)$ and $\mathcal{K} = \{\mathbf{Y}, \mathbf{N}\}$				
Feature	ID	Var.	Domain	Coded Domain
Ethnic Group	1	E	{European, African, Asian}	$\{E_1, E_2, E_3\}$
Value of Mortgage	2	M	$\{\leqslant 100\text{K}, > 100\text{K}\}$	$\{M_1, M_2\}$
Salary of Applicant	3	S	$\{\leqslant 20\text{K}, > 20\text{K} \wedge \leqslant 50\text{K}, > 50\text{K}\}$	$\{S_1, S_2, S_3\}$
Age of Applicant	4	A	$\{\leqslant 40, > 40 \wedge \leqslant 55, > 55\}$	$\{A_1, A_2, A_3\}$

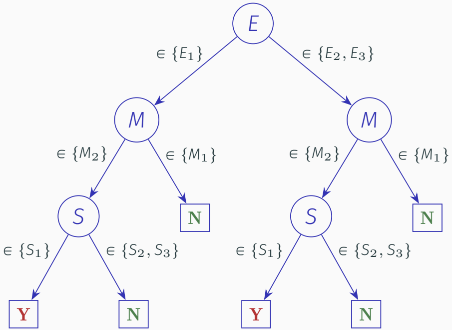
- Instance: $((E_2, M_2, S_1, A_3), \mathbf{Y})$
- Tree path: $(E = E_2, M = M_2, S = S_1)$, i.e. ethnic group matters... but **does it really matter?**



Another DT example

Bank loan default, with $\mathbf{x} = (E, M, S, A)$ and $\mathcal{K} = \{\mathbf{Y}, \mathbf{N}\}$				
Feature	ID	Var.	Domain	Coded Domain
Ethnic Group	1	E	{European, African, Asian}	$\{E_1, E_2, E_3\}$
Value of Mortgage	2	M	$\{\leqslant 100\text{K}, > 100\text{K}\}$	$\{M_1, M_2\}$
Salary of Applicant	3	S	$\{\leqslant 20\text{K}, > 20\text{K} \wedge \leqslant 50\text{K}, > 50\text{K}\}$	$\{S_1, S_2, S_3\}$
Age of Applicant	4	A	$\{\leqslant 40, > 40 \wedge \leqslant 55, > 55\}$	$\{A_1, A_2, A_3\}$

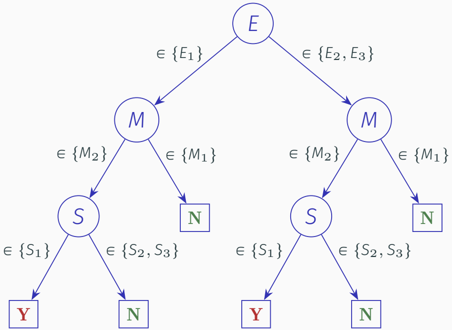
- Instance: $((E_2, M_2, S_1, A_3), \mathbf{Y})$
- Tree path: $(E = E_2, M = M_2, S = S_1)$, i.e. ethnic group matters... but **does it really matter?**
- If E can take any value, prediction does not change



Another DT example

Bank loan default, with $\mathbf{x} = (E, M, S, A)$ and $\mathcal{K} = \{\mathbf{Y}, \mathbf{N}\}$				
Feature	ID	Var.	Domain	Coded Domain
Ethnic Group	1	E	{European, African, Asian}	$\{E_1, E_2, E_3\}$
Value of Mortgage	2	M	$\{\leqslant 100\text{K}, > 100\text{K}\}$	$\{M_1, M_2\}$
Salary of Applicant	3	S	$\{\leqslant 20\text{K}, > 20\text{K} \wedge \leqslant 50\text{K}, > 50\text{K}\}$	$\{S_1, S_2, S_3\}$
Age of Applicant	4	A	$\{\leqslant 40, > 40 \wedge \leqslant 55, > 55\}$	$\{A_1, A_2, A_3\}$

- Instance:$((E_2, M_2, S_1, A_3), \mathbf{Y})$
- Tree path: $(E = E_2, M = M_2, S = S_1)$, i.e. ethnic group matters... but **does it really matter?**
- If E can take any value, prediction does not change
- $\therefore$ if $(M = M_2, S = S_1)$, then prediction is $\mathbf{Y}$, **independently** of ethnic group



