Machine Learning Meets Automated Reasoning: Explainability, Fairness, Robustness and Model Learning

Joao Marques-Silva

ANITI, IRIT & CNRS, Toulouse, France

November 2020

Context – my team's recent & not so recent work...

- SAT Solving (Clause learning, UIPs, ...)
- Quantification & CEGAR (QBF, QMaxSAT, etc.)
- Function Synthesis (Min DNF cover, ...)
- Inconsistency (MUS, MCS, etc.)
- Certification of Reasoners
- Model Checking, Synthesizing Invariants, ATPG, Reconfiguration
- Optimization (MaxSAT, MinSAT, PBO, WBO, etc.)
- Propositional Encodings, Backbones, Autarkies, Minimal models, etc.
- Enumeration (MUSes, MCSes, etc.)
- Proof Systems (DRMaxSAT, etc.)
- Primes, Abduction, DLs, etc.

Context – new area of research, since 2018...

SAT Solving (Clause learning, UIPs, ...)	Quantification & CEGAR (QBF, QMaxSAT, etc.)	Function Synthesis (Min DNF cover, ...)
Inconsistency (MUS, MCS, etc.)	Certification of Reasoners	Model Checking, Synthesizing Invariants, ATPG, Reconfiguration
Optimization (MaxSAT, MinSAT, PBO, WBO, etc.)	Propositional Encodings, Backbones, Autarkies, Minimal models, etc.	Enumeration (MUSes, MCSes, etc.)
Proof Systems (DRMaxSAT, etc.)	Primes, Abduction, DLs, etc.	Explainability & Interpretability in ML

Context – new area of research, since 2018...

SAT Solving (Clause learning, UIPs, ...)

Quantification & CEGAR (QBF, QMaxSAT, etc.)

Function Synthesis (Min DNF cover, ...)

Inconsistency (MUS, MCS, etc.)

Certification of Reasoners

Model Checking, Synthesizing Invariants, ATPG, Reconfiguration

Optimization (MaxSAT, MinSAT, PBO, WBO, etc.)

Propositional Encodings, Backbones, Autarkies, Minimal models, etc.

Enum

(MU

Understanding how to apply AR & FM in ML !

Proof Systems (DRMaxSAT, etc.)

Primes, Abduction, DLs, etc.

Explainability & Interpretability in ML

Recent & ongoing ML successes

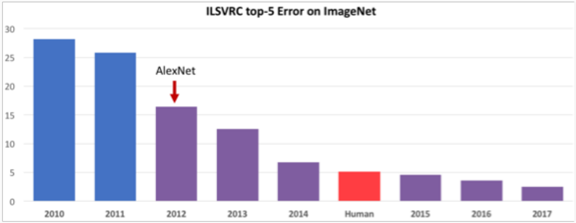
https://en.wikipedia.org/wiki/Waymo

DeepMind

AlphaGo

AlphaGo Zero & **Alpha Zero**

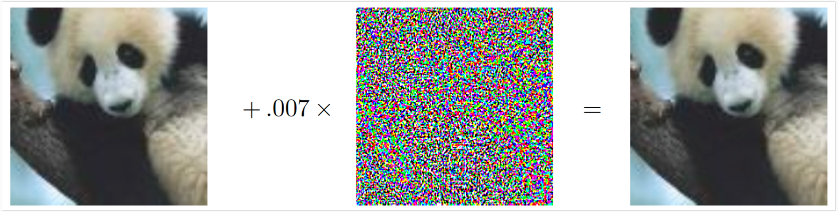
Image & Speech Recognition



http://gradientscience.org/intro_adversarial/

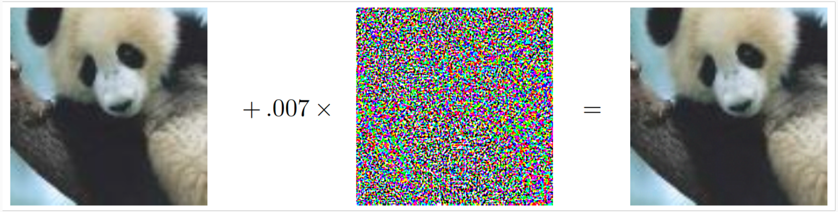
https://fr.wikipedia.org/wiki/Pepper_(robot)

But ML models are brittle — adversarial examples



Goodfellow et al., ICLR'15

But ML models are brittle — adversarial examples



Goodfellow et al., ICLR'15

Eykholt et al'18

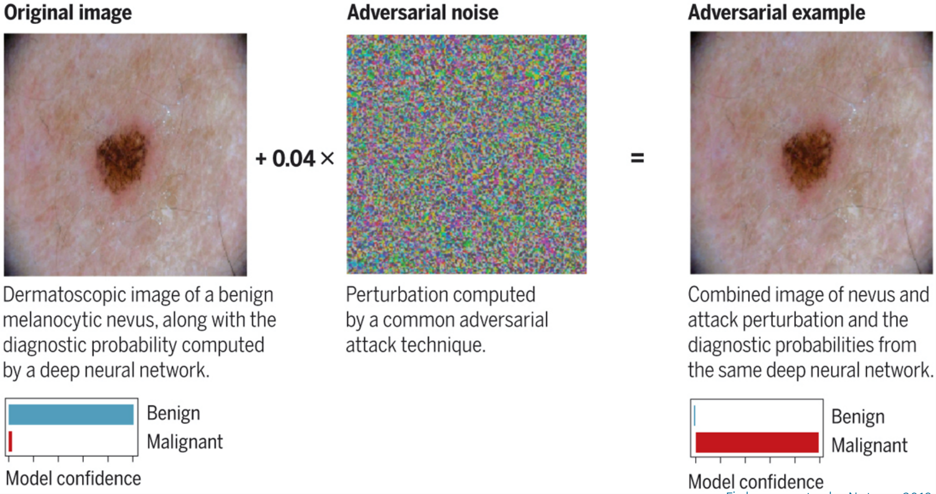
Aung et al'17

But ML models are brittle — adversarial examples

Eykholt et al'18

Aung et al'17

Adversarial examples can be very problematic



Finlayson et al., Nature 2019

Also, some ML models are interpretable

decision|rule lists|sets
decision trees; ...

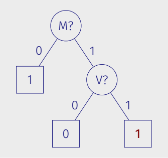
Ex.	Vacation (V)	Concert (C)	Meeting (M)	Expo (E)	Hike (H)
e_1	0	0	1	0	0
e_2	1	0	0	0	1
e_3	0	0	1	1	0
e_4	1	0	0	1	1
e_5	0	1	1	0	0
e_6	0	1	1	1	0
e_7	1	1	0	1	1

Also, some ML models are interpretable

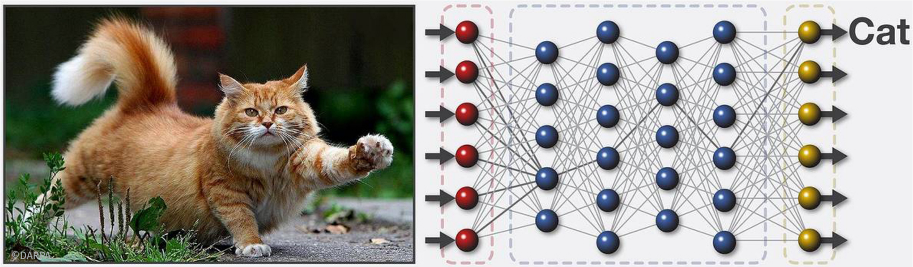
decision|rule lists|sets
decision trees; ...

if ¬Meeting **then** Hike
if ¬Vacation **then** ¬Hike

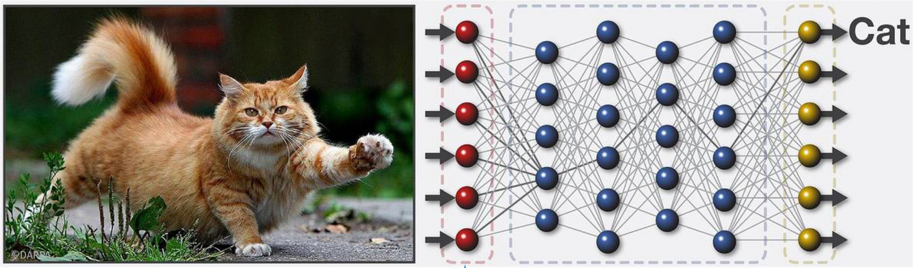
Ex.	Vacation (V)	Concert (C)	Meeting (M)	Expo (E)	Hike (H)
e_1	0	0	1	0	0
e_2	1	0	0	0	1
e_3	0	0	1	1	0
e_4	1	0	0	1	1
e_5	0	1	1	0	0
e_6	0	1	1	1	0
e_7	1	1	0	1	1



But other ML models are not (interpretable)...

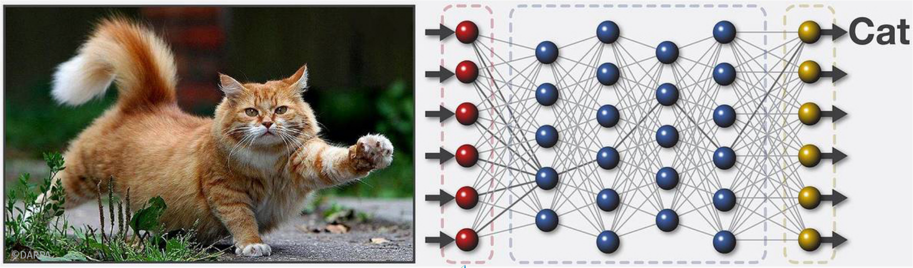


But other ML models are not (interpretable)...



Why does the NN predict a cat?

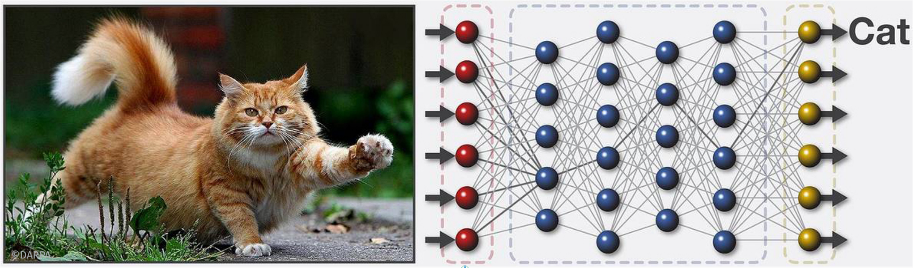
But other ML models are not (interpretable)...



Why does the NN predict a cat?

Which features matter?

But other ML models are not (interpretable)...

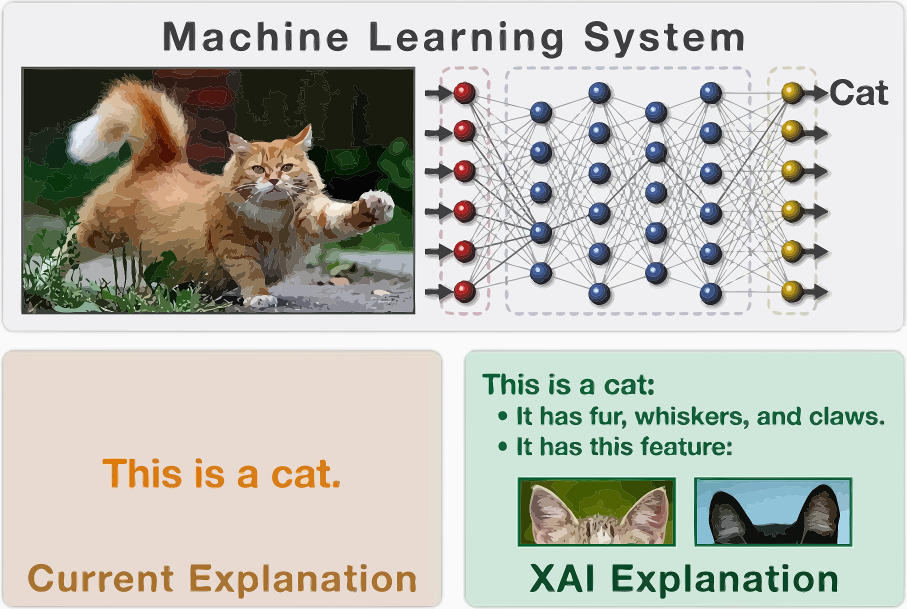


Why does the NN predict a cat?

Which features matter?

Are there general explanations??

What is eXplainable AI (XAI)?



©DARPA

Why 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)

Why 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]

Why 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]

■ *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.*

Why 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]

POLICY \ US & WORLD \ TECH \ TheVerge.com

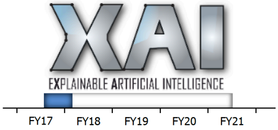
A new bill would force companies to check their algorithms for bias

By Adi Robertson | @thedextriarchy | Apr 10, 2019, 3:52pm EDT

Algorithmic Accountability Act

■ *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)



David Gunning
DARPA/I2O
Program Update November 2017

DARPA

Why XAI?

REGULATION (EU) 2016/67... THE COUNCIL

on the p... ...ta and on the free
move... ...tion Regulation)

European Union regulation
and a "righ

Bryce Goodma

TheVerge.com

...mpanies to check their

Algorithmic Accountability Act

■ *We summarize the potential im... that the European Union's new Gen... Data Protection Regulation will have ... the routine use of machine-learni... algorithms. Slated to take effect as la... across the European Union in 2018, ... will place restrictions on automate... 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.*

In order to trust deployed AI systems, we must not only improve their robustness,[5] but also develop ways to make their reasoning intelligible. Intelligibility will help us spot AI that makes mistakes due to distributional drift or incomplete representations of goals and features. Intelligibility will also facilitate control by humans in increasingly common collaborative human/AI teams. Furthermore, intelligibility will help humans learn from AI. Finally, there are legal reasons to want intelligible AI, including the European GDPR and a growing need to assign liability when AI errs.

Weld & Bansal, CACM, Jun'19

...ence (XAI)

...ber 2017

DARPA

XAI & EU guidelines (AI HLEG)

Search

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

Digital Single Market

REPORT / STUDY | 8 April 2019

Ethics guidelines for trustworthy AI

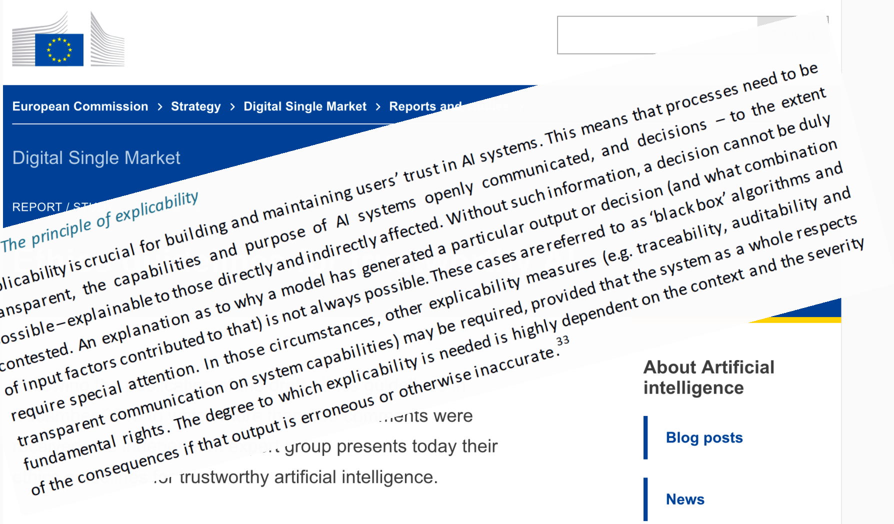
Following the publication of the draft ethics guidelines in December 2018 to which more than 500 comments were received, the independent expert group presents today their ethics guidelines for trustworthy artificial intelligence.

About Artificial intelligence

Blog posts

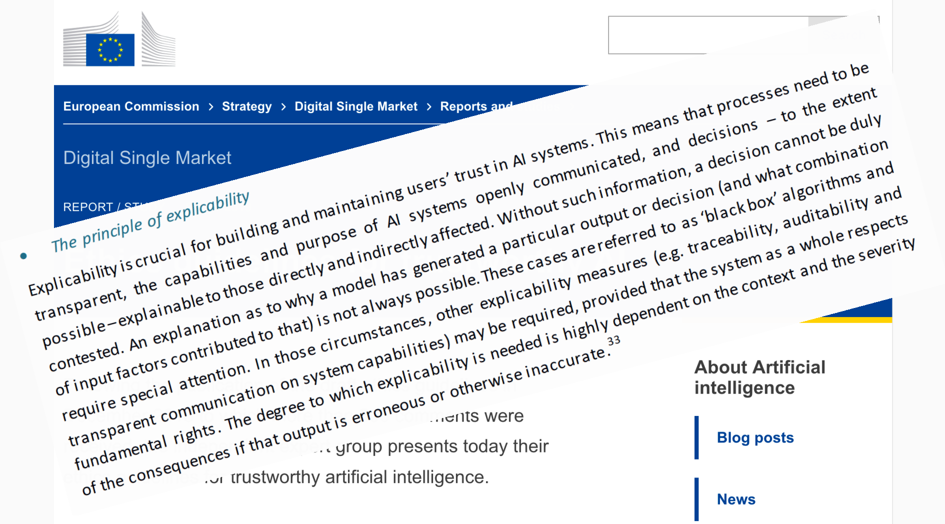
News

XAI & the principle of explicability



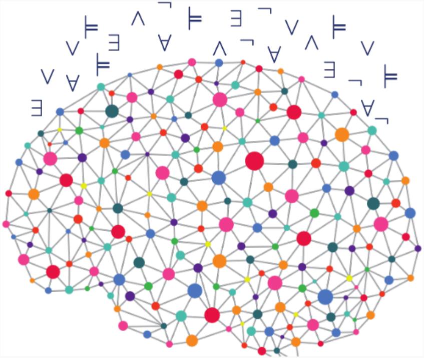
The principle of explicability

- Explicability is crucial for building and maintaining users' trust in AI systems. This means that processes need to be transparent, the capabilities and purpose of AI systems openly communicated, and decisions – to the extent possible – explainable to those directly and indirectly affected. Without such information, a decision cannot be duly contested. An explanation as to why a model has generated a particular output or decision (and what combination of input factors contributed to that) is not always possible. These cases are referred to as 'black box' algorithms and require special attention. In those circumstances, other explicability measures (e.g. traceability, auditability and transparent communication on system capabilities) may be required, provided that the system as a whole respects fundamental rights. The degree to which explicability is needed is highly dependent on the context and the severity of the consequences if that output is erroneous or otherwise inaccurate.[33]



& hundreds of recent papers!

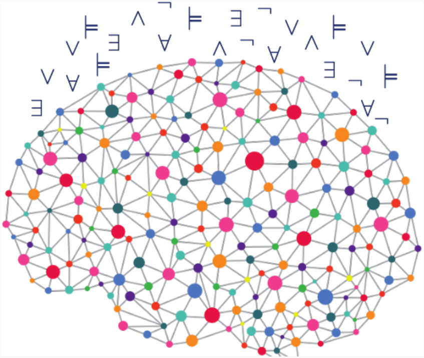
ML vs. AR



"Combining machine learning with logic is the challenge of the day"

M. Vardi, MLmFM'18 Summit

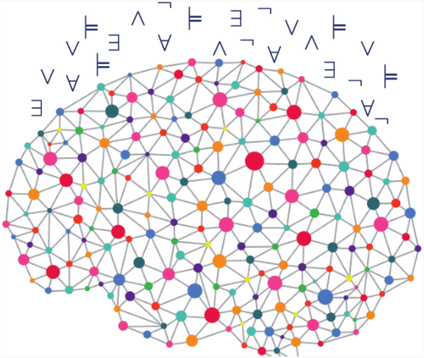
ML vs. AR – among today's grand challenges?



"Combining machine learning with logic is the challenge of the day"

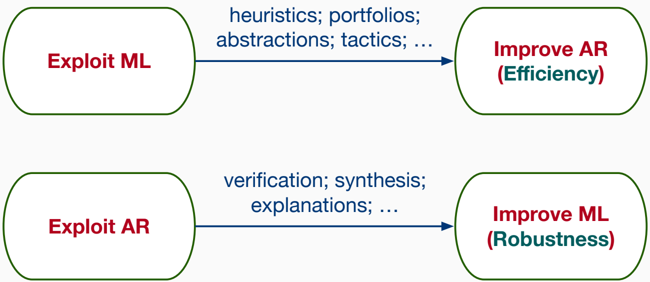
M. Vardi, MLmFM'18 Summit

ML vs. AR – among today's grand challenges?

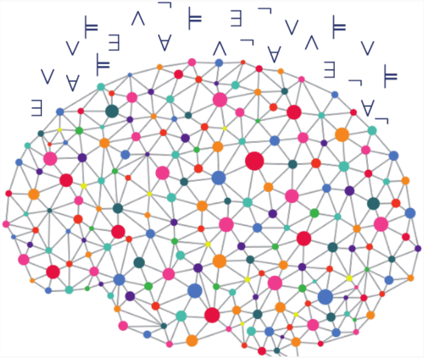


"Combining machine learning with logic is the challenge of the day"

M. Vardi, MLmFM'18 Summit

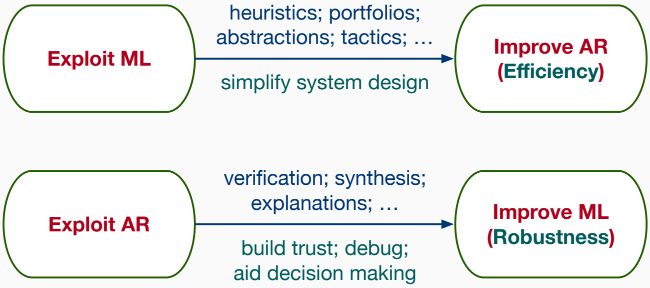


ML vs. AR – among today's grand challenges?