Conclusions

- Critical limitations of widely used XAI approaches:
  - Model-agnostic methods can compute **incorrect** explanations
  - Intrinsic interpretability explanations can be (very) redundant

Conclusions

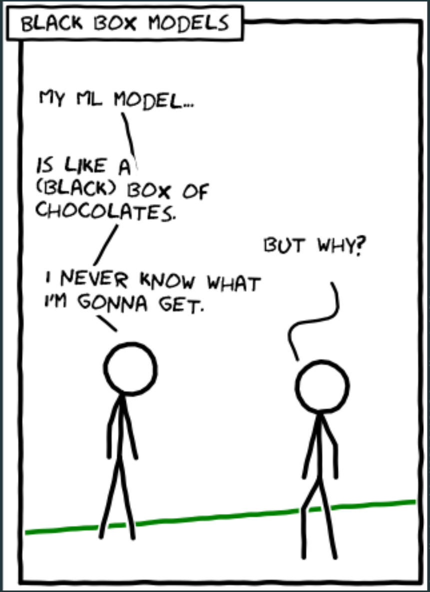
- Critical limitations of widely used XAI approaches:
  - Model-agnostic methods can compute **incorrect** explanations
  - Intrinsic interpretability explanations can be (very) redundant
- Formal explainability in AI (FXAI):
  - Logic-based, rigorous definitions of explanations
    - Initial theoretical insights, e.g. duality between AXp's and CXp's (and so between "Why?" and "Why not?" explanations)
  - Other problems of crucial importance: enumeration, membership, etc.

Conclusions

- Critical limitations of widely used XAI approaches:
  - Model-agnostic methods can compute **incorrect** explanations
  - Intrinsic interpretability explanations can be (very) redundant
- **Formal explainability in AI** (FXAI):
  - Logic-based, rigorous definitions of explanations
    - Initial theoretical insights, e.g. duality between AXp's and CXp's (and so between "Why?" and "Why not?" explanations)
  - Other problems of crucial importance: enumeration, membership, etc.
- Ongoing research related with:
  - Computation of AXp's & CXp's
  - Enumeration of explanations
  - Deciding membership of features in explanations
  - Probabilistic rigorous explanations

Q & A

Acknowledgment: joint work with Y. Izza, X. Huang, M. Cooper, N. Asher, N. Narodytska, E. Hebrard, M. Siala, et al.



https://arxiv.org/abs/1901.01686 & http://cmx.io/edit/

References i

[ANS20] Gaël Aglin, Siegfried Nijssen, and Pierre Schaus.
PyDL8.5: a library for learning optimal decision trees.
In *IJCAI*, pages 5222–5224, 2020.

[Bre01] Leo Breiman.
Statistical modeling: The two cultures.
Statistical science, 16(3):199–231, 2001.

[CM21] Martin C. Cooper and Joao Marques-Silva.
On the tractability of explaining decisions of classifiers.
In *CP*, October 2021.

[EG95] Thomas Eiter and Georg Gottlob.
Identifying the minimal transversals of a hypergraph and related problems.
SIAM J. Comput., 24(6):1278–1304, 1995.

[EU21] EU.
European Artificial Intelligence Act – Proposal.
`https://eur-lex.europa.eu/legal-content/EN/TXT/?qid=1623335154975&uri=CELEX%3A52021PC0206`, 2021.

References ii

[FJ18] Matteo Fischetti and Jason Jo.
Deep neural networks and mixed integer linear optimization.
Constraints, 23(3):296–309, 2018.

[HII+22] Xuanxiang Huang, Yacine Izza, Alexey Ignatiev, Martin Cooper, Nicholas Asher, and Joao Marques-Silva.
Tractable explanations for d-DNNF classifiers.
In *AAAI*, February 2022.

[HIIM21] Xuanxiang Huang, Yacine Izza, Alexey Ignatiev, and Joao Marques-Silva.
On efficiently explaining graph-based classifiers.
In *KR*, November 2021.
Preprint available from `https://arxiv.org/abs/2106.01350`.

[HM22] Xuanxiang Huang and João Marques-Silva.
On deciding feature membership in explanations of SDD & related classifiers.
CoRR, abs/2202.07553, 2022.