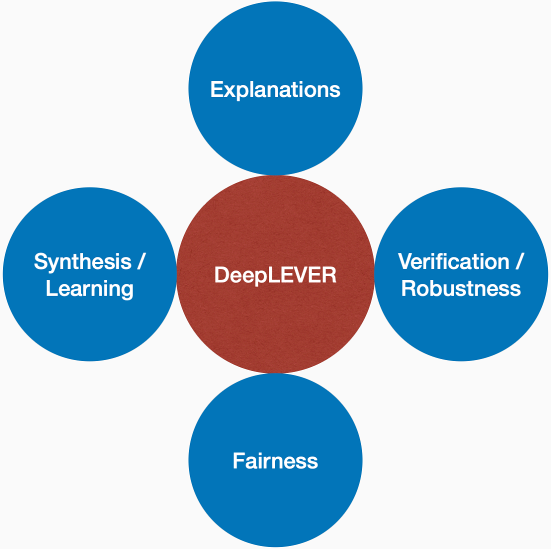


"Combining machine learning with logic is the challenge of the day"

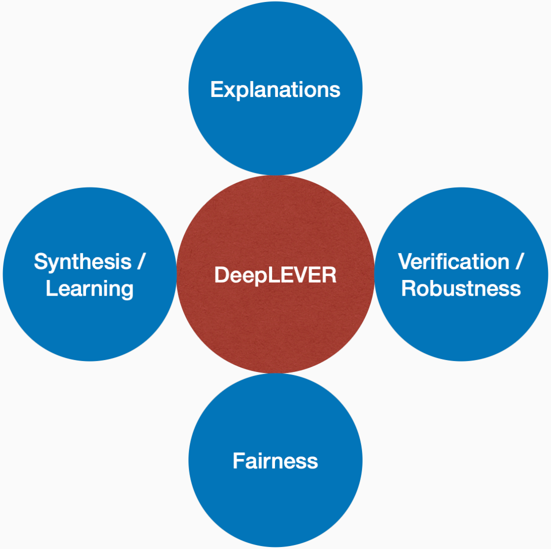
M. Vardi, MLmFM'18 Summit



ANITI's DeepLEVER chair – our current work



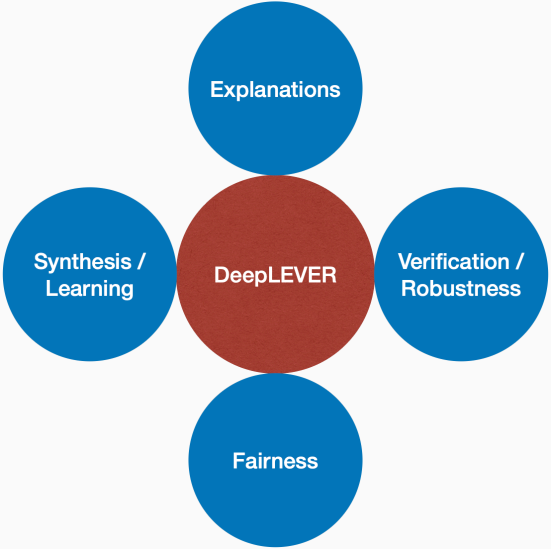
ANITI's DeepLEVER chair – our current work



Explanations

- What is a rigorous explanation?
- Which explanations to compute?
- Computing rigorous explanations
- Assessing heuristic explanations
- Heuristic explanations (with guarantees)
- Tractable explanations
- *High-level* explanations?
- ...

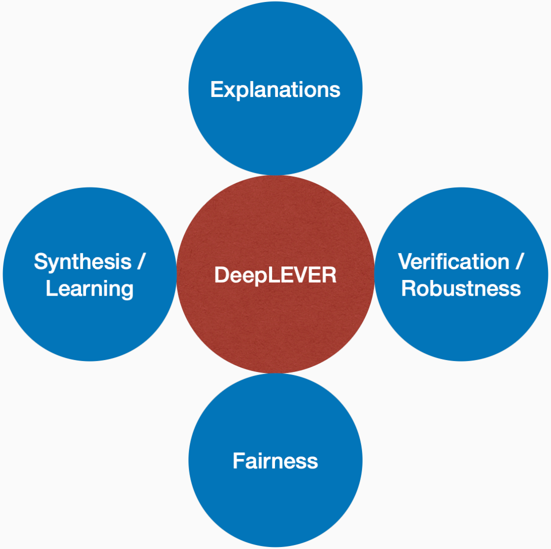
[INM19a, INM19b, INM19c, Ign20, MGC+20]



Synthesis/Learning

- Learning ML models can be cast as a function synthesis problem
  - Learning optimal decision trees and sets
  - Can conceivably exploit constraint/logic based methods to synthesize **any** ML model
    - Scalability is a known issue!
- What about synthesis for robustness?
- What about synthesis for fairness?

[NIPM18, IPNM18, YISB20, HSHH20]

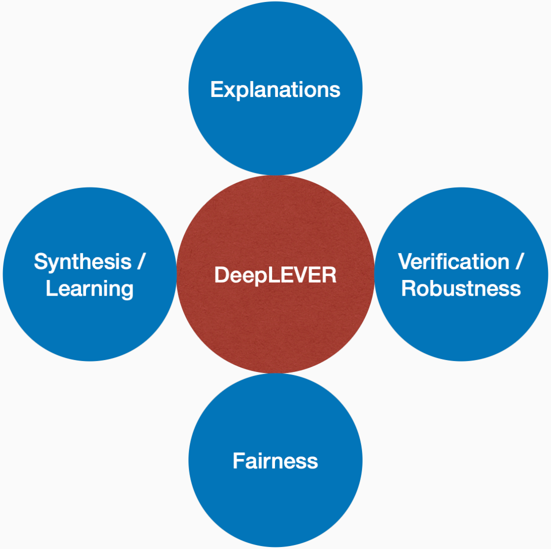


Fairness

- Which fairness criteria to use?
- Dataset bias vs. model fairness
- Links with explainability
- Links with robustness

[ICS+20]

ANITI's DeepLEVER chair – our current work



Verification/Robustness

- More efficient reasoning tools
  - E.g. more efficient NN reasoning?
- More effective/compact constraint-based encodings
- Alternatives to neural networks
  - Binarized NNs
  - Extensions of BTs, (D)RFs, etc.

Today's lecture

- Part #1: Preliminaries
  - Logic-based representations of ML models
- Part #2: Explainability
  - Formal explanations vs. heuristic explanations
  - Tractable explanations
  - Duality in explanations
- Part #3: Fairness
  - First inroads into applying formal methods in fairness
- Part #4: Learning (interpretable models)
  - Learning decision sets (DSs) & decision trees (DTs)
- Part #5: Robustness (brief comments)
  - Applying formal methods in validating robustness of ML models

Part 1

Preliminaries

Outline

Classification problems

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

Classification problems

- Set of features $\mathcal{F} = \{1, 2, \ldots, n\}$, each taking values from a domain D_i
  - Features can be categorical or ordinal, discrete or real-valued
  - Feature space: $\mathbb{F} = \Pi_{i=1}^{n} D_i$
- ML model $\mathbb{M}$ computes classification function $\varphi : \mathbb{F} \rightarrow \mathcal{K}$
  - For simplicity, we will use $\mathcal{K} = \{\boxplus, \boxminus\}$

Classification problems

- Set of features $\mathcal{F} = \{1, 2, \ldots, n\}$, each taking values from a domain D_i
  - Features can be categorical or ordinal, discrete or real-valued
  - Feature space: $\mathbb{F} = \Pi_{i=1}^{n} D_i$
- ML model $\mathbb{M}$ computes classification function $\varphi : \mathbb{F} \to \mathcal{K}$
  - For simplicity, we will use $\mathcal{K} = \{\boxplus, \boxminus\}$
- Instance $\mathbf{v} \in \mathbb{F}$, with prediction $c = \varphi(\mathbf{v})$, $c \in \mathcal{K}$
  - **Obs:** instance $\approx$ example $\approx$ sample $\approx$ point

Classification problems

- Set of features $\mathcal{F} = \{1, 2, \ldots, n\}$, each taking values from a domain D_i
  - Features can be categorical or ordinal, discrete or real-valued
  - Feature space: $\mathbb{F} = \Pi_{i=1}^{n} D_i$
- ML model $\mathbb{M}$ computes classification function $\varphi : \mathbb{F} \rightarrow \mathcal{K}$
  - For simplicity, we will use $\mathcal{K} = \{\boxplus, \boxminus\}$
- Instance $\mathbf{v} \in \mathbb{F}$, with prediction $c = \varphi(\mathbf{v})$, $c \in \mathcal{K}$
  - **Obs:** instance $\approx$ example $\approx$ sample $\approx$ point
- Each $\mathbf{v} \in \mathbb{F}$ is also represented as a set of literals, $\mathcal{C}_{\mathbf{v}} = \{(x_i = v_i) | i \in \mathcal{F}\}$
  - For boolean features, $x_i = 0$ represented by $\neg x_i$ and $x_i = 1$ represented by x_i

Outline

Entailment

- Let φ represent some formula, defined on feature space $\mathbb{F}$, and representing a function $\varphi : \mathbb{F} \rightarrow \{0, 1\}$
- Let τ represent some other formula, also defined on $\mathbb{F}$, and with $\tau : \mathbb{F} \rightarrow \{0, 1\}$

Entailment

- Let φ represent some formula, defined on feature space $\mathbb{F}$, and representing a function $\varphi : \mathbb{F} \to \{0, 1\}$
- Let τ represent some other formula, also defined on $\mathbb{F}$, and with $\tau : \mathbb{F} \to \{0, 1\}$
- We say that τ **entails** φ, written as $\tau \vDash \varphi$, if:

$$\forall(\mathbf{x} \in \mathbb{F}).[\tau(\mathbf{x}) \to \varphi(\mathbf{x})]$$

Entailment

- Let φ represent some formula, defined on feature space $\mathbb{F}$, and representing a function $\varphi : \mathbb{F} \to \{0, 1\}$
- Let τ represent some other formula, also defined on $\mathbb{F}$, and with $\tau : \mathbb{F} \to \{0, 1\}$
- We say that τ **entails** φ, written as $\tau \vDash \varphi$, if:

$$\forall(\mathbf{x} \in \mathbb{F}).[\tau(\mathbf{x}) \to \varphi(\mathbf{x})]$$

  - An example:
    - $\mathbb{F} = \{0, 1\}^2$
    - $\varphi(x_1, x_2) = x_1 \lor \neg x_2$
    - Clearly, $x_1 \vDash \varphi$ and $\neg x_2 \vDash \varphi$

Entailment

- Let φ represent some formula, defined on feature space $\mathbb{F}$, and representing a function $\varphi : \mathbb{F} \to \{0, 1\}$
- Let τ represent some other formula, also defined on $\mathbb{F}$, and with $\tau : \mathbb{F} \to \{0, 1\}$
- We say that τ **entails** φ, written as $\tau \vDash \varphi$, if:

$$\forall(\mathbf{x} \in \mathbb{F}).[\tau(\mathbf{x}) \to \varphi(\mathbf{x})]$$

  - An example:
    - $\mathbb{F} = \{0, 1\}^2$
    - $\varphi(x_1, x_2) = x_1 \lor \neg x_2$
    - Clearly, $x_1 \vDash \varphi$ and $\neg x_2 \vDash \varphi$
  - Another example:
    - $\mathbb{F} = \{0, 1\}^3$
    - $\varphi(x_1, x_2, x_3) = x_1 \land x_2 \lor x_1 \land x_3$
    - Clearly, $x_1 \land x_2 \vDash \varphi$ and $x_1 \land x_3 \vDash \varphi$

Entailment

- Let φ represent some formula, defined on feature space $\mathbb{F}$, and representing a function $\varphi : \mathbb{F} \to \{0, 1\}$
- Let τ represent some other formula, also defined on $\mathbb{F}$, and with $\tau : \mathbb{F} \to \{0, 1\}$
- We say that τ **entails** φ, written as $\tau \vDash \varphi$, if:

$$\forall(\mathbf{x} \in \mathbb{F}).[\tau(\mathbf{x}) \to \varphi(\mathbf{x})]$$

  - An example:
    - $\mathbb{F} = \{0, 1\}^2$
    - $\varphi(x_1, x_2) = x_1 \vee \neg x_2$
    - Clearly, $x_1 \vDash \varphi$ and $\neg x_2 \vDash \varphi$
  - Another example:
    - $\mathbb{F} = \{0, 1\}^3$
    - $\varphi(x_1, x_2, x_3) = x_1 \wedge x_2 \vee x_1 \wedge x_3$
    - Clearly, $x_1 \wedge x_2 \vDash \varphi$ and $x_1 \wedge x_3 \vDash \varphi$
- For non-boolean feature spaces, we let φ_c denote the predicate $\varphi(\mathbf{x}) = c$, i.e. $\varphi_c(\mathbf{x}) \in \{0, 1\}$

Prime implicants & implicates

- A conjunction of literals π (which will be viewed as a set of literals where convenient) is a prime implicant of some function φ if,
    1. $\pi \models \varphi$
    2. For any $\pi' \subsetneq \pi$, $\pi' \not\models \varphi$

Prime implicants & implicates

- A conjunction of literals π (which will be viewed as a set of literals where convenient) is a prime implicant of some function φ if,
  1. $\pi \models \varphi$
  2. For any $\pi' \subsetneq \pi$, $\pi' \not\models \varphi$
  - Example:
    - $\mathbb{F} = \{0, 1\}^3$
    - $\varphi(x_1, x_2, x_3) = x_1 \wedge x_2 \vee x_1 \wedge x_3$
    - Clearly, $x_1 \wedge x_2 \models \varphi$
    - Also, $x_1 \not\models \varphi$ and $x_2 \not\models \varphi$

Prime implicants & implicates

- A conjunction of literals π (which will be viewed as a set of literals where convenient) is a prime implicant of some function φ if,
  1. $\pi \models \varphi$
  2. For any $\pi' \subsetneq \pi$, $\pi' \not\models \varphi$
  - Example:
    - $\mathbb{F} = \{0, 1\}^3$
    - $\varphi(x_1, x_2, x_3) = x_1 \wedge x_2 \vee x_1 \wedge x_3$
    - Clearly, $x_1 \wedge x_2 \models \varphi$
    - Also, $x_1 \not\models \varphi$ and $x_2 \not\models \varphi$
- A disjunction of literals ρ (also viewed as a set of literals where convenient) is a prime implicate of some function φ if
  1. $\varphi \models \rho$
  2. For any $\rho' \subsetneq \rho$, $\varphi \not\models \rho'$

Recap tools of the trade

- SAT: decision problem for propositional logic
  - Formulas most often represented in CNF
  - There are optimization variants: MaxSAT, PBO, MinSAT, etc.
  - There are quantified variants: QBF, QMaxSAT, etc.
- SMT: decision problem for (decidable) fragments of first-order logic (FOL)
  - There are optimization variants: MaxSMT, etc.
  - There are quantified variants
- MILP: decision/optimization problems defined on conjunctions of linear inequalities over integer & real-valued variables
- CP: constraint programming
  - There are optimization/quantified variants

Recap tools of the trade

- SAT: decision problem for propositional logic
  - Formulas most often represented in CNF
  - There are optimization variants: MaxSAT, PBO, MinSAT, etc.
  - There are quantified variants: QBF, QMaxSAT, etc.

Lecture on SAT & SMT assumed. See links below.

- SMT: decision problem for (decidable) fragments of first-order logic (FOL)
  - There are optimization variants: MaxSMT, etc.
  - There are quantified variants
- MILP: decision/optimization problems defined on conjunctions of linear inequalities over integer & real-valued variables
- CP: constraint programming
  - There are optimization/quantified variants
- Background on SAT/SMT:
  - `https://alexeyignatiev.github.io/ssa-school-2019/`
  - `https://alexeyignatiev.github.io/ijcai19tut/`

Outline

Rules with ordinal features

- Example ML model:

  Features: $x_1, x_2 \in \{0, 1, 2\}$ (integer)

  Rules:

$$\text{IF} \quad 2x_1 + x_2 > 0 \quad \text{THEN} \quad \text{predict } \boxplus$$
$$\text{IF} \quad 2x_1 - x_2 \leqslant 0 \quad \text{THEN} \quad \text{predict } \boxminus$$

Rules with ordinal features

- Example ML model:

  Features: $x_1, x_2 \in \{0, 1, 2\}$ (integer)

  Rules:

  $$\text{IF} \quad 2x_1 + x_2 > 0 \quad \text{THEN} \quad \text{predict } \boxplus$$
  $$\text{IF} \quad 2x_1 - x_2 \leqslant 0 \quad \text{THEN} \quad \text{predict } \boxminus$$

- **Q:** Can the model predict both $\boxplus$ and $\boxminus$ for some instance?

Rules with ordinal features

- Example ML model:

  Features: $x_1, x_2 \in \{0, 1, 2\}$ (integer)

  Rules:

  $$\text{IF} \quad 2x_1 + x_2 > 0 \quad \text{THEN} \quad \text{predict } \boxplus$$
  $$\text{IF} \quad 2x_1 - x_2 \leqslant 0 \quad \text{THEN} \quad \text{predict } \boxminus$$

- **Q:** Can the model predict both $\boxplus$ and $\boxminus$ for some instance?
  - Yes, of course: pick $x_1 = 0$ and $x_2 = 1$

Rules with ordinal features

- Example ML model:

  Features: $x_1, x_2 \in \{0, 1, 2\}$ (integer)

  Rules:

  $$\text{IF} \quad 2x_1 + x_2 > 0 \quad \text{THEN} \quad \text{predict } \boxplus$$
  $$\text{IF} \quad 2x_1 - x_2 \leqslant 0 \quad \text{THEN} \quad \text{predict } \boxminus$$

- **Q:** Can the model predict both $\boxplus$ and $\boxminus$ for some instance?
  - Yes, of course: pick $x_1 = 0$ and $x_2 = 1$
  - A formalization:

$$y_p \leftrightarrow (2x_1 + x_2 > 0) \;\wedge\; y_n \leftrightarrow (2x_1 - x_2 \leqslant 0) \;\wedge\; (y_p) \;\wedge\; (y_n)$$

... and solve with SMT solver

$\therefore$ There exists a model iff there exists a point in feature space yielding both predictions

Decision sets

- Example ML model:

  Features: $x_1, x_2 \in \{0, 1\}$ (boolean)

  Rules:

$$\begin{array}{llll} \text{IF} & x_1 \wedge \neg x_2 \wedge x_3 & \text{THEN} & \text{predict } \boxplus \\ \text{IF} & x_1 \wedge \neg x_3 \wedge x_4 & \text{THEN} & \text{predict } \boxminus \\ \text{IF} & x_3 \wedge x_4 & \text{THEN} & \text{predict } \boxminus \end{array}$$

Decision sets

- Example ML model:

  Features: $x_1, x_2 \in \{0, 1\}$ (boolean)

  Rules:

  $$\begin{array}{llll} \text{IF} & x_1 \wedge \neg x_2 \wedge x_3 & \text{THEN} & \text{predict } \boxplus \\ \text{IF} & x_1 \wedge \neg x_3 \wedge x_4 & \text{THEN} & \text{predict } \boxminus \\ \text{IF} & x_3 \wedge x_4 & \text{THEN} & \text{predict } \boxminus \end{array}$$

- **Q:** Can the model predict both $\boxplus$ and $\boxminus$ for some instance?

Decision sets

- Example ML model:

  Features: $x_1, x_2 \in \{0,1\}$ (boolean)

  Rules:

  $$\begin{array}{llll} \text{IF} & x_1 \wedge \neg x_2 \wedge x_3 & \text{THEN} & \text{predict } \boxplus \\ \text{IF} & x_1 \wedge \neg x_3 \wedge x_4 & \text{THEN} & \text{predict } \boxminus \\ \text{IF} & x_3 \wedge x_4 & \text{THEN} & \text{predict } \boxminus \end{array}$$

- **Q:** Can the model predict both $\boxplus$ and $\boxminus$ for some instance?
  - Yes, certainly: pick $(x_1, x_2, x_3, x_4) = (1, 0, 1, 1)$

Decision sets

- Example ML model:

  Features: $x_1, x_2 \in \{0,1\}$ (boolean)

  Rules:

$$\begin{array}{llll} \text{IF} & x_1 \land \neg x_2 \land x_3 & \text{THEN} & \text{predict } \boxplus \\ \text{IF} & x_1 \land \neg x_3 \land x_4 & \text{THEN} & \text{predict } \boxminus \\ \text{IF} & x_3 \land x_4 & \text{THEN} & \text{predict } \boxminus \end{array}$$

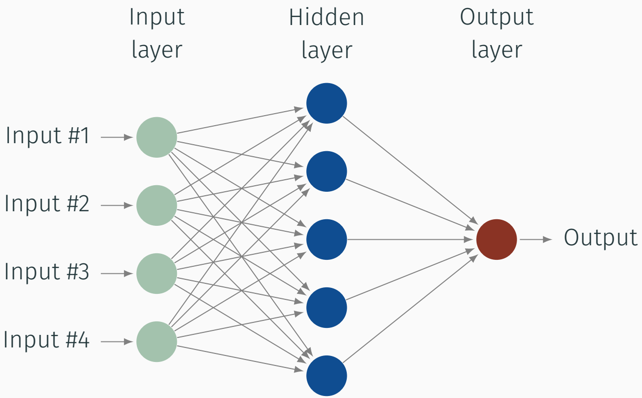
- **Q:** Can the model predict both $\boxplus$ and $\boxminus$ for some instance?
  - Yes, certainly: pick $(x_1, x_2, x_3, x_4) = (1, 0, 1, 1)$
  - A formalization:

$$\begin{array}{l} y_{p,1} \leftrightarrow (x_1 \land \neg x_2 \land x_3) \land \\ y_{n,1} \leftrightarrow (x_1 \land \neg x_3 \land x_4) \land \\ y_{n,2} \leftrightarrow (x_3 \land x_4) \land (y_p \leftrightarrow y_{p,1}) \land \\ (y_n \leftrightarrow (y_{n,1} \lor y_{n,2})) \land (y_p) \land (y_n) \end{array}$$

  ... and solve with SAT solver (after clausification)

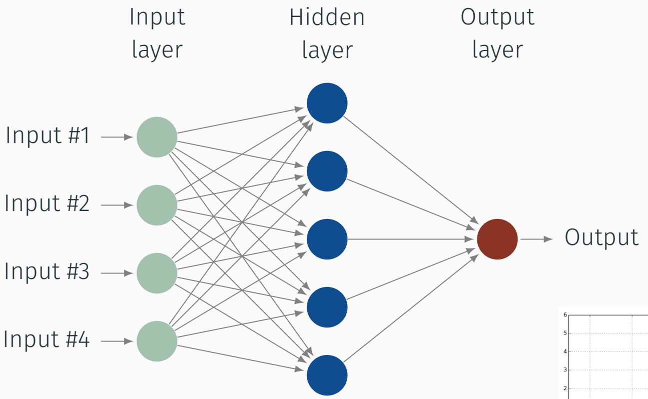
  $\therefore$ There exists a model iff there exists a point in feature space yielding both predictions

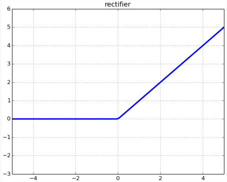
Neural networks



- 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

Neural networks





- 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 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]

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

Simpler encodings exist, but **not** as effective [KBD+17]

Encoding NNs using MILP

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

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

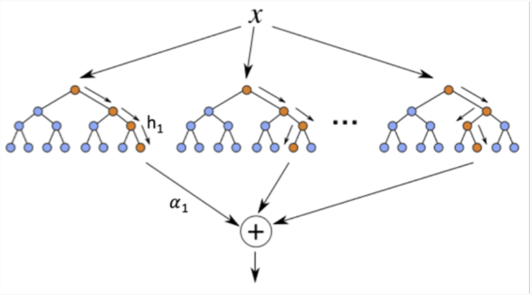
Modeling ML models with logic is not only possible but also simple !