[Ign20] Alexey Ignatiev.
Towards trustable explainable AI.
In *IJCAI*, pages 5154–5158, 2020.

[IIM20] Yacine Izza, Alexey Ignatiev, and Joao Marques-Silva.
On explaining decision trees.
CoRR, abs/2010.11034, 2020.

[IISMS22] Alexey Ignatiev, Yacine Izza, Peter J. Stuckey, and Joao Marques-Silva.
Using MaxSAT for efficient explanations of tree ensembles.
In *AAAI*, February 2022.

[IM21] Alexey Ignatiev and Joao Marques-Silva.
SAT-based rigorous explanations for decision lists.
In *SAT*, pages 251–269, July 2021.

[IMS21] Yacine Izza and Joao Marques-Silva.
On explaining random forests with SAT.
In *IJCAI*, pages 2584–2591, July 2021.

[INAM20] Alexey Ignatiev, Nina Narodytska, Nicholas Asher, and João Marques-Silva.
From contrastive to abductive explanations and back again.
In *AIxIA*, pages 335–355, 2020.

[INM19a] Alexey Ignatiev, Nina Narodytska, and Joao Marques-Silva.
Abduction-based explanations for machine learning models.
In *AAAI*, pages 1511–1519, 2019.

[INM19b] Alexey Ignatiev, Nina Narodytska, and Joao Marques-Silva.
On relating explanations and adversarial examples.
In *NeurIPS*, pages 15857–15867, 2019.

[INM19c] Alexey Ignatiev, Nina Narodytska, and Joao Marques-Silva.
On validating, repairing and refining heuristic ML explanations.
CoRR, abs/1907.02509, 2019.

[Int21] Interpretable AI.
https://www.interpretable.ai/, 21.

[Lip18] Zachary C. Lipton.
The mythos of model interpretability.
Commun. ACM, 61(10):36–43, 2018.

[LL17] Scott M. Lundberg and Su-In Lee.
A unified approach to interpreting model predictions.
In *NIPS*, pages 4765–4774, 2017.

[MGC+20] Joao Marques-Silva, Thomas Gerspacher, Martin C. Cooper, Alexey Ignatiev, and Nina Narodytska.
Explaining naive bayes and other linear classifiers with polynomial time and delay.
In *NeurIPS*, 2020.

[MGC+21] Joao Marques-Silva, Thomas Gerspacher, Martinc C. Cooper, Alexey Ignatiev, and Nina Narodytska.
Explanations for monotonic classifiers.
In *ICML*, pages 7469–7479, July 2021.

[Mil19] Tim Miller.
Explanation in artificial intelligence: Insights from the social sciences.
Artif. Intell., 267:1–38, 2019.

[Mol20] Christoph Molnar.
Interpretable machine learning.
Lulu.com, 2020.
`https://christophm.github.io/interpretable-ml-book/`.

[NH10] Vinod Nair and Geoffrey E. Hinton.
Rectified linear units improve restricted boltzmann machines.
In *ICML*, pages 807–814, 2010.

[NSM+19] Nina Narodytska, Aditya A. Shrotri, Kuldeep S. Meel, Alexey Ignatiev, and Joao Marques-Silva.
Assessing heuristic machine learning explanations with model counting.
In *SAT*, pages 267–278, 2019.

References vi

[RSG16] Marco Túlio Ribeiro, Sameer Singh, and Carlos Guestrin.
"why should I trust you?": Explaining the predictions of any classifier.
In *KDD*, pages 1135–1144, 2016.

[RSG18] Marco Túlio Ribeiro, Sameer Singh, and Carlos Guestrin.
Anchors: High-precision model-agnostic explanations.
In *AAAI*, pages 1527–1535. AAAI Press, 2018.

[Rud19] Cynthia Rudin.
Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead.
Nature Machine Intelligence, 1(5):206–215, 2019.

[SCD18] Andy Shih, Arthur Choi, and Adnan Darwiche.
A symbolic approach to explaining bayesian network classifiers.
In *IJCAI*, pages 5103–5111, 2018.

[WMHK21] Stephan Wäldchen, Jan MacDonald, Sascha Hauch, and Gitta Kutyniok.
The computational complexity of understanding binary classifier decisions.
J. Artif. Intell. Res., 70:351–387, 2021.