Encoding each **block**: [FJ18]

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

Simpler encodings exist, but **not** as effective [KBD+17]

Boosted trees – glimpse of SMT encoding



https://ieeexplore.ieee.org/abstract/document/8374841

- Number of trees: $m \times q$, with m classes and q trees per class
- Each non-leaf represented by literal (f_j is true?)
    - Associate boolean variable with literal: $b_i \leftrightarrow (f_i?)$
- Each leaf node represented by some real value
- For each path in each tree:
    - If path condition holds, then tree value is leaf value

$$\bigwedge_{n_j \in R_p} b_{n_j.idx} \bigwedge_{n_j \in L_p} \neg b_{n_j.idx} \rightarrow r_l = n_d.val$$

- Score of class j is sum over its q trees: $v_j = \sum_{l=1}^{q} r_{qj+l}$

Questions for part 1?

Part 2

Explainability

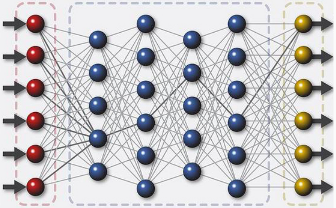
Outline

Computing explanations – assumptions

[INM19a]

- Categorical features, $\mathcal{F} = \{1, 2, \ldots, n\}$, each taking values from a(n unordered) domain D_i
- Feature space: $\mathbb{F} = \Pi_{i=1}^{n} D_i$
- ML model $\mathbb{M}$ computes classification function $\mathcal{M}(\mathbf{x}) \in \{\boxplus, \boxminus\}$, with $\mathbf{x} \in \mathbb{F}$
- Instance $\mathbf{v} \in \mathbb{F}$, with prediction $c = \mathcal{M}(\mathbf{v})$
  - Prediction literal: $\mathcal{L} \triangleq (\mathcal{M}(\mathbf{v}) = c)$
- Each point $\mathbf{v} \in \mathbb{F}$ is also represented as a set of literals (a cube), $\mathcal{C} = \{(x_i = v_i) | i \in \mathcal{F}\}$

Our approach

Component	Representation	Notes
	$\mathcal{C}$	Conjunction of literals, i.e. cube
	$\mathcal{M}$	Model encoding, e.g. SAT/SMT/CP/ILP/FOL
Cat	$\mathcal{L}$	Predicted class, i.e. literal

Relating with abduction

What we know	$\mathcal{C} \wedge \mathcal{M} \models \mathcal{L}$

Relating with abduction

What we know	$\mathcal{C} \wedge \mathcal{M} \models \mathcal{L}$

Propositional Abduction	Hypotheses	$\mathcal{C}$
	Theory	$\mathcal{M}$
	Manifestation	$\mathcal{L}$
Goal	Find $\mathcal{C}_m \subseteq \mathcal{C}$, s.t.	$\mathcal{C}_m \wedge \mathcal{M} \not\models \bot \wedge \mathcal{C}_m \wedge \mathcal{M} \models \mathcal{L}$

Relating with abduction

What we know	$\mathcal{C} \wedge \mathcal{M} \models \mathcal{L}$

Propositional Abduction	Hypotheses	$\mathcal{C}$
	Theory	$\mathcal{M}$
	Manifestation	$\mathcal{L}$
Goal	Find $\mathcal{C}_m \subseteq \mathcal{C}$, s.t.	$\mathcal{C}_m \wedge \mathcal{M} \not\models \bot \wedge \mathcal{C}_m \wedge \mathcal{M} \models \mathcal{L}$

But,	$\mathcal{C}_m \wedge \mathcal{M} \not\models \bot$ is tautology
And,	$\mathcal{C}_m \wedge \mathcal{M} \models \mathcal{L}$ iff $\mathcal{C}_m \models \mathcal{M} \rightarrow \mathcal{L}$
Thus,	$\mathcal{C}_m$ is **prime implicant** of $\mathcal{M} \rightarrow \mathcal{L}$

Relating with abduction

What we know	$\mathcal{C} \wedge \mathcal{M} \vDash \mathcal{L}$

Propositional Abduction

Hypotheses	$\mathcal{C}$
Theory	$\mathcal{M}$
Manifestation	$\mathcal{L}$

Goal — Find $\mathcal{C}_m \subseteq \mathcal{C}$, s.t. $\mathcal{C}_m \wedge \mathcal{M} \nvDash \bot \wedge \mathcal{C}_m \wedge \mathcal{M} \vDash \mathcal{L}$

But,	$\mathcal{C}_m \wedge \mathcal{M} \nvDash \bot$ is tautology
And,	$\mathcal{C}_m \wedge \mathcal{M} \vDash \mathcal{L}$ iff $\mathcal{C}_m \vDash \mathcal{M} \rightarrow \mathcal{L}$
Thus,	$\mathcal{C}_m$ is **prime implicant** of $\mathcal{M} \rightarrow \mathcal{L}$

We can compute **subset-**/**cardinality-**minimal (prime) implicants

Relating with abduction

What we know	$\mathcal{C} \wedge \mathcal{M} \vDash \mathcal{L}$

Propositional Abduction

Hypotheses

Theory

Manifestation $\mathcal{L}$

Goal: Find $\mathcal{C}_m \subseteq \mathcal{C}$, s.t. $\mathcal{C}_m \wedge \mathcal{M} \nvDash \bot \wedge \mathcal{C}_m \wedge \mathcal{M} \vDash \mathcal{L}$

Obs: For any instance consistent with $\mathcal{C}_m$, and given the model $\mathcal{M}$, the prediction is $\mathcal{L}$!

But,	$\mathcal{C}_m \wedge \mathcal{M} \nvDash \bot$ is tautology
And,	$\mathcal{C}_m \wedge \mathcal{M} \vDash \mathcal{L}$ iff $\mathcal{C}_m \vDash \mathcal{M} \rightarrow \mathcal{L}$
Thus,	$\mathcal{C}_m$ is prime implicant of $\mathcal{M} \rightarrow \mathcal{L}$

We can compute **subset-/cardinality-**minimal (prime) implicants – i.e. explanations!

Computing one subset-minimal explanation

Input: formula $\mathcal{M}$, input cube $\mathcal{C}$, prediction $\mathcal{L}$
Output: ***Subset-minimal*** explanation $\mathcal{C}_m \subseteq \mathcal{C}$

```
begin
    for l ∈ C :
        if Entails(C\{l}, M → L) :
            C ← C\{l}
    return C
end
```

Computing one subset-minimal explanation

Input: formula $\mathcal{M}$, input cube $\mathcal{C}$, prediction $\mathcal{L}$
Output: ***Subset-minimal*** explanation $\mathcal{C}_m \subseteq \mathcal{C}$

```
begin
    for l ∈ C :
        if Entails(C\{l}, M → L) :
            C ← C\{l}
    return C
end
```

Computes some prime

Computing one cardinality-minimal explanation

Input: formula $\mathcal{M}$, input cube $\mathcal{C}$, prediction $\mathcal{L}$
Output: *Cardinality-minimal* explanation $\mathcal{C}_m \subseteq \mathcal{C}$

```
Γ ← ∅
while true do
    C_m ← MinimumHS(Γ)                          // Implicit hitting set dualization
    if Entails(C_m, M → L) :
        return C_m
    else:
        μ ← GetAssignment()
        C_T ← PickFalseLits(C \ C_m, μ)
        Γ ← Γ ∪ C_T
end
```

Computing one cardinality-minimal explanation

Input: formula $\mathcal{M}$, input cube $\mathcal{C}$, prediction $\mathcal{L}$
Output: *Cardinality-minimal* explanation $\mathcal{C}_m \subseteq \mathcal{C}$

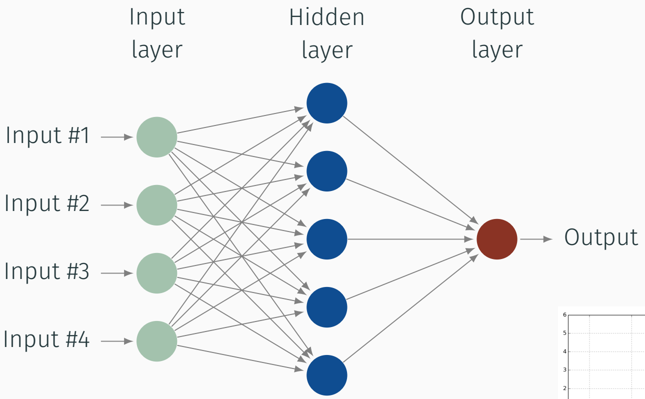
```
Γ ← ∅
while true do
    C_m ← MinimumHS(Γ)                                   // Implicit hitting set dualization
    if Entails(C_m, M → L) :
        return C_m
    else:
        μ ← GetAssignment()
        C_T ← PickFalseLits(C \ C_m, μ)
        Γ ← Γ ∪ C_T
end
```

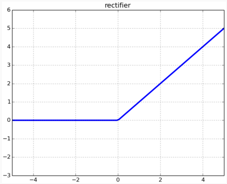
Computes **smallest** prime

In summary

- Target (minimal) **sufficient** conditions for prediction:
  - I.e. we equate explanations with (prime) implicants
- Approach computes set of literals $\mathcal{C}_m \subseteq \mathcal{C}$ such that $\forall(\mathbf{x} \in \mathbb{F}).\mathcal{C}_m(\mathbf{x}) \rightarrow (\mathcal{M}(\mathbf{x}) = \boxplus)$
- **Note:** Equating explanations with prime implicants also proposed in compilation-based approaches [SCD18, SCD19, DH20, Dar20]
  - Referred to as PI-explanations
  - Main difference: compilation vs. use of NP oracles

Recap – 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]

Recap – 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]

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

Sample of experimental results

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

Sample of experimental results

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

Sample of experimental results

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...

Outline

Computing heuristic explanations

- Many (highly visible) heuristic explanation approaches:
  - LIME [RSG16]
  - SHAP [LL17]
  - Anchor [RSG18]
  - ...

Computing heuristic explanations

- Many (highly visible) heuristic explanation approaches:
  - LIME [RSG16]
  - SHAP [LL17]
  - Anchor [RSG18]
  - ...
- **Q:** How to assess the quality of heuristic explanations? [NSM+19, INM19c, Ign20]

Overview of heuristic approaches

- LIME & SHAP: [RSG16, LL17]
  - Goal: learn a simple interpretable ML model, e.g. linear classifier, decision tree, etc.
  - Approach: train classifier vs. game theory
    - LIME is sample-based
    - **Obs 01:** Exact SHAP explanations are as hard as computing the expected value of the model [dBLSS20]
    - **Obs 02:** Exact SHAP explanations are #P-hard for some classes of models [dBLSS20]
- Anchor: [RSG18]
  - Goal: Learn features deemed more relevant for prediction
  - Anchor is sample-based
- No formal guarantees of rigor in computed explanations

A first experiment

[INM19c]

What is the overall quality of heuristic explanations in light of computed heuristic explanations?

Approach

- Learn ML model
  - Focused on **boosted trees** obtained with XGBoost

Approach

- Learn ML model
  - Focused on **boosted trees** obtained with XGBoost
- Compute heuristic explanation for some instance

Approach

- Learn ML model
  - Focused on **boosted trees** obtained with XGBoost
- Compute heuristic explanation for some instance
- Use our abduction-based approach to assess whether heuristic explanation holds globally, i.e. whether it is a PI-explanation, and

Approach

- Learn ML model
  - Focused on boosted trees obtained with XGBoost
- Compute heuristic explanation for some instance
- Use our abduction-based approach to assess whether heuristic explanation holds globally, i.e. whether it is a PI-explanation, and
  1. If it does **not** hold globally, then fix it
     - Explanation is **incorrect**: set of literals is **not** sufficient for prediction!

Approach

- Learn ML model
  - Focused on **boosted trees** obtained with XGBoost
- Compute heuristic explanation for some instance
- Use our abduction-based approach to assess whether heuristic explanation holds globally, i.e. whether it is a PI-explanation, and
  1. If it does **not** hold globally, then fix it
     - Explanation is **incorrect**: set of literals is **not** sufficient for prediction!
  2. If it holds globally but has redundant literals, then **refine** it
     - Explanation is **redundant**: set of literals is sufficient for prediction, but some literals are unnecessary

Approach

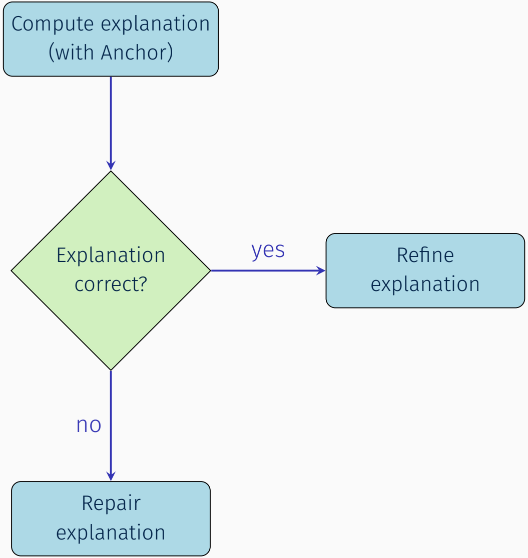
- Learn ML model
  - Focused on **boosted trees** obtained with XGBoost
- Compute heuristic explanation for some instance
- Use our abduction-based approach to assess whether heuristic explanation holds globally, i.e. whether it is a PI-explanation, and
  1. If it does **not** hold globally, then fix it
     - Explanation is **incorrect**: set of literals is **not** sufficient for prediction!
  2. If it holds globally but has redundant literals, then **refine** it
     - Explanation is **redundant**: set of literals is sufficient for prediction, but some literals are unnecessary
  3. Otherwise, report the heuristic explanation as a PI-explanation

Approach

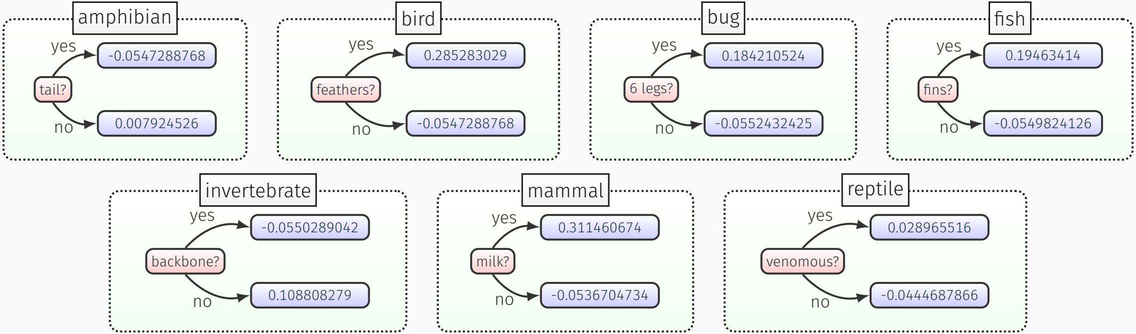
- Learn ML model
  - Focused on **boosted trees** obtained with XGBoost
- Compute heuristic explanation for some instance
- Use our abduction-based approach to assess whether heuristic explanation holds globally, i.e. whether it is a PI-explanation, and
  1. If it does **not** hold globally, then fix it
     - Explanation is **incorrect**: set of literals is **not** sufficient for prediction!
  2. If it holds globally but has redundant literals, then **refine** it
     - Explanation is **redundant**: set of literals is sufficient for prediction, but some literals are unnecessary
  3. Otherwise, report the heuristic explanation as a PI-explanation

Scales to **realistic** size boosted trees...

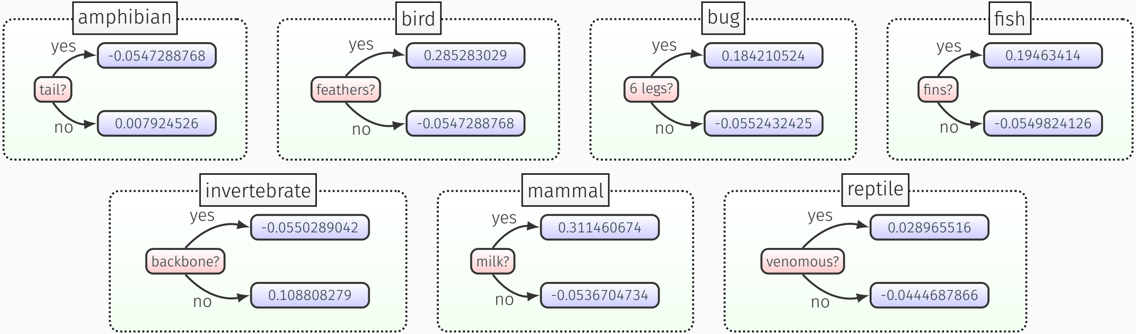
XPlainer – validating, refining & repairing heuristic explanations



An example – zoo dataset



An example – zoo dataset

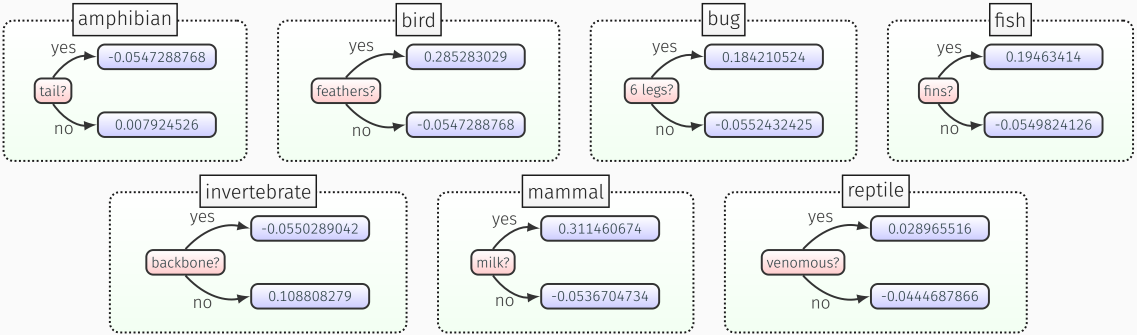


- Example instance:

IF $(\text{animal\_name} = \text{pitviper}) \wedge \neg\text{hair} \wedge \neg\text{feathers} \wedge \text{eggs} \wedge \neg\text{milk} \wedge \neg\text{airborne} \wedge \neg\text{aquatic} \wedge \text{predator} \wedge \neg\text{toothed} \wedge \text{backbone} \wedge \text{breathes} \wedge \text{venomous} \wedge \neg\text{fins} \wedge (\text{legs} = 0) \wedge \text{tail} \wedge \neg\text{domestic} \wedge \neg\text{catsize}$

THEN $(\text{class} = \text{reptile})$

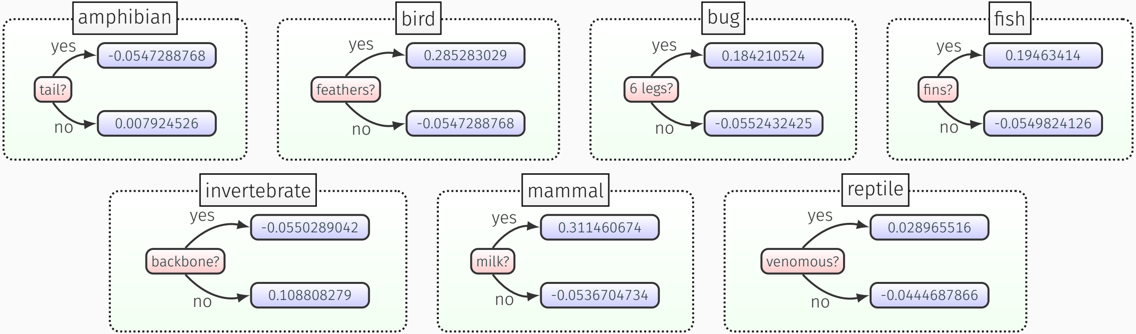
An example – zoo dataset



- Example instance (& Anchor picks):

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 – zoo dataset

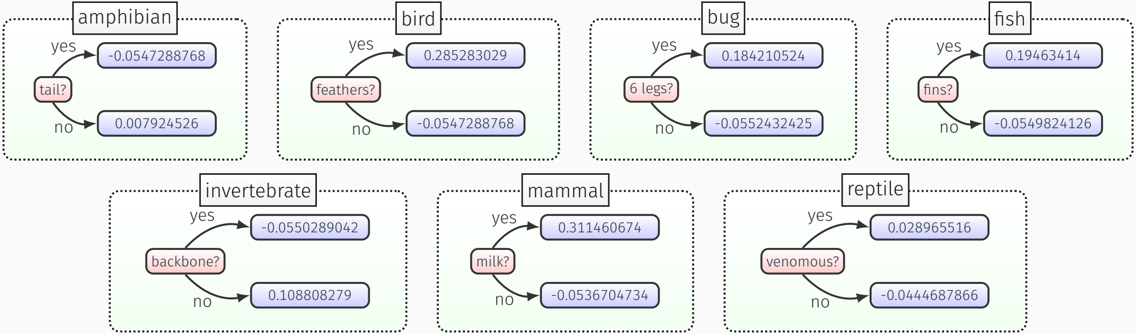


- Explanation obtained with Anchor: [RSG18]

IF $\neg \textit{hair} \wedge \neg \textit{milk} \wedge \neg \textit{toothed} \wedge \neg \textit{fins}$

THEN (class = reptile)

An example – zoo dataset



- **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)

Some results

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%

Some results

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%

& Google XAI service most likely similar...

A second experiment

[NSM+19]

How often are heuristic explanations consistent with prediction?

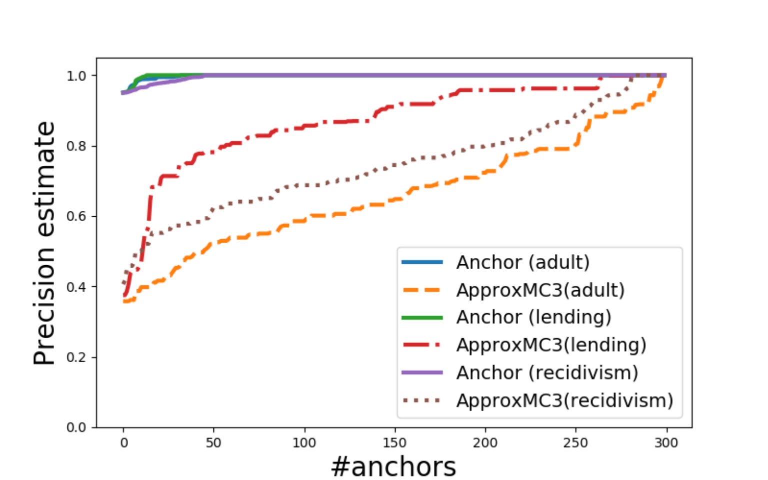
Approach

- Exploit ML model with SAT-based encoding
  - In our case: used binarized neural networks (BNNs)

- Compute heuristic explanations with Anchor (similar results with LIME or SHAP)

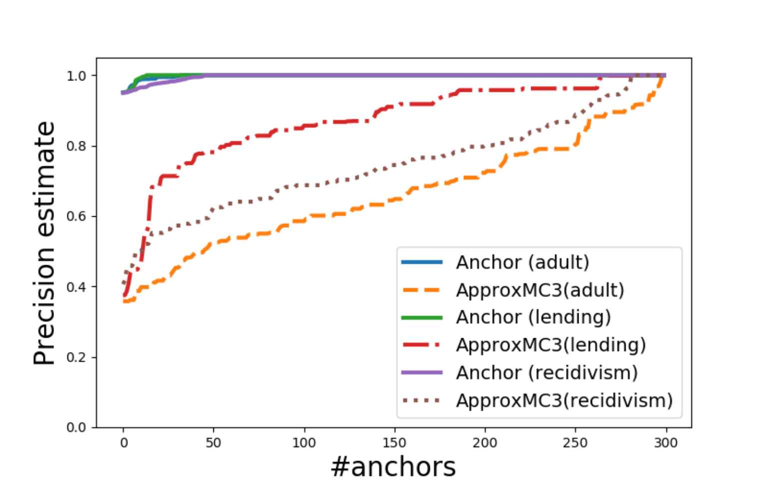
- Use (approximate) model counter to assess how often explanation is consistent with prediction

Preliminary results



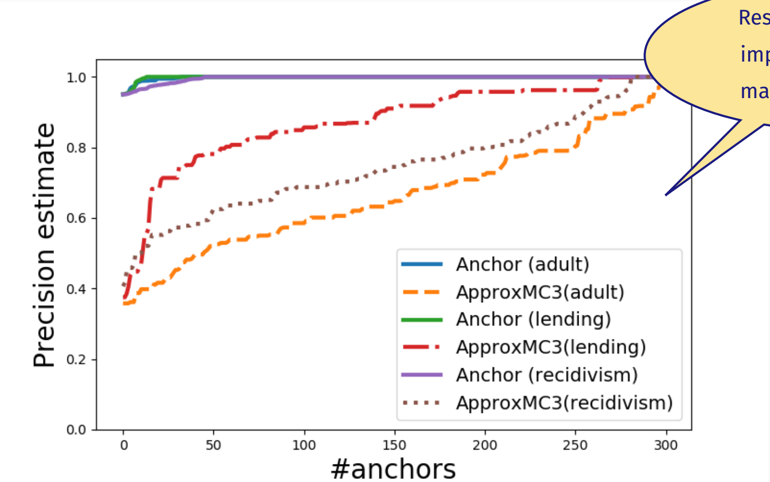
- Anchor often claims $\approx 99\%$ precision

Preliminary results



- Anchor often claims $\approx 99\%$ precision; our results **demonstrate** otherwise

Preliminary results



Results underscore importance of formal explanations !

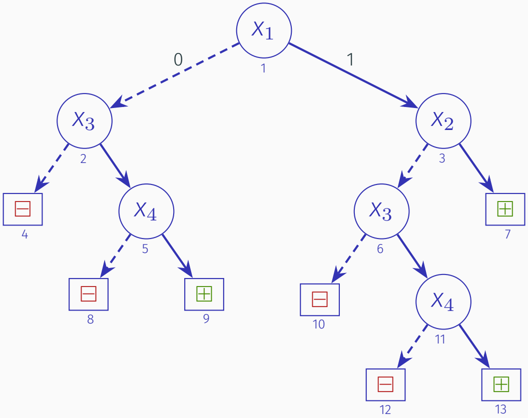
- Anchor often claims $\approx 99\%$ precision; our results **demonstrate** otherwise

Questions on formal vs. heuristic explanations?

Outline

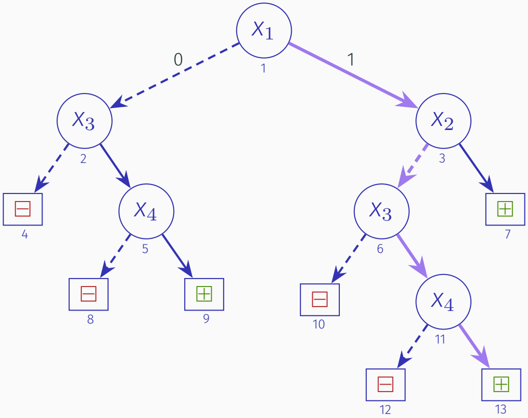
Why PI-explanations for DTs?

[IIM20]



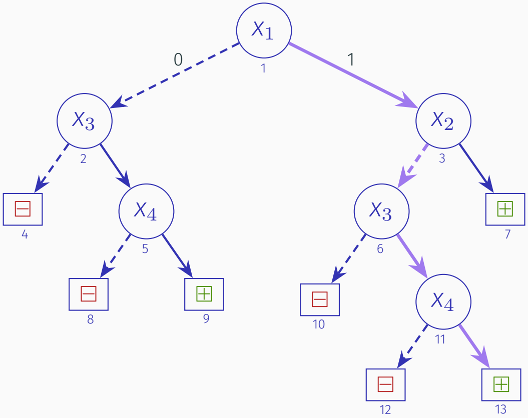
Why PI-explanations for DTs?

[IIM20]



- Instance: $(x_1, x_2, x_3, x_4) = (1, 0, 1, 1)$

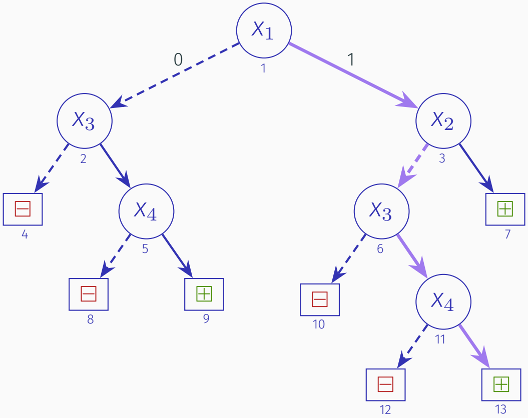
Why PI-explanations for DTs?



[IIM20]

- Instance: $(x_1, x_2, x_3, x_4) = (1, 0, 1, 1)$
- Why is prediction ⊞?
  - PI-explanation for prediction ⊞ given instance $(x_1, x_2, x_3, x_4) = (1, 0, 1, 1)$?

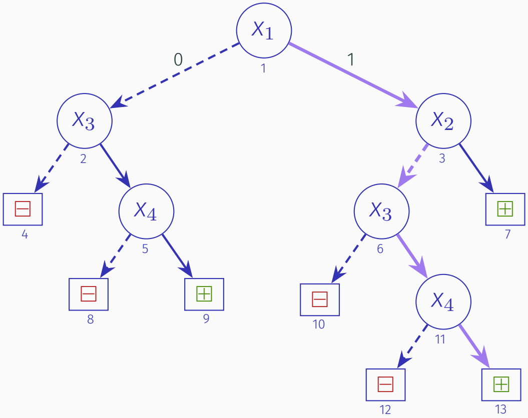
Why PI-explanations for DTs?



[IIM20]

- Instance: $(x_1, x_2, x_3, x_4) = (1, 0, 1, 1)$
- Why is prediction ⊞?
  - PI-explanation for prediction ⊞ given instance $(x_1, x_2, x_3, x_4) = (1, 0, 1, 1)$?
- Analysis:

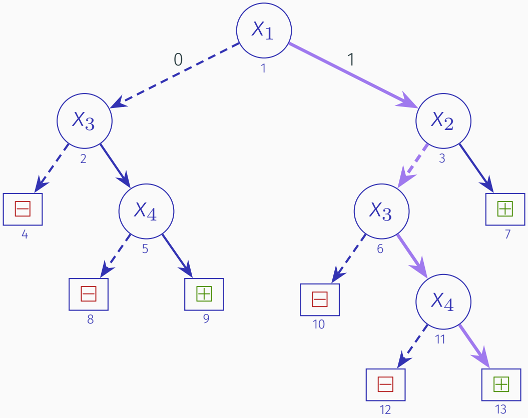
Why PI-explanations for DTs?



[IIM20]

- Instance: $(x_1, x_2, x_3, x_4) = (1, 0, 1, 1)$
- Why is prediction ⊞?
  - PI-explanation for prediction ⊞ given instance $(x_1, x_2, x_3, x_4) = (1, 0, 1, 1)$?
- Analysis:
  - Prediction changes if x_1 can take any value in $\{0, 1\}$?

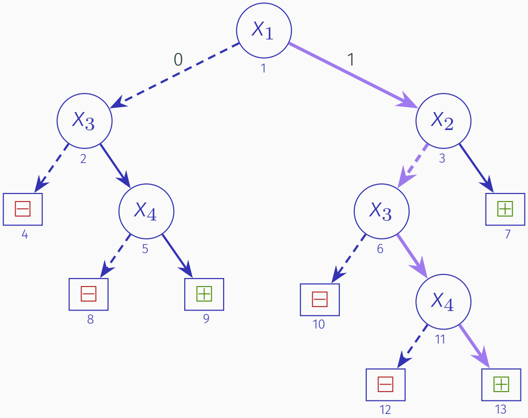
Why PI-explanations for DTs?



[IIM20]

- Instance: $(x_1, x_2, x_3, x_4) = (1, 0, 1, 1)$
- Why is prediction ⊞?
  - PI-explanation for prediction ⊞ given instance $(x_1, x_2, x_3, x_4) = (1, 0, 1, 1)$?
- Analysis:
  - Prediction changes if x_1 can take any value in $\{0, 1\}$? **No**

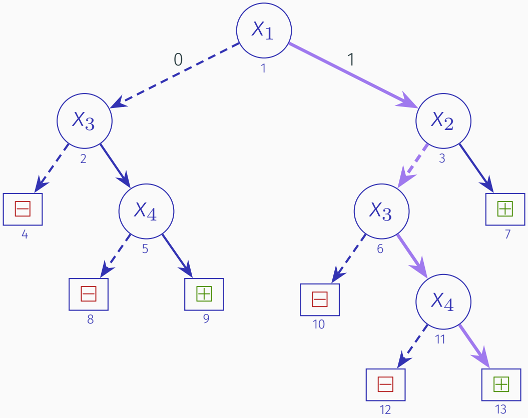
Why PI-explanations for DTs?



[IIM20]

- Instance: $(x_1, x_2, x_3, x_4) = (1, 0, 1, 1)$
- Why is prediction ⊞?
  - PI-explanation for prediction ⊞ given instance $(x_1, x_2, x_3, x_4) = (1, 0, 1, 1)$?
- Analysis:
  - Prediction changes if x_1 can take any value in $\{0, 1\}$? **No**
  - Prediction changes if x_2 and x_1 can take any value in $\{0, 1\}$?

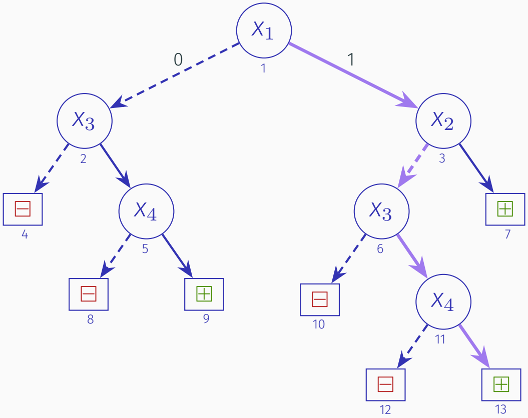
Why PI-explanations for DTs?



[IIM20]

- Instance: $(x_1, x_2, x_3, x_4) = (1, 0, 1, 1)$
- Why is prediction ⊞?
  - PI-explanation for prediction ⊞ given instance $(x_1, x_2, x_3, x_4) = (1, 0, 1, 1)$?
- Analysis:
  - Prediction changes if x_1 can take any value in $\{0, 1\}$? **No**
  - Prediction changes if x_2 and x_1 can take any value in $\{0, 1\}$? **No**

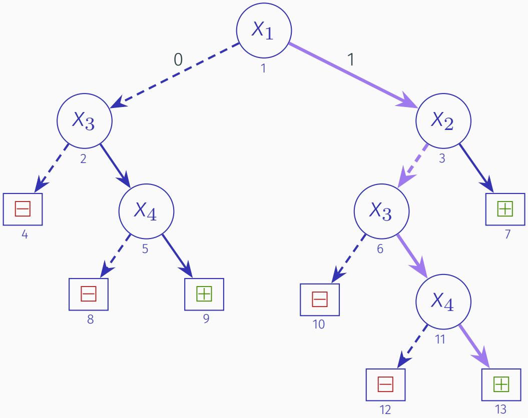
Why PI-explanations for DTs?



[IIM20]

- Instance: $(x_1, x_2, x_3, x_4) = (1, 0, 1, 1)$
- Why is prediction ⊞?
  - PI-explanation for prediction ⊞ given instance $(x_1, x_2, x_3, x_4) = (1, 0, 1, 1)$?
- Analysis:
  - Prediction changes if x_1 can take any value in $\{0, 1\}$? **No**
  - Prediction changes if x_2 and x_1 can take any value in $\{0, 1\}$? **No**
  - PI-explanation: $(x_3 = 1) \wedge (x_4 = 1)$

Why PI-explanations for DTs?

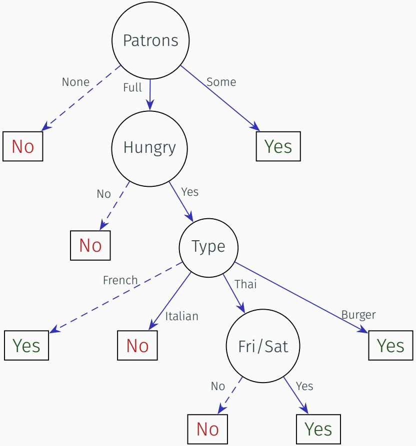


[IIM20]

- Instance: $(x_1, x_2, x_3, x_4) = (1, 0, 1, 1)$
- Why is prediction ⊞?
  - PI-explanation for prediction ⊞ given instance $(x_1, x_2, x_3, x_4) = (1, 0, 1, 1)$?
- Analysis:
  - Prediction changes if x_1 can take any value in $\{0, 1\}$? **No**
  - Prediction changes if x_2 and x_1 can take any value in $\{0, 1\}$? **No**
  - PI-explanation: $(x_3 = 1) \wedge (x_4 = 1)$
  - **Obs:** There are functions for which some paths grows with number of features, and PI-explanation is of constant-size

Need for PI-explanations in DTs is ubiquitous– Russell&Norving's book

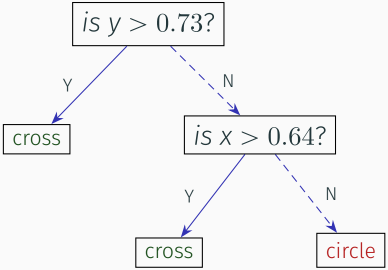
[RN10]



- PI-explanation for $(\mathrm{P}, \mathrm{H}, \mathrm{T}, \mathrm{W}) = (\mathrm{Full}, \mathrm{Yes}, \mathrm{Thai}, \mathrm{No})$?

Need for PI-explanations in DTs is ubiquitous– Zhou's book

[Zho12]

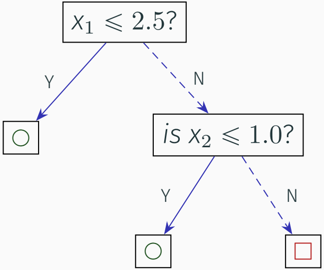


- PI-explanation for $(x, y) = (1.25, -1.13)$?

Obs: PI-explanations can be computed for categorical, ordinal, integer or real-valued features !

Need for PI-explanations in DTs is ubiquitous– Alpaydin's book

[Alp14]

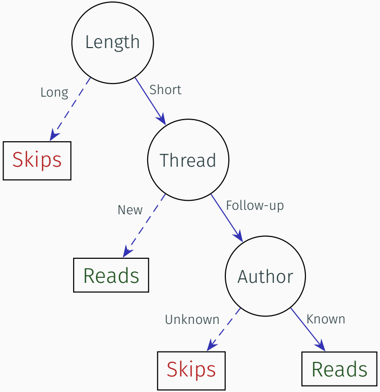


- PI-explanation for $(x_1, x_2) = (3.14, 0.87)$?

Obs: PI-explanations can be computed for categorical, ordinal, integer or real-valued features **!**

Need for PI-explanations in DTs is ubiquitous– Poole&Mackworth's book

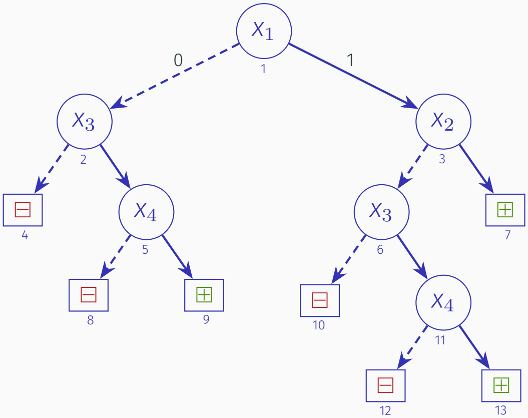
[PM17]



- PI-explanation for $(L, T, A) = (\text{Short}, \text{Follow-Up}, \text{Unknown})$?
- PI-explanation for $(L, T, A) = (\text{Short}, \text{Follow-Up}, \text{Known})$?

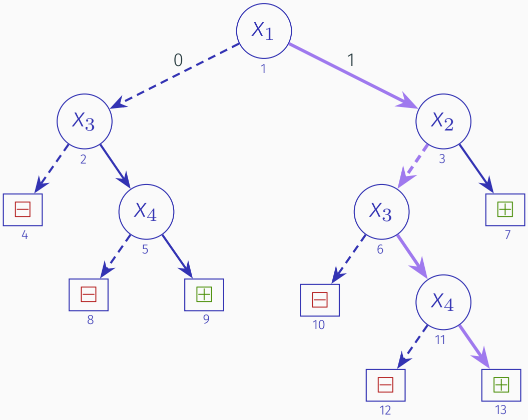
DT explanations

[IIM20]



DT explanations

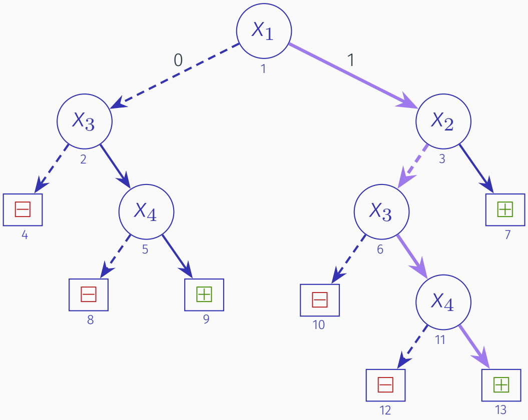
[IIM20]



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

DT explanations

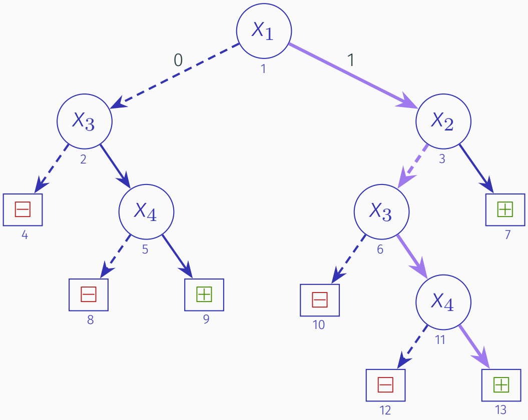
[IIM20]



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

DT explanations in polynomial time

[IIM20]



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

Experimental evidence

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	—	—	—

Questions on explaining DTs?

Background & contribution

Classification problems: $\mathcal{K} = \{\boxplus, \boxminus\}$

Features & feature space: $\mathcal{F} = \{1, \ldots, n\}, \quad \mathbb{F}$

Classifiers: NBCs & LCs

Background & contribution

Classification problems: $\mathcal{K} = \{\boxplus, \boxminus\}$

Features & feature space: $\mathcal{F} = \{1, \ldots, n\}$, $\mathbb{F}$

Classifiers: NBCs & LCs

Goal: PI-explanations [SCD18, INM19a]

Example

$x_1, x_2 \in \{0, 1, 2\}$ Instance: $\mathbf{a} = (2, 0)$, Literals: $(x_1 = 2) \land (x_2 = 0)$

Background & contribution

Classification problems: $\mathcal{K} = \{\boxplus, \boxminus\}$

Features & feature space: $\mathcal{F} = \{1, \ldots, n\}$, $\mathbb{F}$

Classifiers: NBCs & LCs

Goal: PI-explanations [SCD18, INM19a]

Example

$x_1, x_2 \in \{0, 1, 2\}$ Instance: $\mathbf{a} = (2, 0)$, Literals: $(x_1 = 2) \wedge (x_2 = 0)$

Predict $\boxplus$ if: $2x_1 - x_2 > 1$

Predict $\boxminus$ if: $2x_1 - x_2 \leqslant 1$

Background & contribution

Classification problems: $\mathcal{K} = \{\boxplus, \boxminus\}$

Features & feature space: $\mathcal{F} = \{1, \ldots, n\}$, $\mathbb{F}$

Classifiers: NBCs & LCs

Goal: PI-explanations [SCD18, INM19a]

Example

$x_1, x_2 \in \{0, 1, 2\}$ Instance: $\mathbf{a} = (2, 0)$, Literals: $(x_1 = 2) \wedge (x_2 = 0)$

Predict $\boxplus$ if: $2x_1 - x_2 > 1$

Predict $\boxminus$ if: $2x_1 - x_2 \leqslant 1$

Prediction w/ $\mathbf{a} = (2, 0)$: $\boxplus$

Background & contribution

Classification problems: $\mathcal{K} = \{\boxplus, \boxminus\}$

Features & feature space: $\mathcal{F} = \{1, \ldots, n\}$, $\mathbb{F}$

Classifiers: NBCs & LCs

Goal: PI-explanations [SCD18, INM19a]

Example

$x_1, x_2 \in \{0, 1, 2\}$ Instance: $\mathbf{a} = (2, 0)$, Literals: $(x_1 = 2) \wedge (x_2 = 0)$

Predict $\boxplus$ if: $2x_1 - x_2 > 1$

Predict $\boxminus$ if: $2x_1 - x_2 \leqslant 1$

Prediction w/ $\mathbf{a} = (2, 0)$: $\boxplus$

PI-explanation: $\{(x_1 = 2)\}$, i.e. $(x_2 = 0)$ is **irrelevant** for prediction

Background & contribution

Classification problems: $\mathcal{K} = \{\boxplus, \boxminus\}$

Features & feature space: $\mathcal{F} = \{1, \ldots, n\}$, $\mathbb{F}$

Classifiers: NBCs & LCs

Goal: PI-explanations [SCD18, INM19a]

Example

$x_1, x_2 \in \{0, 1, 2\}$ Instance: $\mathbf{a} = (2, 0)$, Literals: $(x_1 = 2) \wedge (x_2 = 0)$

Predict $\boxplus$ if: $2x_1 - x_2 > 1$

Predict $\boxminus$ if: $2x_1 - x_2 \leqslant 1$

Prediction w/ $\mathbf{a} = (2, 0)$: $\boxplus$

By default we consider class $\boxplus$

PI-explanation: $\{(x_1 = 2)\}$, i.e. $(x_2 = 0)$ is **irrelevant** for prediction

Recap PI-explanation: **minimal** set of literals **sufficient** for prediction

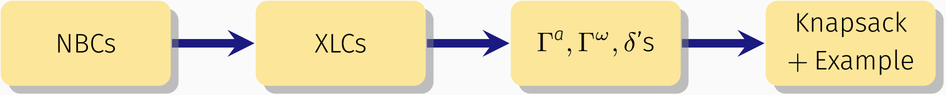
Background & contribution – outline

Classification problems: $\mathcal{K} = \{\boxplus, \boxminus\}$

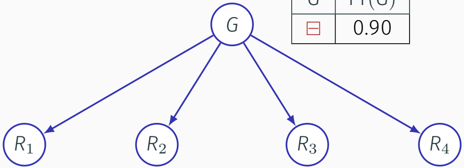
Features & feature space: $\mathcal{F} = \{1, \ldots, n\}$, $\mathbb{F}$

Classifiers: NBCs & LCs

Goal: PI-explanations [SCD18, INM19a]



G	$\Pr(G)$
⊟	0.90



G	$\Pr(R_1 \mid G)$
⊞	0.95
⊟	0.03

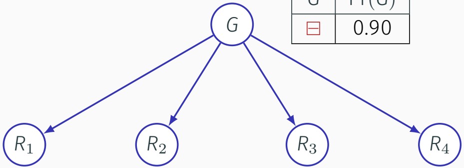
G	$\Pr(R_2 \mid G)$
⊞	0.05
⊟	0.95

G	$\Pr(R_3 \mid G)$
⊞	0.02
⊟	0.34

G	$\Pr(R_4 \mid G)$
⊞	0.20
⊟	0.75

NBC classifier (def): $\tau(\mathbf{e}) = \text{argmax}_{c \in \mathcal{K}}(\Pr(c \mid \mathbf{e}))$

G	$\Pr(G)$
$\boxminus$	0.90



G	$\Pr(R_1 \mid G)$
$\boxplus$	0.95
$\boxminus$	0.03

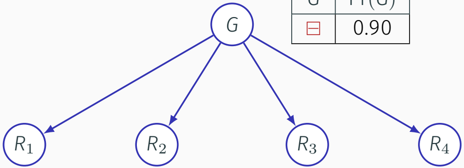
G	$\Pr(R_2 \mid G)$
$\boxplus$	0.05
$\boxminus$	0.95

G	$\Pr(R_3 \mid G)$
$\boxplus$	0.02
$\boxminus$	0.34

G	$\Pr(R_4 \mid G)$
$\boxplus$	0.20
$\boxminus$	0.75

NBC classifier (def): $\tau(\mathbf{e}) = \text{argmax}_{c \in \mathcal{K}}(\Pr(c|\mathbf{e})) = \text{argmax}_{c \in \mathcal{K}} \left(\Pr(c) \times \prod_i \Pr(e_i|c)\right)$

Key concepts & outcomes – NBCs & lPr



G	$\Pr(G)$
$\boxminus$	0.90

G	$\Pr(R_1 \mid G)$
$\boxplus$	0.95
$\boxminus$	0.03

G	$\Pr(R_2 \mid G)$
$\boxplus$	0.05
$\boxminus$	0.95

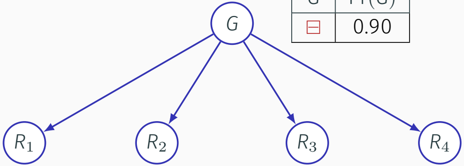
G	$\Pr(R_3 \mid G)$
$\boxplus$	0.02
$\boxminus$	0.34

G	$\Pr(R_4 \mid G)$
$\boxplus$	0.20
$\boxminus$	0.75

NBC classifier (def): $\tau(\mathbf{e}) = \text{argmax}_{c\in\mathcal{K}}(\Pr(c|\mathbf{e})) = \text{argmax}_{c\in\mathcal{K}}\left(\Pr(c) \times \prod_i \Pr(e_i|c)\right)$

NBC classifier (alt): $\tau(\mathbf{e}) = \text{argmax}_{c\in\mathcal{K}}\left((\mathbb{T} + \log \Pr(c)) + \sum_i (\mathbb{T} + \log \Pr(e_i|c))\right)$

Key concepts & outcomes – NBCs & lPr



G	$\Pr(G)$
$\boxminus$	0.90

G	$\Pr(R_1 \vert G)$
$\boxplus$	0.95
$\boxminus$	0.03

G	$\Pr(R_2 \vert G)$
$\boxplus$	0.05
$\boxminus$	0.95

G	$\Pr(R_3 \vert G)$
$\boxplus$	0.02
$\boxminus$	0.34

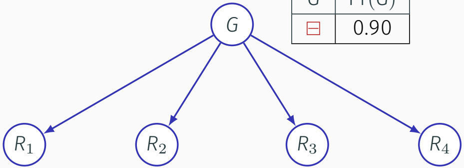
G	$\Pr(R_4 \vert G)$
$\boxplus$	0.20
$\boxminus$	0.75

NBC classifier (def): $\tau(\mathbf{e}) = \text{argmax}_{c \in \mathcal{K}}(\Pr(c|\mathbf{e})) = \text{argmax}_{c \in \mathcal{K}}\left(\Pr(c) \times \prod_i \Pr(e_i|c)\right)$

NBC classifier (alt): $\tau(\mathbf{e}) = \text{argmax}_{c \in \mathcal{K}}\left((\mathbb{T} + \log \Pr(c)) + \sum_i(\mathbb{T} + \log \Pr(e_i|c))\right)$

Using oper. lPr$(\cdot)$: $\tau(\mathbf{e}) = \text{argmax}_{c \in \mathcal{K}}(\text{lPr}(c|\mathbf{e})) = \text{argmax}_{c \in \mathcal{K}}\left((\text{lPr}(c)) + \sum_i(\text{lPr}(e_i|c))\right)$

Key concepts & outcomes – working with lPr



G	$\Pr(G)$
$\boxminus$	0.90

G	$\Pr(R_1 \mid G)$
$\boxplus$	0.95
$\boxminus$	0.03

G	$\Pr(R_2 \mid G)$
$\boxplus$	0.05
$\boxminus$	0.95

G	$\Pr(R_3 \mid G)$
$\boxplus$	0.02
$\boxminus$	0.34

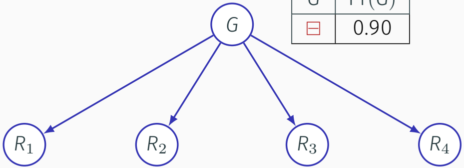
G	$\Pr(R_4 \mid G)$
$\boxplus$	0.20
$\boxminus$	0.75

$\mathbf{a} = (1,0,1,0)$	$\Pr(\boxplus)$	$\Pr(r_1 \mid \boxplus)$	$\Pr(\neg r_2 \mid \boxplus)$	$\Pr(r_3 \mid \boxplus)$	$\Pr(\neg r_4 \mid \boxplus)$	$\mathrm{lPr}(\boxplus \mid \mathbf{a})$
$\Pr(\cdot)$	0.10	0.95	0.95	0.02	0.80	
$\mathrm{lPr}(\cdot)$	1.70	3.95	3.95	0.09	3.78	13.47

$\mathbf{a} = (1,0,1,0)$	$\Pr(\boxminus)$	$\Pr(r_1 \mid \boxminus)$	$\Pr(\neg r_2 \mid \boxminus)$	$\Pr(r_3 \mid \boxminus)$	$\Pr(\neg r_4 \mid \boxminus)$	$\mathrm{lPr}(\boxminus \mid \mathbf{a})$
$\Pr(\cdot)$	0.90	0.03	0.05	0.34	0.25	
$\mathrm{lPr}(\cdot)$	3.89	0.49	1.00	2.92	2.61	10.91

Key concepts & outcomes – working with lPr

G	$\Pr(G)$
$\boxminus$	0.90



G	$\Pr(R_1 \mid G)$
$\boxplus$	0.95
$\boxminus$	0.03

G	$\Pr(R_2 \mid G)$
$\boxplus$	0.05
$\boxminus$	0.95

G	$\Pr(R_3 \mid G)$
$\boxplus$	0.02
$\boxminus$	0.34

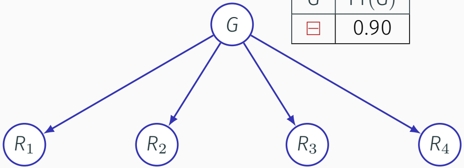
G	$\Pr(R_4 \mid G)$
$\boxplus$	0.20
$\boxminus$	0.75

Pick class $\boxplus$!

$\mathbf{a} = (1, 0, 1, 0)$	$\Pr(\boxplus)$	$\Pr(r_1 \mid \boxplus)$	$\Pr(\neg r_2 \mid \boxplus)$	$\Pr(r_3 \mid \boxplus)$	$\Pr(\neg r_4 \mid \boxplus)$	$\text{lPr}(\boxplus \mid \mathbf{a})$
$\Pr(\cdot)$	0.10	0.95	0.95	0.02	0.80	
$\text{lPr}(\cdot)$	1.70	3.95	3.95	0.09	3.78	13.47

$\mathbf{a} = (1, 0, 1, 0)$	$\Pr(\boxminus)$	$\Pr(r_1 \mid \boxminus)$	$\Pr(\neg r_2 \mid \boxminus)$	$\Pr(r_3 \mid \boxminus)$	$\Pr(\neg r_4 \mid \boxminus)$	$\text{lPr}(\boxminus \mid \mathbf{a})$
$\Pr(\cdot)$	0.90	0.03	0.05	0.34	0.25	
$\text{lPr}(\cdot)$	3.89	0.49	1.00	2.92	2.61	10.91

Key concepts & outcomes – XLCs



G	$\Pr(G)$
$\boxminus$	0.90

G	$\Pr(R_1\vert G)$
$\boxplus$	0.95
$\boxminus$	0.03

G	$\Pr(R_2\vert G)$
$\boxplus$	0.05
$\boxminus$	0.95

G	$\Pr(R_3\vert G)$
$\boxplus$	0.02
$\boxminus$	0.34

G	$\Pr(R_4\vert G)$
$\boxplus$	0.20
$\boxminus$	0.75

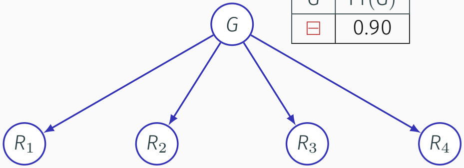
NBC classifier (def): $\tau(\mathbf{e}) = \text{argmax}_{c \in \mathcal{K}} \left(\Pr(c) \times \prod_i \Pr(e_i|c)\right)$

NBC classifier (alt): $\tau(\mathbf{e}) = \text{argmax}_{c \in \mathcal{K}} \left((\mathbb{T} + \log \Pr(c)) + \sum_i (\mathbb{T} + \log \Pr(e_i|c))\right)$

Using oper. $\text{lPr}(\cdot)$: $\tau(\mathbf{e}) = \text{argmax}_{c \in \mathcal{K}} \left((\text{lPr}(c)) + \sum_i (\text{lPr}(e_i|c))\right)$

XLC classifier: $\nu(\mathbf{e}) \triangleq w_0 + \sum_{i \in \mathcal{R}} w_i e_i + \sum_{j \in \mathcal{C}} \sigma(e_j, v_j^1, v_j^2, \ldots, v_j^{d_j})$

Key concepts & outcomes – XLCs



G	$\Pr(G)$
$\boxminus$	0.90

G	$\Pr(R_1 \mid G)$
$\boxplus$	0.95
$\boxminus$	0.03

G	$\Pr(R_2 \mid G)$
$\boxplus$	0.05
$\boxminus$	0.95

G	$\Pr(R_3 \mid G)$
$\boxplus$	0.02
$\boxminus$	0.34

G	$\Pr(R_4 \mid G)$
$\boxplus$	0.20
$\boxminus$	0.75

Can reduce NBC to XLC

NBC classifier (def): $\tau(\mathbf{e}) = \text{argmax}_{c \in \mathcal{K}} \left(\Pr(c) \times \prod_i \Pr(e_i|c)\right)$

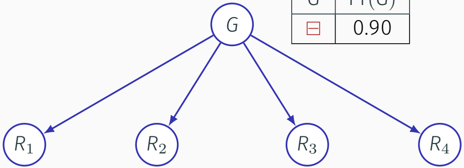
NBC classifier (alt): $\tau(\mathbf{e}) = \text{argmax}_{c \in \mathcal{K}} \left((\mathbb{T} + \log \Pr(c)) + \sum_i (\mathbb{T} + \log \Pr(e_i|c))\right)$

Using oper. $\text{lPr}(\cdot)$: $\tau(\mathbf{e}) = \text{argmax}_{c \in \mathcal{K}} \left((\text{lPr}(c)) + \sum_i (\text{lPr}(e_i|c))\right)$

XLC classifier: $\nu(\mathbf{e}) \triangleq w_0 + \sum_{i \in \mathcal{R}} w_i e_i + \sum_{j \in \mathcal{C}} \sigma(e_j, v_j^1, v_j^2, \ldots, v_j^{d_j})$

Key concepts & outcomes – NBC to XLC

G	$\Pr(G)$
$\boxminus$	0.90



G	$\Pr(R_1 \mid G)$
$\boxplus$	0.95
$\boxminus$	0.03

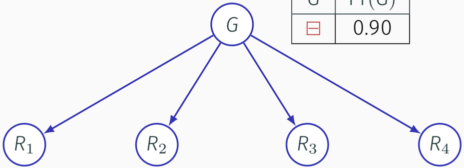
G	$\Pr(R_2 \mid G)$
$\boxplus$	0.05
$\boxminus$	0.95

G	$\Pr(R_3 \mid G)$
$\boxplus$	0.02
$\boxminus$	0.34

G	$\Pr(R_4 \mid G)$
$\boxplus$	0.20
$\boxminus$	0.75

Eliminate argmax:

$$\mathrm{lPr}(\boxplus) - \mathrm{lPr}(\boxminus) + \sum_{i=1}^{n} (\mathrm{lPr}(\neg e_i \mid \boxplus) - \mathrm{lPr}(\neg e_i \mid \boxminus)) \neg e_i + \sum_{i=1}^{n} (\mathrm{lPr}(e_i \mid \boxplus) - \mathrm{lPr}(e_i \mid \boxminus)) e_i > \mathbf{0}$$



G	$\Pr(G)$
$\boxminus$	0.90

G	$\Pr(R_1\|G)$
$\boxplus$	0.95
$\boxminus$	0.03

G	$\Pr(R_2\|G)$
$\boxplus$	0.05
$\boxminus$	0.95

G	$\Pr(R_3\|G)$
$\boxplus$	0.02
$\boxminus$	0.34

G	$\Pr(R_4\|G)$
$\boxplus$	0.20
$\boxminus$	0.75

Eliminate argmax:

$$\mathrm{lPr}(\boxplus) - \mathrm{lPr}(\boxminus) + \sum_{i=1}^{n}(\mathrm{lPr}(\neg e_i|\boxplus) - \mathrm{lPr}(\neg e_i|\boxminus))\neg e_i + \sum_{i=1}^{n}(\mathrm{lPr}(e_i|\boxplus) - \mathrm{lPr}(e_i|\boxminus))e_i > \mathbf{0}$$

Mapping to XLC:

$$w_0 \triangleq \mathrm{lPr}(\boxplus) - \mathrm{lPr}(\boxminus)$$

$$v_j^1 \triangleq \mathrm{lPr}(\neg e_j|\boxplus) - \mathrm{lPr}(\neg e_j|\boxminus)$$

$$v_j^2 \triangleq \mathrm{lPr}(e_j|\boxplus) - \mathrm{lPr}(e_j|\boxminus)$$

Key concepts & outcomes – example reduction

	$\Pr(\boxplus)$	$\Pr(\neg r_1 \mid \boxplus)$	$\Pr(r_1 \mid \boxplus)$	$\Pr(\neg r_2 \mid \boxplus)$	$\Pr(r_2 \mid \boxplus)$	$\Pr(\neg r_3 \mid \boxplus)$	$\Pr(r_3 \mid \boxplus)$	$\Pr(\neg r_4 \mid \boxplus)$	$\Pr(r_4 \mid \boxplus)$
$\Pr(\cdot)$	0.10	0.05	0.95	0.95	0.05	0.98	0.02	0.80	0.20
$\mathrm{lPr}(\cdot)$	1.70	1.00	3.95	3.95	1.00	3.98	0.09	3.78	2.39

	$\Pr(\boxminus)$	$\Pr(\neg r_1 \mid \boxminus)$	$\Pr(r_1 \mid \boxminus)$	$\Pr(\neg r_2 \mid \boxminus)$	$\Pr(r_2 \mid \boxminus)$	$\Pr(\neg r_3 \mid \boxminus)$	$\Pr(r_3 \mid \boxminus)$	$\Pr(\neg r_4 \mid \boxminus)$	$\Pr(r_4 \mid \boxminus)$
$\Pr(\cdot)$	0.90	0.97	0.03	0.05	0.95	0.66	0.34	0.25	0.75
$\mathrm{lPr}(\cdot)$	3.89	3.97	0.49	1.00	3.95	3.58	2.92	2.61	3.71

w_0	v_1^1	v_1^2	v_2^1	v_2^2	v_3^1	v_3^2	v_4^1	v_4^2
-2.19	-2.97	3.46	2.95	-2.95	0.4	-2.83	1.17	-1.32

Key concepts & outcomes – minding the gap

	$\Pr(\boxplus)$	$\Pr(\neg r_1 \mid \boxplus)$	$\Pr(r_1 \mid \boxplus)$	$\Pr(\neg r_2 \mid \boxplus)$	$\Pr(r_2 \mid \boxplus)$	$\Pr(\neg r_3 \mid \boxplus)$	$\Pr(r_3 \mid \boxplus)$	$\Pr(\neg r_4 \mid \boxplus)$	$\Pr(r_4 \mid \boxplus)$
$\Pr(\cdot)$	0.10	0.05	0.95	0.95	0.05	0.98	0.02	0.80	0.20
$l\Pr(\cdot)$	1.70	1.00	3.95	3.95	1.00	3.98	0.09	3.78	2.39

	$\Pr(\boxminus)$	$\Pr(\neg r_1 \mid \boxminus)$	$\Pr(r_1 \mid \boxminus)$	$\Pr(\neg r_2 \mid \boxminus)$	$\Pr(r_2 \mid \boxminus)$	$\Pr(\neg r_3 \mid \boxminus)$	$\Pr(r_3 \mid \boxminus)$	$\Pr(\neg r_4 \mid \boxminus)$	$\Pr(r_4 \mid \boxminus)$
$\Pr(\cdot)$	0.90	0.97	0.03	0.05	0.95	0.66	0.34	0.25	0.75
$l\Pr(\cdot)$	3.89	3.97	0.49	1.00	3.95	3.58	2.92	2.61	3.71

Gap value: $\Gamma^a \triangleq \nu(\mathbf{a}) = w_0 + \sum_{j \in \mathcal{C}} \sigma(a_j, v_j^1, v_j^2, \ldots, v_i^{d_j}) > \mathbf{0}$

Key concepts & outcomes – minding the gap

	$\Pr(\boxplus)$	$\Pr(\neg r_1 \mid \boxplus)$	$\Pr(r_1 \mid \boxplus)$	$\Pr(\neg r_2 \mid \boxplus)$	$\Pr(r_2 \mid \boxplus)$	$\Pr(\neg r_3 \mid \boxplus)$	$\Pr(r_3 \mid \boxplus)$	$\Pr(\neg r_4 \mid \boxplus)$	$\Pr(r_4 \mid \boxplus)$
$\Pr(\cdot)$	0.10	0.05	0.95	0.95	0.05	0.98	0.02	0.80	0.20
$l\Pr(\cdot)$	1.70	1.00	3.95	3.95	1.00	3.98	0.09	3.78	2.39

	$\Pr(\boxminus)$	$\Pr(\neg r_1 \mid \boxminus)$	$\Pr(r_1 \mid \boxminus)$	$\Pr(\neg r_2 \mid \boxminus)$	$\Pr(r_2 \mid \boxminus)$	$\Pr(\neg r_3 \mid \boxminus)$	$\Pr(r_3 \mid \boxminus)$	$\Pr(\neg r_4 \mid \boxminus)$	$\Pr(r_4 \mid \boxminus)$
$\Pr(\cdot)$	0.90	0.97	0.03	0.05	0.95	0.66	0.34	0.25	0.75
$l\Pr(\cdot)$	3.89	3.97	0.49	1.00	3.95	3.58	2.92	2.61	3.71

Gap value: $\Gamma^a \triangleq \nu(\mathbf{a}) = w_0 + \sum_{j \in \mathcal{C}} \sigma(a_j, v_j^1, v_j^2, \ldots, v_i^{d_j}) > \mathbf{0}$

Worst-case gap: $\Gamma^\omega \triangleq w_0 + \sum_{j \in \mathcal{C}} v_j^\omega < \mathbf{0}$

Key concepts & outcomes – minding the gap

	$\Pr(\boxplus)$	$\Pr(\neg r_1 \vert \boxplus)$	$\Pr(r_1 \vert \boxplus)$	$\Pr(\neg r_2 \vert \boxplus)$	$\Pr(r_2 \vert \boxplus)$	$\Pr(\neg r_3 \vert \boxplus)$	$\Pr(r_3 \vert \boxplus)$	$\Pr(\neg r_4 \vert \boxplus)$	$\Pr(r_4 \vert \boxplus)$
$\Pr(\cdot)$	0.10	0.05	0.95	0.95	0.05	0.98	0.02	0.80	0.20
$\text{lPr}(\cdot)$	1.70	1.00	3.95	3.95	1.00	3.98	0.09	3.78	2.39

	$\Pr(\boxminus)$	$\Pr(\neg r_1 \vert \boxminus)$	$\Pr(r_1 \vert \boxminus)$	$\Pr(\neg r_2 \vert \boxminus)$	$\Pr(r_2 \vert \boxminus)$	$\Pr(\neg r_3 \vert \boxminus)$	$\Pr(r_3 \vert \boxminus)$	$\Pr(\neg r_4 \vert \boxminus)$	$\Pr(r_4 \vert \boxminus)$
$\Pr(\cdot)$	0.90	0.97	0.03	0.05	0.95	0.66	0.34	0.25	0.75
$\text{lPr}(\cdot)$	3.89	3.97	0.49	1.00	3.95	3.58	2.92	2.61	3.71

Gap value: $\Gamma^a \triangleq \nu(\mathbf{a}) = w_0 + \sum_{j\in\mathcal{C}} \sigma(a_j, v_j^1, v_j^2, \ldots, v_j^{d_j}) > \mathbf{0}$

Worst-case gap: $\Gamma^\omega \triangleq w_0 + \sum_{j\in\mathcal{C}} v_j^\omega < \mathbf{0}$

Relate Γ^a and Γ^ω: $\Gamma^\omega = w_0 + \sum_{j\in\mathcal{C}} v_j^{a_j} - \sum_{j\in\mathcal{C}}(v_j^{a_j} - v_j^\omega) = \Gamma^a - \sum_{j\in\mathcal{C}} \delta_j = -\Phi$

where, $\delta_j \triangleq v_j^{a_j} - v_j^\omega = v_j^{a_j} - \min\{v_j^1, v_j^2, \ldots\}$

Key concepts & outcomes – minding the gap

	$\Pr(\boxplus)$	$\Pr(\neg r_1 \mid \boxplus)$	$\Pr(r_1 \mid \boxplus)$	$\Pr(\neg r_2 \mid \boxplus)$	$\Pr(r_2 \mid \boxplus)$	$\Pr(\neg r_3 \mid \boxplus)$	$\Pr(r_3 \mid \boxplus)$	$\Pr(\neg r_4 \mid \boxplus)$	$\Pr(r_4 \mid \boxplus)$
$\Pr(\cdot)$	0.10	0.05	0.95	0.95	0.05	0.98	0.02	0.80	0.20
$\text{lPr}(\cdot)$	1.70	1.00	3.95	3.95	1.00	3.98	0.09	3.78	2.39

	$\Pr(\boxminus)$	$\Pr(\neg r_1 \mid \boxminus)$	$\Pr(r_1 \mid \boxminus)$	$\Pr(\neg r_2 \mid \boxminus)$	$\Pr(r_2 \mid \boxminus)$	$\Pr(\neg r_3 \mid \boxminus)$	$\Pr(r_3 \mid \boxminus)$	$\Pr(\neg r_4 \mid \boxminus)$	$\Pr(r_4 \mid \boxminus)$
$\Pr(\cdot)$	0.90	0.97	0.03	0.05	0.95	0.66	0.34	0.25	0.75
$\text{lPr}(\cdot)$	3.89	3.97	0.49	1.00	3.95	3.58	2.92	2.61	3.71

Gap value: $\Gamma^a \triangleq \nu(\mathbf{a}) = w_0 + \sum_{j\in\mathcal{C}} \sigma(a_j, v_j^1, v_j^2, \ldots, v_j^{d_j}) > \mathbf{0}$

Worst-case gap: $\Gamma^\omega \triangleq w_0 + \sum_{j\in\mathcal{C}} v_j^\omega < \mathbf{0}$

Relate Γ^a and Γ^ω: $\Gamma^\omega = w_0 + \sum_{j\in\mathcal{C}} v_j^{a_j} - \sum_{j\in\mathcal{C}} (v_j^{a_j} - v_j^\omega) = \Gamma^a - \sum_{j\in\mathcal{C}} \delta_j = -\Phi$

where, $\delta_j \triangleq v_j^{a_j} - v_j^\omega = v_j^{a_j} - \min\{v_j^1, v_j^2, \ldots\}$

Worst-case, given some min. $\mathcal{P}$: $w_0 + \sum_{j\in\mathcal{P}} v_j^{a_j} + \sum_{j\notin\mathcal{P}} v_j^\omega = -\Phi + \sum_{j\in\mathcal{P}} \delta_j > 0$

Key concepts & outcomes – computing δ's

	$Pr(\boxplus)$	$Pr(\neg r_1 \mid \boxplus)$	$Pr(r_1 \mid \boxplus)$	$Pr(\neg r_2 \mid \boxplus)$	$Pr(r_2 \mid \boxplus)$	$Pr(\neg r_3 \mid \boxplus)$	$Pr(r_3 \mid \boxplus)$	$Pr(\neg r_4 \mid \boxplus)$	$Pr(r_4 \mid \boxplus)$
$Pr(\cdot)$	0.10	0.05	0.95	0.95	0.05	0.98	0.02	0.80	0.20
$lPr(\cdot)$	1.70	1.00	3.95	3.95	1.00	3.98	0.09	3.78	2.39

	$Pr(\boxminus)$	$Pr(\neg r_1 \mid \boxminus)$	$Pr(r_1 \mid \boxminus)$	$Pr(\neg r_2 \mid \boxminus)$	$Pr(r_2 \mid \boxminus)$	$Pr(\neg r_3 \mid \boxminus)$	$Pr(r_3 \mid \boxminus)$	$Pr(\neg r_4 \mid \boxminus)$	$Pr(r_4 \mid \boxminus)$
$Pr(\cdot)$	0.90	0.97	0.03	0.05	0.95	0.66	0.34	0.25	0.75
$lPr(\cdot)$	3.89	3.97	0.49	1.00	3.95	3.58	2.92	2.61	3.71

w_0	v_1^1	v_1^2	v_2^1	v_2^2	v_3^1	v_3^2	v_4^1	v_4^2
-2.19	-2.97	3.46	2.95	-2.95	0.4	-2.83	1.17	-1.32

Γ^a	δ_1	δ_2	δ_3	δ_4	$\Phi = -\Gamma^\omega$
2.56	6.43	5.90	0	2.49	12.26

Key concepts & outcomes – computing δ's

	$\Pr(\boxplus)$	$\Pr(\neg r_1 \mid \boxplus)$	$\Pr(r_1 \mid \boxplus)$	$\Pr(\neg r_2 \mid \boxplus)$	$\Pr(r_2 \mid \boxplus)$	$\Pr(\neg r_3 \mid \boxplus)$	$\Pr(r_3 \mid \boxplus)$	$\Pr(\neg r_4 \mid \boxplus)$	$\Pr(r_4 \mid \boxplus)$
$\Pr(\cdot)$	0.10	0.05	0.95	0.95	0.05	0.98	0.02	0.80	0.20
$\mathrm{lPr}(\cdot)$	1.70	1.00	3.95	3.95	1.00	3.98	0.09	3.78	2.39

	$\Pr(\boxminus)$	$\Pr(\neg r_1 \mid \boxminus)$	$\Pr(r_1 \mid \boxminus)$	$\Pr(\neg r_2 \mid \boxminus)$	$\Pr(r_2 \mid \boxminus)$	$\Pr(\neg r_3 \mid \boxminus)$	$\Pr(r_3 \mid \boxminus)$	$\Pr(\neg r_4 \mid \boxminus)$	$\Pr(r_4 \mid \boxminus)$
$\Pr(\cdot)$	0.90	0.97	0.03	0.05	0.95	0.66	0.34	0.25	0.75
$\mathrm{lPr}(\cdot)$	3.89	3.97	0.49	1.00	3.95	3.58	2.92	2.61	3.71

w_0	v_1^1	v_1^2	v_2^1	v_2^2	v_3^1	v_3^2	v_4^1	v_4^2
-2.19	-2.97	3.46	2.95	-2.95	0.4	-2.83	1.17	-1.32

Γ^a	δ_1	δ_2	δ_3	δ_4	$\Phi = -\Gamma^\omega$
2.56	6.43	5.90	0	2.49	12.26

Key concepts & outcomes – computing δ's

	$\Pr(\boxplus)$	$\Pr(\neg r_1 \mid \boxplus)$	$\Pr(r_1 \mid \boxplus)$	$\Pr(\neg r_2 \mid \boxplus)$	$\Pr(r_2 \mid \boxplus)$	$\Pr(\neg r_3 \mid \boxplus)$	$\Pr(r_3 \mid \boxplus)$	$\Pr(\neg r_4 \mid \boxplus)$	$\Pr(r_4 \mid \boxplus)$
$\Pr(\cdot)$	0.10	0.05	0.95	0.95	0.05	0.98	0.02	0.80	0.20
$l\Pr(\cdot)$	1.70	1.00	3.95	3.95	1.00	3.98	0.09	3.78	2.39

	$\Pr(\boxminus)$	$\Pr(\neg r_1 \mid \boxminus)$	$\Pr(r_1 \mid \boxminus)$	$\Pr(\neg r_2 \mid \boxminus)$	$\Pr(r_2 \mid \boxminus)$	$\Pr(\neg r_3 \mid \boxminus)$	$\Pr(r_3 \mid \boxminus)$	$\Pr(\neg r_4 \mid \boxminus)$	$\Pr(r_4 \mid \boxminus)$
$\Pr(\cdot)$	0.90	0.97	0.03	0.05	0.95	0.66	0.34	0.25	0.75
$l\Pr(\cdot)$	3.89	3.97	0.49	1.00	3.95	3.58	2.92	2.61	3.71

w_0	v_1^1	v_1^2	v_2^1	v_2^2	v_3^1	v_3^2	v_4^1	v_4^2
-2.19	-2.97	3.46	2.95	-2.95	0.4	-2.83	1.17	-1.32

Γ^a	δ_1	δ_2	δ_3	δ_4	$\Phi = -\Gamma^\omega$
2.56	6.43	5.90	0	2.49	12.26

Key concepts & outcomes – computing δ's

	$\Pr(\boxplus)$	$\Pr(\neg r_1 \mid \boxplus)$	$\Pr(r_1 \mid \boxplus)$	$\Pr(\neg r_2 \mid \boxplus)$	$\Pr(r_2 \mid \boxplus)$	$\Pr(\neg r_3 \mid \boxplus)$	$\Pr(r_3 \mid \boxplus)$	$\Pr(\neg r_4 \mid \boxplus)$	$\Pr(r_4 \mid \boxplus)$
$\Pr(\cdot)$	0.10	0.05	0.95	0.95	0.05	0.98	0.02	0.80	0.20
$\mathrm{lPr}(\cdot)$	1.70	1.00	3.95	3.95	1.00	3.98	0.09	3.78	2.39

	$\Pr(\boxminus)$	$\Pr(\neg r_1 \mid \boxminus)$	$\Pr(r_1 \mid \boxminus)$	$\Pr(\neg r_2 \mid \boxminus)$	$\Pr(r_2 \mid \boxminus)$	$\Pr(\neg r_3 \mid \boxminus)$	$\Pr(r_3 \mid \boxminus)$	$\Pr(\neg r_4 \mid \boxminus)$	$\Pr(r_4 \mid \boxminus)$
$\Pr(\cdot)$	0.90	0.97	0.03	0.05	0.95	0.66	0.34	0.25	0.75
$\mathrm{lPr}(\cdot)$	3.89	3.97	0.49	1.00	3.95	3.58	2.92	2.61	3.71

w_0	v_1^1	v_1^2	v_2^1	v_2^2	v_3^1	v_3^2	v_4^1	v_4^2
-2.19	-2.97	3.46	2.95	-2.95	0.4	-2.83	1.17	-1.32

Γ^a	δ_1	δ_2	δ_3	δ_4	$\Phi = -\Gamma^\omega$
2.56	6.43	5.90	0	2.49	12.26

Key concepts & outcomes – 0-1 ILP

	$\Pr(\boxplus)$	$\Pr(\neg r_1 \mid \boxplus)$	$\Pr(r_1 \mid \boxplus)$	$\Pr(\neg r_2 \mid \boxplus)$	$\Pr(r_2 \mid \boxplus)$	$\Pr(\neg r_3 \mid \boxplus)$	$\Pr(r_3 \mid \boxplus)$	$\Pr(\neg r_4 \mid \boxplus)$	$\Pr(r_4 \mid \boxplus)$
$\Pr(\cdot)$	0.10	0.05	0.95	0.95	0.05	0.98	0.02	0.80	0.20
$\mathrm{lPr}(\cdot)$	1.70	1.00	3.95	3.95	1.00	3.98	0.09	3.78	2.39

	$\Pr(\boxminus)$	$\Pr(\neg r_1 \mid \boxminus)$	$\Pr(r_1 \mid \boxminus)$	$\Pr(\neg r_2 \mid \boxminus)$	$\Pr(r_2 \mid \boxminus)$	$\Pr(\neg r_3 \mid \boxminus)$	$\Pr(r_3 \mid \boxminus)$	$\Pr(\neg r_4 \mid \boxminus)$	$\Pr(r_4 \mid \boxminus)$
$\Pr(\cdot)$	0.90	0.97	0.03	0.05	0.95	0.66	0.34	0.25	0.75
$\mathrm{lPr}(\cdot)$	3.89	3.97	0.49	1.00	3.95	3.58	2.92	2.61	3.71

Optimization problem:

$$\begin{aligned} \min \quad & \textstyle\sum_{i=1}^{n} p_i \\ \text{s.t.} \quad & \textstyle\sum_{i=1}^{n} \delta_i p_i > \Phi \\ & p_i \in \{0, 1\} \end{aligned}$$

Key concepts & outcomes – 0-1 ILP

	$\Pr(\boxplus)$	$\Pr(\neg r_1 \mid \boxplus)$	$\Pr(r_1 \mid \boxplus)$	$\Pr(\neg r_2 \mid \boxplus)$	$\Pr(r_2 \mid \boxplus)$	$\Pr(\neg r_3 \mid \boxplus)$	$\Pr(r_3 \mid \boxplus)$	$\Pr(\neg r_4 \mid \boxplus)$	$\Pr(r_4 \mid \boxplus)$
$\Pr(\cdot)$	0.10	0.05	0.95	0.95	0.05	0.98	0.02	0.80	0.20
$\mathrm{lPr}(\cdot)$	1.70	1.00	3.95	3.95	1.00	3.98	0.09	3.78	2.39

	$\Pr(\boxminus)$	$\Pr(\neg r_1 \mid \boxminus)$	$\Pr(r_1 \mid \boxminus)$	$\Pr(\neg r_2 \mid \boxminus)$	$\Pr(r_2 \mid \boxminus)$	$\Pr(\neg r_3 \mid \boxminus)$	$\Pr(r_3 \mid \boxminus)$	$\Pr(\neg r_4 \mid \boxminus)$	$\Pr(r_4 \mid \boxminus)$
$\Pr(\cdot)$	0.90	0.97	0.03	0.05	0.95	0.66	0.34	0.25	0.75
$\mathrm{lPr}(\cdot)$	3.89	3.97	0.49	1.00	3.95	3.58	2.92	2.61	3.71

Optimization problem:

$$\begin{aligned} \min \quad & \textstyle\sum_{i=1}^{n} p_i \\ \text{s.t.} \quad & \textstyle\sum_{i=1}^{n} \delta_i p_i > \Phi \\ & p_i \in \{0, 1\} \end{aligned}$$

Special case of knapsack;
can solve in log-linear time

Key concepts & outcomes – 0-1 ILP

	$\Pr(\boxplus)$	$\Pr(\neg r_1 \mid \boxplus)$	$\Pr(r_1 \mid \boxplus)$	$\Pr(\neg r_2 \mid \boxplus)$	$\Pr(r_2 \mid \boxplus)$	$\Pr(\neg r_3 \mid \boxplus)$	$\Pr(r_3 \mid \boxplus)$	$\Pr(\neg r_4 \mid \boxplus)$	$\Pr(r_4 \mid \boxplus)$
$\Pr(\cdot)$	0.10	0.05	0.95	0.95	0.05	0.98	0.02	0.80	0.20
$\mathrm{lPr}(\cdot)$	1.70	1.00	3.95	3.95	1.00	3.98	0.09	3.78	2.39

	$\Pr(\boxminus)$	$\Pr(\neg r_1 \mid \boxminus)$	$\Pr(r_1 \mid \boxminus)$	$\Pr(\neg r_2 \mid \boxminus)$	$\Pr(r_2 \mid \boxminus)$	$\Pr(\neg r_3 \mid \boxminus)$	$\Pr(r_3 \mid \boxminus)$	$\Pr(\neg r_4 \mid \boxminus)$	$\Pr(r_4 \mid \boxminus)$
$\Pr(\cdot)$	0.90	0.97	0.03	0.05	0.95	0.66	0.34	0.25	0.75
$\mathrm{lPr}(\cdot)$	3.89	3.97	0.49	1.00	3.95	3.58	2.92	2.61	3.71

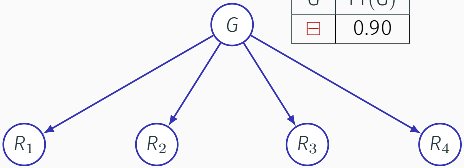
Can enumerate min. sols
w/ log-linear delay

Optimization problem:

$$
\begin{aligned}
\min \quad & \textstyle\sum_{i=1}^{n} p_i \\
\text{s.t.} \quad & \textstyle\sum_{i=1}^{n} \delta_i p_i > \Phi \\
& p_i \in \{0, 1\}
\end{aligned}
$$

Special case of knapsack;
can solve in log-linear time

Key concepts & outcomes – finding one PI-explanation



G	$\Pr(G)$
⊟	0.90

G	$\Pr(R_1 \mid G)$
⊞	0.95
⊟	0.03

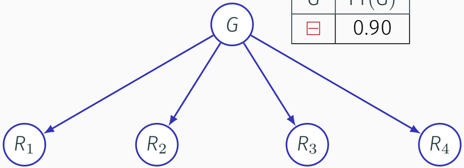
G	$\Pr(R_2 \mid G)$
⊞	0.05
⊟	0.95

G	$\Pr(R_3 \mid G)$
⊞	0.02
⊟	0.34

G	$\Pr(R_4 \mid G)$
⊞	0.20
⊟	0.75

	δ_1	δ_2	δ_4	δ_3	
Sorted	6.43	5.90	2.49	0	$\Phi = 12.26$
Sum					0

Key concepts & outcomes – finding one PI-explanation



G	$\Pr(G)$
$\boxminus$	0.90

G	$\Pr(R_1 \mid G)$
$\boxplus$	0.95
$\boxminus$	0.03

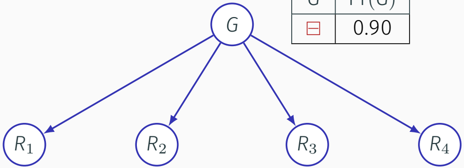
G	$\Pr(R_2 \mid G)$
$\boxplus$	0.05
$\boxminus$	0.95

G	$\Pr(R_3 \mid G)$
$\boxplus$	0.02
$\boxminus$	0.34

G	$\Pr(R_4 \mid G)$
$\boxplus$	0.20
$\boxminus$	0.75

	δ_1	δ_2	δ_4	δ_3	
Sorted	6.43	5.90	2.49	0	$\Phi = 12.26$
Sum	6.43				6.43

Key concepts & outcomes – finding one PI-explanation



G	$\Pr(G)$
$\boxminus$	0.90

G	$\Pr(R_1 \mid G)$
$\boxplus$	0.95
$\boxminus$	0.03

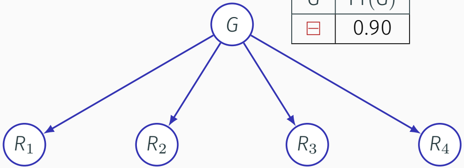
G	$\Pr(R_2 \mid G)$
$\boxplus$	0.05
$\boxminus$	0.95

G	$\Pr(R_3 \mid G)$
$\boxplus$	0.02
$\boxminus$	0.34

G	$\Pr(R_4 \mid G)$
$\boxplus$	0.20
$\boxminus$	0.75

	δ_1	δ_2	δ_4	δ_3	
Sorted	6.43	5.90	2.49	0	$\Phi = 12.26$
Sum	6.43	12.33			$12.33 > \Phi$!

Key concepts & outcomes – finding one PI-explanation



G	$\Pr(G)$
$\boxminus$	0.90

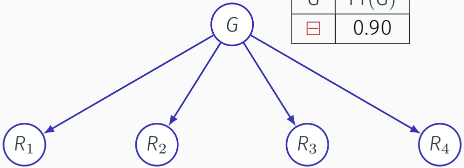
G	$\Pr(R_1 \mid G)$
$\boxplus$	0.95
$\boxminus$	0.03

G	$\Pr(R_2 \mid G)$
$\boxplus$	0.05
$\boxminus$	0.95

G	$\Pr(R_3 \mid G)$
$\boxplus$	0.02
$\boxminus$	0.34

G	$\Pr(R_4 \mid G)$
$\boxplus$	0.20
$\boxminus$	0.75

	δ_1	δ_2	δ_4	δ_3	
Sorted	6.43	5.90	2.49	0	$\Phi = 12.26$
Sum	6.43	12.33	–	–	$12.33 > \Phi$!



G	$\Pr(G)$
$\boxminus$	0.90

G	$\Pr(R_1 \mid G)$
$\boxplus$	0.95
$\boxminus$	0.03

G	$\Pr(R_2 \mid G)$
$\boxplus$	0.05
$\boxminus$	0.95

G	$\Pr(R_3 \mid G)$
$\boxplus$	0.02
$\boxminus$	0.34

G	$\Pr(R_4 \mid G)$
$\boxplus$	0.20
$\boxminus$	0.75

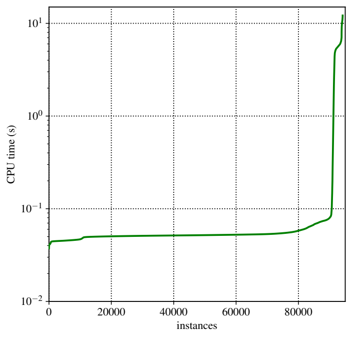
	δ_1	δ_2	δ_4	δ_3	
Sorted	6.43	5.90	2.49	0	$\Phi = 12.26$
Sum	6.43	12.33	–	–	$12.33 > \Phi$!

PI-explanation:

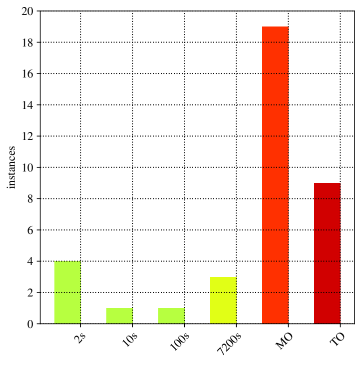
$(p_1 = 1) \wedge (p_2 = 1)$

i.e. $(e_1 = 1) \wedge (e_2 = 0)$

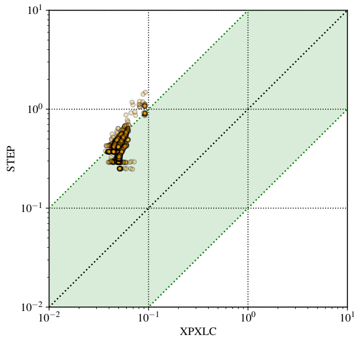
Overview of experimental results



(a) Raw performance of XPXLC



(b) Performance of STEP (with MOs & TOs)



(c) XPXLC vs STEP (no comp. time)

Our work (XPXLC) vs. STEP [SCD18, DH20]

Questions on explaining NBCs & XLCs?

Outline

Overview

[INM19b]

- Vast body of work on computing explanations (XPs)
  - Mostly heuristic approaches, with recent rigorous solutions

Overview

[INM19b]

- Vast body of work on computing explanations (XPs)
  - Mostly heuristic approaches, with recent rigorous solutions
- Vast body of work on coping with adversarial examples (AEs)
  - Both heuristic and rigorous approaches

Overview

[INM19b]

- Vast body of work on computing explanations (XPs)
  - Mostly heuristic approaches, with recent rigorous solutions
- Vast body of work on coping with adversarial examples (AEs)
  - Both heuristic and rigorous approaches
- Can XPs and AEs be somehow related?

Overview

[INM19b]

- Vast body of work on computing explanations (XPs)
    - Mostly heuristic approaches, with recent rigorous solutions
- Vast body of work on coping with adversarial examples (AEs)
    - Both heuristic and rigorous approaches
- Can XPs and AEs be somehow related?
    - Recent work observed that some connection existed, but formal connection has been elusive

Overview

[INM19b]

- Vast body of work on computing explanations (XPs)
  - Mostly heuristic approaches, with recent rigorous solutions
- Vast body of work on coping with adversarial examples (AEs)
  - Both heuristic and rigorous approaches
- Can XPs and AEs be somehow related?
  - Recent work observed that some connection existed, but formal connection has been elusive
- We proposed a (first) link between XPs and AEs [INM19b]

Overview

[INM19b]

- Vast body of work on computing explanations (XPs)
  - Mostly heuristic approaches, with recent rigorous solutions
- Vast body of work on coping with adversarial examples (AEs)
  - Both heuristic and rigorous approaches
- Can XPs and AEs be somehow related?
  - Recent work observed that some connection existed, but formal connection has been elusive
- We proposed a (first) link between XPs and AEs [INM19b]
  - The work exploits hitting set duality, first studied in model-based diagnosis [Rei87]

A well-known example

[RN10]

Example	Input Attributes										Goal
	Alt	Bar	Fri	Hun	Pat	Price	Rain	Res	Type	Est	WillWait
x_1	Yes	No	No	Yes	Some	$$$	No	Yes	French	0–10	$y_1 =$ Yes
x_2	Yes	No	No	Yes	Full	$	No	No	Thai	30–60	$y_2 =$ No
x_3	No	Yes	No	No	Some	$	No	No	Burger	0–10	$y_3 =$ Yes
x_4	Yes	No	Yes	Yes	Full	$	Yes	No	Thai	10–30	$y_4 =$ Yes
x_5	Yes	No	Yes	No	Full	$$$	No	Yes	French	>60	$y_5 =$ No
x_6	No	Yes	No	Yes	Some	$$	Yes	Yes	Italian	0–10	$y_6 =$ Yes
x_7	No	Yes	No	No	None	$	Yes	No	Burger	0–10	$y_7 =$ No
x_8	No	No	No	Yes	Some	$$	Yes	Yes	Thai	0–10	$y_8 =$ Yes
x_9	No	Yes	Yes	No	Full	$	Yes	No	Burger	>60	$y_9 =$ No
x_{10}	Yes	Yes	Yes	Yes	Full	$$$	No	Yes	Italian	10–30	$y_{10} =$ No
x_{11}	No	No	No	No	None	$	No	No	Thai	0–10	$y_{11} =$ No
x_{12}	Yes	Yes	Yes	Yes	Full	$	No	No	Burger	30–60	$y_{12} =$ Yes

A well-known example (Cont.)

- 10 features:

$$\{\text{A(lternate)}, \text{B(ar)}, \text{W(eekend)}, \text{H(ungry)}, \text{Pa(trons)}, \text{Pr(ice)}, \text{Ra(in)}, \text{Re(serv.)}, \text{T(ype)}, \text{E(stim.)}\}$$

- Example instance (x_1, with outcome $y_1 = \text{Yes}$):

$$\{\text{A}, \neg\text{B}, \neg\text{W}, \text{H}, (\text{Pa} = \text{Some}), (\text{Pr} = \$\$\$), \neg\text{Ra}, \text{Re}, (\text{T} = \text{French}), (\text{E} = 0\text{–}10)\}$$

- A possible **decision set** (obtained with some off-the-shelf tool):

$$
\begin{array}{llll}
\textbf{IF} & (\text{Pa} = \text{Some}) \land \neg(\text{E} = {>}60) & \textbf{THEN} & (\text{Wait} = \text{Yes}) \quad (\text{R1}) \\
\textbf{IF} & \text{W} \land \neg(\text{Pr} = \$\$\$) \land \neg(\text{E} = {>}60) & \textbf{THEN} & (\text{Wait} = \text{Yes}) \quad (\text{R2}) \\
\textbf{IF} & \neg\text{W} \land \neg(\text{Pa} = \text{Some}) & \textbf{THEN} & (\text{Wait} = \text{No}) \quad (\text{R3}) \\
\textbf{IF} & (\text{E} = {>}60) & \textbf{THEN} & (\text{Wait} = \text{No}) \quad (\text{R4}) \\
\textbf{IF} & \neg(\text{Pa} = \text{Some}) \land (\text{Pr} = \$\$\$) & \textbf{THEN} & (\text{Wait} = \text{No}) \quad (\text{R5})
\end{array}
$$

Counterexamples & breaks

- Counterexamples:

  A subset-minimal set $\mathcal{C}$ of literals is a counterexample (CEx) to a prediction π, if $\mathcal{C} \vDash (\mathcal{M} \rightarrow \rho)$, with $\rho \in \mathbb{K} \wedge \rho \neq \pi$

Counterexamples & breaks

- Counterexamples:

  A subset-minimal set $\mathcal{C}$ of literals is a counterexample (CEx) to a prediction π, if $\mathcal{C} \vDash (\mathcal{M} \rightarrow \rho)$, with $\rho \in \mathbb{K} \wedge \rho \nequiv \pi$

- Breaks:

  A literal τ_i breaks a set of literals $\mathcal{S}$ (each denoting a different feature) if $\mathcal{S}$ contains a literal inconsistent with τ_i

Counterexamples & breaks

- Counterexamples:

  A subset-minimal set $\mathcal{C}$ of literals is a counterexample (CEx) to a prediction π, if $\mathcal{C} \vDash (\mathcal{M} \rightarrow \rho)$, with $\rho \in \mathbb{K} \wedge \rho \neq \pi$

- Breaks:

  A literal τ_i breaks a set of literals $\mathcal{S}$ (each denoting a different feature) if $\mathcal{S}$ contains a literal inconsistent with τ_i

- Back to the example, consider prediction $(\text{Wait} = \text{Yes})$:

Counterexamples & breaks

- Counterexamples:

  A subset-minimal set $\mathcal{C}$ of literals is a counterexample (CEx) to a prediction π, if $\mathcal{C} \vDash (\mathcal{M} \rightarrow \rho)$, with $\rho \in \mathbb{K} \land \rho \neq \pi$

- Breaks:

  A literal τ_i breaks a set of literals $\mathcal{S}$ (each denoting a different feature) if $\mathcal{S}$ contains a literal inconsistent with τ_i

- Back to the example, consider prediction $(\text{Wait} = \text{Yes})$:
  - Using (R1) (and assuming a consistent instance), an explanation is:

$$(\text{Pa} = \text{Some}) \land \neg(\text{E} = {>}60)$$

Counterexamples & breaks

- Counterexamples:

  A subset-minimal set $\mathcal{C}$ of literals is a counterexample (CEx) to a prediction π, if $\mathcal{C} \vDash (\mathcal{M} \rightarrow \rho)$, with $\rho \in \mathbb{K} \wedge \rho \neq \pi$

- Breaks:

  A literal τ_i breaks a set of literals $\mathcal{S}$ (each denoting a different feature) if $\mathcal{S}$ contains a literal inconsistent with τ_i

- Back to the example, consider prediction $(\text{Wait} = \text{Yes})$:
  - Using (R1) (and assuming a consistent instance), an explanation is:

$$(\text{Pa} = \text{Some}) \wedge \neg(\text{E} = {>}60)$$

  - Due to (R5), a counterexample is:

$$\neg(\text{Pa} = \text{Some}) \wedge (\text{Pr} = \$\$\$)$$

Counterexamples & breaks

- Counterexamples:

  A subset-minimal set $\mathcal{C}$ of literals is a counterexample (CEx) to a prediction π, if $\mathcal{C} \vDash (\mathcal{M} \to \rho)$, with $\rho \in \mathbb{K} \wedge \rho \nVdash \pi$

- Breaks:

  A literal τ_i breaks a set of literals $\mathcal{S}$ (each denoting a different feature) if $\mathcal{S}$ contains a literal inconsistent with τ_i

- Back to the example, consider prediction $(\text{Wait} = \text{Yes})$:
  - Using (R1) (and assuming a consistent instance), an explanation is:

    $$(\text{Pa} = \text{Some}) \wedge \neg(\text{E} = {>}60)$$

  - Due to (R5), a counterexample is:

    $$\neg(\text{Pa} = \text{Some}) \wedge (\text{Pr} = \$\$\$)$$

  - XP $\mathcal{S}_1 = \{(\text{Pa} = \text{Some}), \neg(\text{E} = {>}60)\}$ breaks CEx $\mathcal{S}_2 = \{\neg(\text{Pa} = \text{Some}), (\text{Pr} = \$\$\$)\}$ and vice-versa

Some preliminary results

1. Relationship between XPs with CEx's:

Some preliminary results

1. Relationship between XPs with CEx's:
   - Each XP breaks every CEx

Some preliminary results

1. Relationship between XPs with CEx's:
   - Each XP breaks every CEx
   - Each CEx breaks every XP

Some preliminary results

1. Relationship between XPs with CEx's:
   - Each XP breaks every CEx
   - Each CEx breaks every XP

∴ XPs can be computed from all CEx's (by HSD) and vice-versa

Some preliminary results

1. Relationship between XPs with CEx's:
   - Each XP breaks every CEx
   - Each CEx breaks every XP

   ∴ XPs can be computed from all CEx's (by HSD) and vice-versa

2. Given instance $\mathcal{I}$, an AE can be computed from closest CEx

Revisiting the example

- Restaurant dataset
- ML model is decision set (shown earlier)
- Prediction is $(\text{Wait} = \text{Yes})$
- Global explanations:
  1. $(\text{Pa} = \text{Some}) \land \neg(\text{E} = {>}60)$
  2. $\text{W} \land \neg(\text{Pr} = \$\$\$) \land \neg(\text{E} = {>}60)$
- Counterexamples:
  1. $\neg\text{W} \land \neg(\text{Pa} = \text{Some})$
  2. $(\text{E} = {>}60)$
  3. $\neg(\text{Pa} = \text{Some}) \land (\text{Pr} = \$\$\$)$
- The XP's break the CEx's and vice-versa

Questions for part 2?

Part 3

Fairness

Outline

Understanding fairness

Fairness Through Unawareness

Relating Fairness with Explanations

Learning Fair Models

Some questions regarding fairness

[ICS+20]

- What should be the criterion for fairness of a model (a classifier)?
- What should be the criterion for dataset bias?
- What should be the criterion for fairness of a particular decision?
- How to learn a fair model from biased data?

Basic definitions

- Classifier: boolean function $\varphi(\mathbf{x}, \mathbf{y}) \in \{0, 1\}$, where
  - $\mathbf{x}$: values of **non-protected** features (salary, age, ...), and
  - $\mathbf{y}$: values of **protected** features (gender, race, ...).
- Dataset: set of tuples $\langle \mathbf{x}, \mathbf{y}, c \rangle$ with $c \in \{0, 1\}$
- Examples:
  1. Should a bank approve a loan to a customer?
  2. Should a judge release a prisoner on probation?

Outline

Criterion: fairness through unawareness (FTU)

- FTU: φ is a function only of the non-protected features $\mathbf{x}$
- FTU criterion for testing unfairness of model:

$$\exists \mathbf{x}\, \exists (\mathbf{y}_1, \mathbf{y}_2).\, [\mathbf{y}_1 \neq \mathbf{y}_2 \;\wedge\; \varphi(\mathbf{x}, \mathbf{y}_1) \neq \varphi(\mathbf{x}, \mathbf{y}_2)]$$

E.g. Alice and Bob are identical (same salary, age, ...), Alice is refused a loan but Bob isn't

- Optimisation: only need to test criterion for $\mathbf{y}_1, \mathbf{y}_2$ which differ on a single feature

Criterion: fairness through unawareness (FTU)

- FTU: φ is a function only of the non-protected features $\mathbf{x}$
- FTU criterion for testing unfairness of model:

$$\exists \mathbf{x}\, \exists (\mathbf{y}_1, \mathbf{y}_2).\, [\mathbf{y}_1 \neq \mathbf{y}_2 \;\wedge\; \varphi(\mathbf{x}, \mathbf{y}_1) \neq \varphi(\mathbf{x}, \mathbf{y}_2)]$$

E.g. Alice and Bob are identical (same salary, age, ...), Alice is refused a loan but Bob isn't

- Optimisation: only need to test criterion for $\mathbf{y}_1, \mathbf{y}_2$ which differ on a single feature

Possible drawbacks of **FTU:**

- There may be correlations between protected and non-protected features

  E.g.: the bank isn't unfair to women, they just don't give loans to people who are pregnant!
- Positive discrimination may be a good thing

  E.g.: height restrictions for army recruits are less strict for women

FTU as a criterion for dataset bias

- FTU criterion for testing bias of a dataset $\mathcal{D}$:

$$\exists \mathbf{x}, \mathbf{y}_1, \mathbf{y}_2.[\mathbf{y}_1 \neq \mathbf{y}_2 \ \wedge \ \langle \mathbf{x}, \mathbf{y}_1, 0 \rangle, \langle \mathbf{x}, \mathbf{y}_2, 1 \rangle \in \mathcal{D}]$$

  - Criterion can be applied even if $\mathcal{D}$ is inconsistent (i.e. $\exists \mathbf{x}, \mathbf{y}[\langle \mathbf{x}, \mathbf{y}, 0 \rangle, \langle \mathbf{x}, \mathbf{y}, 1 \rangle \in \mathcal{D}]$)
  - Criterion can be tested in linear time (using hash tables) since it is equivalent to: $\exists \mathbf{x}$ such that

$$\begin{aligned} |\{c : \exists \mathbf{y}, \langle \mathbf{x}, \mathbf{y}, c \rangle \in \mathcal{D}\}| &> 1 \\ |\{\mathbf{y} : \exists c, \langle \mathbf{x}, \mathbf{y}, c| \rangle \in \mathcal{D}\}| &> 1 \end{aligned}$$

Which criterion to pick?

- Axioms for a dataset-bias criterion:
  - Coding-independence: independent of renaming or merging of non-protected features/protected features
  - Monotonicity: eliminating unprotected features cannot reduce bias
  - Not arbitrary: if all data is identical on the protected features, then unbiased
  - Discerning: the criterion is non-trivial
  - Simplicity: bias can be proved by exhibiting just 2 examples

Theorem
The only criterion satisfying these 5 axioms is FTU

Theorem
There is no criterion which satisfies the 5 axioms and is invariant to the addition of irrelevant features (such as month of birth)

Outline

Local fairness: fairness of a particular decision

- An example:
  - Emma wants to know if she was refused a loan because she is a woman
  - The bank uses a simple model: refuse a loan if the client is unemployed or if they are a woman
  - This model is clearly unfair with respect to gender, but
    - The bank claims that the *decision* is fair since they refused the loan because Emma is unemployed
    - Emma points out there are two possible explanations for the refusal:

      (1) she is unemployed, or that
      (2) she is a woman,

      and hence the decision should be considered unfair

Local fairness: fairness of a particular decision

- An example:
  - Emma wants to know if she was refused a loan because she is a woman
  - The bank uses a simple model: refuse a loan if the client is unemployed or if they are a woman
  - This model is clearly unfair with respect to gender, but
    - The bank claims that the *decision* is fair since they refused the loan because Emma is unemployed
    - Emma points out there are two possible explanations for the refusal:

      (1) she is unemployed, or that
      (2) she is a woman,

      and hence the decision should be considered unfair
  - **Who is right?**

Fairness of a particular decision from explanations

- **Recap:** a PI-explanation $\mathcal{E}$ of a prediction $\varphi(\mathbf{z}) = c$ is a subset-minimal set of literals from the literals $\mathcal{Z}$ of $\mathbf{z} \in \mathbb{F}$, which entails the prediction c:

$$\forall(\mathbf{x} \in \mathbb{F}).\, [\mathcal{E}(\mathbf{x}) \rightarrow (\varphi(\mathbf{x}) = c)]$$

  - E.g. with $\varphi(x, y) = x \wedge y$, the decision $\varphi(0, 0) = 0$ has 2 PI-explanations: $\mathcal{E}_1 = (\neg x)$, and $\mathcal{E}_2 = (\neg y)$

- An explanation is fair if it includes no protected features
- A prediction $\varphi(\mathbf{z}) = c$ is:
  - Universally fair: if all of its explanations are fair
  - Existentially fair: if at least one of its explanations is fair

Fairness of a particular decision from explanations

- **Recap:** a PI-explanation $\mathcal{E}$ of a prediction $\varphi(\mathbf{z}) = c$ is a subset-minimal set of literals from the literals $\mathcal{Z}$ of $\mathbf{z} \in \mathbb{F}$, which entails the prediction c:

$$\forall(\mathbf{x} \in \mathbb{F}).\, [\mathcal{E}(\mathbf{x}) \rightarrow (\varphi(\mathbf{x}) = c)]$$

  - E.g. with $\varphi(x, y) = x \wedge y$, the decision $\varphi(0, 0) = 0$ has 2 PI-explanations: $\mathcal{E}_1 = (\neg x)$, and $\mathcal{E}_2 = (\neg y)$

- An explanation is fair if it includes no protected features
- A prediction $\varphi(\mathbf{z}) = c$ is:
  - Universally fair: if all of its explanations are fair
  - Existentially fair: if at least one of its explanations is fair

- Back to the example:
  Emma's loan refusal decision is existentially fair but not universally fair

Complexity of checking fairness

- A model φ is fair iff all its decisions are universally fair
  - Checking fairness of a model is in co-NP
- Checking existential fairness of a decision $\varphi(\mathbf{z}) = c$ is in co-NP
  - It can be solved by exhaustive search over only the protected features
- Checking universal fairness of a decision $\varphi(\mathbf{z}) = c$ is in Π_2^P

Outline

Understanding fairness

Fairness Through Unawareness

Relating Fairness with Explanations

Learning Fair Models

Learning fair models (from a possibly biased dataset)

Principle: we impose fairness

- **Obs:** this is necessarily at the cost of accuracy in the case of a biased dataset

Majority-vote solution: since $\varphi(\mathbf{x}, \mathbf{y})$ must be a function of $\mathbf{x}$ only, we maximise accuracy by choosing the most common class c as $\mathbf{y}$ varies and $\mathbf{x}$ remains fixed

Obs: We may further sacrifice accuracy in order to obtain a simple (and hence more human-understandable) model

Fair decision sets with SAT

Problem: learn a boolean function $\varphi(x_1, \ldots, x_m)$ from a set of n examples

- The model φ is necessarily **fair** since it is a function of non-protected features $x_1, \ldots, x_m$ only

- In order to obtain a human-understandable model φ, we construct (multiple) K-term DNFs, where K is a small constant

- We can encode this problem as a SAT instance with variables:
  - $p_{jk} = 1$ if the kth term contains x_j
  - $q_{jk} = 1$ if the kth term contains $\neg x_j$
  - $v_{ik} = 1$ if the ith example satisfies the kth term

Fair decision sets with SAT

- Clauses of the SAT instance (for 1 DNF):
  1. Each positive example satisfies some term ($O(n)$ size-K clauses)
  2. No negative example satisfies any term ($O(nK)$ size-m clauses)
  3. Constraints coding the semantics of the variables ($O(nmK)$ binary clauses)

  where n = number of examples, m = number of features, K = number of terms in the DNF

Example of the *Compas* dataset

- Dataset is derived from the COMPAS algorithm used for scoring a criminal defendant's likelihood of reoffending
  - It includes protected features, such as African American, etc.
  - Dataset is so biased that the *maximum feasible* accuracy is only 69.73%
  - By sacrificing accuracy further to obtain a more interpretable (i.e. smaller) model, we found the following decision set which has 66.32% accuracy and is fair:

$$
\begin{array}{llll}
\text{IF} & \#\mathit{Priors} > 17.5 \wedge \neg \mathit{score\_factor} & \text{THEN} & \mathit{Two\_yr\_Recidivism} \\
\text{IF} & \#\mathit{Priors} > 17.5 \wedge \mathit{Age} > 45 \wedge \mathit{Misdemeanor} & \text{THEN} & \mathit{Two\_yr\_Recidivism} \\
\text{IF} & \#\mathit{Priors} \leqslant 17.5 & \text{THEN} & \neg \mathit{Two\_yr\_Recidivism} \\
\text{IF} & \mathit{score\_factor} \wedge \mathit{Age} \leqslant 45 & \text{THEN} & \neg \mathit{Two\_yr\_Recidivism} \\
\text{IF} & \mathit{score\_factor} \wedge \neg \mathit{Misdemeanor} & \text{THEN} & \neg \mathit{Two\_yr\_Recidivism}
\end{array}
$$

Questions for part 3?

Part 4

Learning (Interpretable Models)

Outline

Classification problems I

Ex.	Vacation (V)	Concert (C)	Meeting (M)	Expo (E)	Hike (H)
e_1	0	0	1	0	0
e_2	1	0	0	0	1
e_3	0	0	1	1	0
e_4	1	0	0	1	1
e_5	0	1	1	0	0
e_6	0	1	1	1	0
e_7	1	1	0	1	1

- Training data (or **examples**/instances): $\mathcal{E} = \{e_1, \ldots, e_M\}$

Classification problems I

Ex.	Vacation (V)	Concert (C)	Meeting (M)	Expo (E)	Hike (H)
e_1	0	0	1	0	0
e_2	1	0	0	0	1
e_3	0	0	1	1	0
e_4	1	0	0	1	1
e_5	0	1	1	0	0
e_6	0	1	1	1	0
e_7	1	1	0	1	1

- Training data (or **examples**/instances): $\mathcal{E} = \{e_1, \ldots, e_M\}$
- Binary **features**: $\mathcal{F} = \{f_1, \ldots, f_K\}$
  - Literals: f_r and $\neg f_r$

Classification problems I

Ex.	Vacation (V)	Concert (C)	Meeting (M)	Expo (E)	Hike (H)
e_1	0	0	1	0	0
e_2	1	0	0	0	1
e_3	0	0	1	1	0
e_4	1	0	0	1	1
e_5	0	1	1	0	0
e_6	0	1	1	1	0
e_7	1	1	0	1	1

- Training data (or **examples**/instances): $\mathcal{E} = \{e_1, \ldots, e_M\}$
- Binary **features**: $\mathcal{F} = \{f_1, \ldots, f_K\}$
  - Literals: f_r and $\neg f_r$
- **Feature space**: $\mathcal{U} \triangleq \prod_{r=1}^{K} \{f_r, \neg f_r\}$

Classification problems I

Ex.	Vacation (V)	Concert (C)	Meeting (M)	Expo (E)	Hike (H)
e_1	0	0	1	0	0
e_2	1	0	0	0	1
e_3	0	0	1	1	0
e_4	1	0	0	1	1
e_5	0	1	1	0	0
e_6	0	1	1	1	0
e_7	1	1	0	1	1

- Training data (or **examples**/instances): $\mathcal{E} = \{e_1, \ldots, e_M\}$
- Binary **features**: $\mathcal{F} = \{f_1, \ldots, f_K\}$
  - Literals: f_r and $\neg f_r$
- **Feature space**: $\mathcal{U} \triangleq \prod_{r=1}^{K} \{f_r, \neg f_r\}$
- Binary classification: $\mathcal{C} = \{c_0 = 0, c_1 = 1\}$
  - $\mathcal{E}$ partitioned into $\mathcal{E}^-$ and $\mathcal{E}^+$

Classification problems II

Ex.	Vacation (V)	Concert (C)	Meeting (M)	Expo (E)	Hike (H)
e_1	0	0	1	0	0
e_2	1	0	0	0	1
e_3	0	0	1	1	0
e_4	1	0	0	1	1
e_5	0	1	1	0	0
e_6	0	1	1	1	0
e_7	1	1	0	1	1

- $e_q \in \mathcal{E}$ represented as a 2-tuple (π_q, ς_q)
  - $\pi_q \in \mathcal{U}$: literals associated with the example
  - $\varsigma_q \in \{0, 1\}$ is the class of example

Classification problems II

Ex.	Vacation (V)	Concert (C)	Meeting (M)	Expo (E)	Hike (H)
e_1	0	0	1	0	0
e_2	1	0	0	0	1
e_3	0	0	1	1	0
e_4	1	0	0	1	1
e_5	0	1	1	0	0
e_6	0	1	1	1	0
e_7	1	1	0	1	1

- $e_q \in \mathcal{E}$ represented as a 2-tuple (π_q, ς_q)
  - $\pi_q \in \mathcal{U}$: literals associated with the example
  - $\varsigma_q \in \{0, 1\}$ is the class of example
- A literal l_r on a feature f_r, $l_r \in \{f_r, \neg f_r\}$, **discriminates** an example e_q if $\pi_q[r] = \neg l_r$
  - I.e. feature r takes the value **opposite** to the value in the tuple of literals of the example

Example

Ex.	Vacation (V)	Concert (C)	Meeting (M)	Expo (E)	Hike (H)
e_1	0	0	1	0	0
e_2	1	0	0	0	1
e_3	0	0	1	1	0
e_4	1	0	0	1	1
e_5	0	1	1	0	0
e_6	0	1	1	1	0
e_7	1	1	0	1	1

- Binary features: $\mathcal{F} = \{f_1, f_2, f_3, f_4\}$
  - $f_1 \triangleq \text{V}$, $f_2 \triangleq \text{C}$, $f_3 \triangleq \text{M}$, and $f_4 \triangleq \text{E}$
- e_1 is represented by the 2-tuple (π_1, ς_1),
  - $\pi_1 = (\neg\text{V}, \neg\text{C}, \text{M}, \neg\text{E})$
  - $\varsigma_1 = 0$
- Literals V, C, ¬M and E **discriminate** e_1
- $\mathcal{U} = \{\text{V}, \neg\text{V}\} \times \{\text{C}, \neg\text{C}\} \times \{\text{M}, \neg\text{M}\} \times \{\text{E}, \neg\text{E}\}$

Goal of explainable classification – our take

[IPNM18]

> Given training data, learn set of DNFs that correctly classify that data, perform suitably well on unseen data, and offer human-understandable explanations for the predictions made

Itemsets & decision sets

- Given $\mathcal{F}$, an **itemset** π is an element of $\mathcal{I} \triangleq \prod_{r=1}^{K} \{f_r, \neg f_r, \mathfrak{u}\}$
  - $\mathfrak{u}$ represents a don't care value

Itemsets & decision sets

- Given $\mathcal{F}$, an **itemset** π is an element of $\mathcal{I} \triangleq \prod_{r=1}^{K} \{f_r, \neg f_r, \mathfrak{u}\}$
  - $\mathfrak{u}$ represents a don't care value

- A **rule** is a 2-tuple (π, ς), with itemset $\pi \in \mathcal{I}$, and class $\varsigma \in \mathcal{C}$
  Rule (π, ς) interpreted as:

  IF all specified literals in π are true, THEN pick class ς

Itemsets & decision sets

- Given $\mathcal{F}$, an **itemset** π is an element of $\mathcal{I} \triangleq \prod_{r=1}^{K}\{f_r, \neg f_r, \mathfrak{u}\}$
  - $\mathfrak{u}$ represents a don't care value

- A **rule** is a 2-tuple (π, ς), with itemset $\pi \in \mathcal{I}$, and class $\varsigma \in \mathcal{C}$
  Rule (π, ς) interpreted as:

  IF all specified literals in π are true, THEN pick class ς

- A **decision set** $\mathbb{S}$ is a finite set of rules – unordered

Itemsets & decision sets

- Given $\mathcal{F}$, an **itemset** π is an element of $\mathcal{I} \triangleq \prod_{r=1}^{K} \{f_r, \neg f_r, \mathfrak{u}\}$
  - $\mathfrak{u}$ represents a don't care value

- A **rule** is a 2-tuple (π, ς), with itemset $\pi \in \mathcal{I}$, and class $\varsigma \in \mathcal{C}$
  Rule (π, ς) interpreted as:

  IF all specified literals in π are true, THEN pick class ς

- A **decision set** $\mathbb{S}$ is a finite set of rules – unordered

- A rule of the form $\mathfrak{D} \triangleq (\emptyset, \varsigma)$ denotes the **default rule** of a decision set $\mathbb{S}$
  - Default rule is **optional** and used **only** when other rules do not apply on some feature space point

Example

Ex.	Vacation (V)	Concert (C)	Meeting (M)	Expo (E)	Hike (H)
e_1	0	0	1	0	0
e_2	1	0	0	0	1
e_3	0	0	1	1	0
e_4	1	0	0	1	1
e_5	0	1	1	0	0
e_6	0	1	1	1	0
e_7	1	1	0	1	1

- Rule 1: $((u, u, \neg M, u), c_1)$
  - Meaning: **if** $\neg$Meeting **then** Hike
- Rule 2: $((\neg V, u, u, u), c_0)$
  - Meaning: **if** $\neg$Vacation **then** $\neg$Hike

Example

Ex.	Vacation (V)	Concert (C)	Meeting (M)	Expo (E)	Hike (H)
e_1	0	0	1	0	0
e_2	1	0	0	0	1
e_3	0	0	1	1	0
e_4	1	0	0	1	1
e_5	0	1	1	0	0
e_6	0	1	1	1	0
e_7	1	1	0	1	1

- Rule 1: $((\mathfrak{u}, \mathfrak{u}, \neg M, \mathfrak{u}), c_1)$
  - Meaning: **if** ¬Meeting **then** Hike
- Rule 2: $((\neg V, \mathfrak{u}, \mathfrak{u}, \mathfrak{u}), c_0)$
  - Meaning: **if** ¬Vacation **then** ¬Hike
- Default rule: $(\varnothing, c_0)$
  - Meaning: if all other rules do not apply, then pick ¬Hike

Issue with unordered rules

- Itemsets $\pi_1, \pi_2 \in \mathcal{I}$ **clash**, $\pi_1 \cap \pi_2 = \emptyset$, if for some coordinate r:
  - $\pi_1[r] = f_r$ and $\pi_2[r] = \neg f_r$, or $\pi_1[r] = \neg f_r$ and $\pi_2[r] = f_r$

Issue with unordered rules – overlap

- Itemsets $\pi_1, \pi_2 \in \mathcal{I}$ **clash**, $\pi_1 \cap \pi_2 = \emptyset$, if for some coordinate r:
  - $\pi_1[r] = f_r$ and $\pi_2[r] = \neg f_r$, or $\pi_1[r] = \neg f_r$ and $\pi_2[r] = f_r$

- Two rules $r_1 = (\pi_1, \varsigma_1)$ and $r_2 = (\pi_2, \varsigma_2)$ **overlap** if π_1 and π_2 do not clash, i.e.

$$\pi_1 \cap \pi_2 \neq \emptyset$$

  - Can be restricted to some set, e.g. $\mathcal{E}$

Issue with unordered rules – overlap

- Itemsets $\pi_1, \pi_2 \in \mathcal{I}$ **clash**, $\pi_1 \cap \pi_2 = \emptyset$, if for some coordinate r:
    - $\pi_1[r] = f_r$ and $\pi_2[r] = \neg f_r$, or $\pi_1[r] = \neg f_r$ and $\pi_2[r] = f_r$
- Two rules $r_1 = (\pi_1, \varsigma_1)$ and $r_2 = (\pi_2, \varsigma_2)$ **overlap** if π_1 and π_2 do not clash, i.e.

$$\pi_1 \cap \pi_2 \neq \emptyset$$

    - Can be restricted to some set, e.g. $\mathcal{E}$
- Forms of overlap:
    - $\oplus$: overall where rules agree in prediction
    - $\ominus$: overlap where rules **disagree** in prediction

Issue with unordered rules – overlap

- Itemsets $\pi_1, \pi_2 \in \mathcal{I}$ **clash**, $\pi_1 \cap \pi_2 = \emptyset$, if for some coordinate r:
  - $\pi_1[r] = f_r$ and $\pi_2[r] = \neg f_r$, or $\pi_1[r] = \neg f_r$ and $\pi_2[r] = f_r$
- Two rules $r_1 = (\pi_1, \varsigma_1)$ and $r_2 = (\pi_2, \varsigma_2)$ **overlap** if π_1 and π_2 do not clash, i.e.

$$\pi_1 \cap \pi_2 \neq \emptyset$$

  - Can be restricted to some set, e.g. $\mathcal{E}$
- Forms of overlap:
  - $\oplus$: overall where rules agree in prediction
  - $\ominus$: overlap where rules **disagree** in prediction
- Our goal:

> Minimize number of rules in decision set, and provide guarantees in terms of overlap, namely $\ominus$-overlap

Example

Ex.	Vacation (V)	Concert (C)	Meeting (M)	Expo (E)	Hike (H)
e_1	0	0	1	0	0
e_2	1	0	0	0	1
e_3	0	0	1	1	0
e_4	1	0	0	1	1
e_5	0	1	1	0	0
e_6	0	1	1	1	0
e_7	1	1	0	1	1

- Decision set:

$$\{((\neg V, \mathfrak{u}, \mathfrak{u}, \mathfrak{u}), c_0), ((\mathfrak{u}, \mathfrak{u}, \neg M, \mathfrak{u}), c_1)\}$$

- No $\mathcal{E}^{\ominus}$-overlap

Example

Ex.	Vacation (V)	Concert (C)	Meeting (M)	Expo (E)	Hike (H)
e_1	0	0	1	0	0
e_2	1	0	0	0	1
e_3	0	0	1	1	0
e_4	1	0	0	1	1
e_5	0	1	1	0	0
e_6	0	1	1	1	0
e_7	1	1	0	1	1

- Decision set:

$$\{((\neg V, u, u, u), c_0), ((u, u, \neg M, u), c_1)\}$$

- No $\mathcal{E}^{\ominus}$-overlap
- But, there exists overlap in feature space
  - $\ominus$-overlap for $(\neg V, \neg C, \neg M, \neg E) \in \mathcal{U} \setminus \mathcal{E}$

Example

Ex.	Vacation (V)	Concert (C)	Meeting (M)	Expo (E)	Hike (H)
e_1	0	0	1	0	0
e_2	1	0	0	0	1
e_3	0	0	1	1	0
e_4	1	0	0	1	1
e_5	0	1	1	0	0
e_6	0	1	1	1	0
e_7	1	1	0	1	1

- Decision set:

  $$\{((\neg V, u, u, u), c_0), ((u, u, \neg M, u), c_1)\}$$

- No $\mathcal{E}^{\ominus}$-overlap
- But, there exists overlap in feature space
  - $\ominus$-overlap for $(\neg V, \neg C, \neg M, \neg E) \in \mathcal{U} \backslash \mathcal{E}$
- However, there exists **no** $\mathcal{U}^{\ominus}$-overlap for decision set:

  $$\{((V, u, u, u), c_1), ((\neg V, u, u, u), c_0)\}$$

Succinct explanations

- If a rule fires, the set of literals represents the **explanation** for the predicted class
  - Explanation is **succinct**: **only** the literals in the rule used; independent of example
- For the default class, **must** pick one **falsified** literal in **every** rule that predicts a different class
  - Explanation is **not succinct**: explanation depends on **each** example
- **Obs:** **Uninteresting** to predict c_1 as **negation** of c_0 (and vice-versa)
  - Explanations also **not** succinct

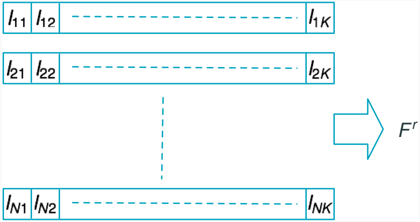
Stating our goals

- Assumptions:
  - Represent $\mathcal{E}^-$ with Boolean function E^0
    - True for each example $\mathcal{E}^-$
  - Represent $\mathcal{E}^+$ with Boolean function E^1
    - True for each example $\mathcal{E}^+$
  - Also, let $E^0 \wedge E^1 \models \bot$

Stating our goals

- Assumptions:
  - Represent $\mathcal{E}^-$ with Boolean function E^0
    - True for each example $\mathcal{E}^-$
  - Represent $\mathcal{E}^+$ with Boolean function E^1
    - True for each example $\mathcal{E}^+$
  - Also, let $E^0 \wedge E^1 \models \bot$

- DNF functions to compute:
  - F^0 for predicting c_0, while **ensuring** $E^0 \models F^0$
  - F^1 for predicting c_1, while **ensuring** $E^1 \models F^1$



An ideal model – MinDS_0

- MinDS_0:

  Find the **smallest** DNF representations of Boolean functions F^0 and F^1, measured in the number of **terms**, such that:

  1. $E^0 \models F^0$
  2. $E^1 \models F^1$
  3. $F^1 \leftrightarrow F^0 \models \bot$

  - No $\mathcal{U}^{\ominus}$-overlap

An ideal model – MinDS_0

- MinDS_0:

  Find the smallest DNF representations of Boolean functions F^0 and F^1, measured in the number of terms, such that:

  1. $E^0 \models F^0$
  2. $E^1 \models F^1$
  3. $F^1 \leftrightarrow F^0 \models \bot$

  - No $\mathcal{U}^{\ominus}$-overlap

- **Obs:** MinDS_0 ensures **succinct** explanations
  - Computes F^0 and F^1 (i.e. **no** negation) and **no** default rule

An ideal model – MinDS$_0$

- MinDS$_0$:

  Find the smallest DNF representations of Boolean functions F^0 and F^1, measured in the number of terms, such that:

  1. $E^0 \models F^0$
  2. $E^1 \models F^1$
  3. $F^1 \leftrightarrow F^0 \models \bot$

  - No $\mathcal{U}^{\ominus}$-overlap

- **Obs:** MinDS$_0$ ensures **succinct** explanations
  - Computes F^0 and F^1 (i.e. **no** negation) and **no** default rule

- Complexity-wise:
  - MinDS$_0$ $\in \Sigma_2^{\mathrm{P}}$
  - A conjecture: MinDS$_0$ hard for Σ_2^{P} (from late 2017)

Curbing our expectations I

- MinDS$_4$: Minimize F^0, given $F^1 \equiv E^1$ constant, and such that
  1. $E^0 \models F^0$
  2. $F^0 \wedge E^1 \models \bot$
  - No $\ominus$-overlap;
  - No succinct explanations for F^1

Curbing our expectations I

- MinDS_4: Minimize F^0, given $F^1 \equiv E^1$ constant, and such that
  1. $E^0 \models F^0$
  2. $F^0 \wedge E^1 \models \bot$
  - No $\ominus$-overlap;
  - No succinct explanations for F^1
- MinDS_3: Same as MinDS_4, but target F^1 given $F^0 \equiv E^0$ constant
  - Also, no $\ominus$-overlap;
  - No succinct explanations for F^0

Curbing our expectations I

- MinDS$_4$: Minimize F^0, given $F^1 \equiv E^1$ constant, and such that
  1. $E^0 \models F^0$
  2. $F^0 \wedge E^1 \models \bot$
  - No $\ominus$-overlap;
  - No succinct explanations for F^1
- MinDS$_3$: Same as MinDS$_4$, but target F^1 given $F^0 \equiv E^0$ constant
  - Also, no $\ominus$-overlap;
  - No succinct explanations for F^0
- MinDS$_2$: Minimize both F^0 and F^1, such that
  1. $E^0 \models F^0$
  2. $E^1 \models F^1$
  3. $F^0 \wedge E^1 \models \bot$
  4. $F^1 \wedge E^0 \models \bot$
  - Also, no $\mathcal{E}^\ominus$-overlap; but $(\mathcal{U} \backslash \mathcal{E})^\ominus$-overlap may exist
  - All explanations succinct

Curbing our expectations II

- MinDS_1: Minimize both F^0 and F^1, such that
    1. $E^0 \models F^0$
    2. $E^1 \models F^1$
    3. $F^1 \wedge F^0 \models \bot$
    - No $\mathcal{U}^{\ominus}$-overlap
    - Default rule may be required for points in $\mathcal{U} \backslash \mathcal{E}$
    - And, default rule explanations not succinct

Curbing our expectations II

- MinDS$_1$: Minimize both F^0 and F^1, such that
  1. $E^0 \models F^0$
  2. $E^1 \models F^1$
  3. $F^1 \wedge F^0 \models \bot$
  - No $\mathcal{U}^{\ominus}$-overlap
  - Default rule may be required for points in $\mathcal{U} \backslash \mathcal{E}$
  - And, default rule explanations not succinct
- Complexity-wise:
  - Decision formulations of MinDS$_1$, MinDS$_2$, MinDS$_3$, MinDS$_4$ are **complete** for NP
    - In principle, could be solved with MaxSAT
    - But no closed MaxSAT models for now

Computing explainable decision sets

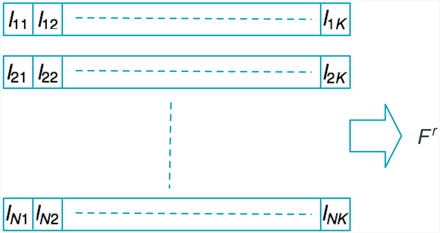
- Our work: [IPNM18]
  - Adapted old SAT encodings to MinDS_3 & MinDS_4 [KKRR'92]
  - Developed new SAT encodings for MinDS_3 & MinDS_4
  - Developed SAT encodings for MinDS_2 and MinDS_1
  - Proposed **symmetry-breaking** constraints (SBPs)

Computing explainable decision sets

- Our work: [IPNM18]
  - Adapted old SAT encodings to MinDS_3 & MinDS_4 [KKRR'92]
  - Developed new SAT encodings for MinDS_3 & MinDS_4
  - Developed SAT encodings for MinDS_2 and MinDS_1
  - Proposed **symmetry-breaking** constraints (SBPs)
- Covered in the lecture: SAT encoding for MinDS_3

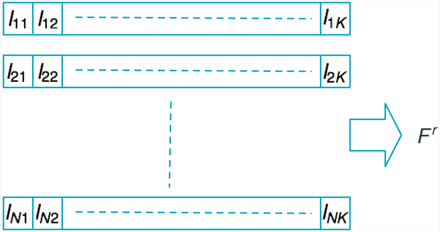
SAT model for MinDS$_3$ – overview

- DNF representation for F^1
- Consider N terms
  - Each term corresponds to a rule



- Allow literals **to be associated or not** with each rule
- Rules for some class **must discriminate** examples of other classes
- Every example **must be covered** by one of the rules for its class

Boolean variables for MinDS$_3$



- s_{jr}: whether a literal in feature r is skipped for rule j
- l_{jr} polarity of literal on feature r for rule j, when the feature is not skipped
- d^0_{jr}: whether feature r of rule j discriminates value 0
- d^1_{jr}: whether feature r of rule j discriminates value 1
- cr_{jq}: whether (used) rule j covers $e_q \in \mathcal{E}^+$

Constraints for MinDS$_3$ I

- Each term must have some literals:

$$\left(\bigvee_{r=1}^{K} \neg s_{jr}\right) \qquad\qquad j \in [N]$$

Constraints for MinDS$_3$ I

- Each term must have some literals:

$$\left(\bigvee_{r=1}^{K} \neg s_{jr}\right) \qquad j \in [N]$$

- Account for which literals are discriminated by which rules:

$$d_{jr}^{0} \leftrightarrow \neg s_{jr} \land l_{jr} \qquad j \in [N] \land r \in [K]$$
$$d_{jr}^{1} \leftrightarrow \neg s_{jr} \land \neg l_{jr} \qquad j \in [N] \land r \in [K]$$

Constraints for MinDS$_3$ I

- Each term must have some literals:

$$\left(\bigvee_{r=1}^{K} \neg s_{jr}\right) \qquad j \in [N]$$

- Account for which literals are discriminated by which rules:

$$d_{jr}^{0} \leftrightarrow \neg s_{jr} \wedge l_{jr} \qquad j \in [N] \wedge r \in [K]$$
$$d_{jr}^{1} \leftrightarrow \neg s_{jr} \wedge \neg l_{jr} \qquad j \in [N] \wedge r \in [K]$$

- Discriminate all the **negative** examples in each term
  - $e_q \in \mathcal{E}^-$: some negative example
  - $\sigma(r, q)$: sign of feature f_r for e_q

$$\left(\bigvee_{r=1}^{K} d_{j,r}^{\sigma(r,q)}\right) \qquad j \in [N] \wedge e_q \in \mathcal{E}^-$$

Constraints for MinDS$_3$ II

- Each positive example must be covered by some rule
  - Define whether a rule covers some specific positive example:

$$cr_{jq} \leftrightarrow \left(\bigwedge_{r=1}^{K} \neg d_{j,r}^{\sigma(r,q)} \right) \qquad j \in [N] \wedge e_q \in \mathcal{E}^+$$

Constraints for MinDS$_3$ II

- Each **positive** example must be covered by some rule
  - Define whether a rule covers some specific positive example:

$$cr_{jq} \leftrightarrow \left(\bigwedge_{r=1}^{K} \neg d_{j,r}^{\sigma(r,q)} \right) \qquad j \in [N] \wedge e_q \in \mathcal{E}^+$$

  - And, each $e_q \in \mathcal{E}^+$ must be covered by some rule:

$$\left(\bigvee_{j=1}^{N} cr_{jq} \right) \qquad e_q \in \mathcal{E}^+$$

Constraints for MinDS$_3$ II

- Each positive example must be covered by some rule
  - Define whether a rule covers some specific positive example:

$$cr_{jq} \leftrightarrow \left(\bigwedge_{r=1}^{K} \neg d_{j,r}^{\sigma(r,q)} \right) \qquad j \in [N] \land e_q \in \mathcal{E}^+$$

  - And, each $e_q \in \mathcal{E}^+$ must be covered by some rule:

$$\left(\bigvee_{j=1}^{N} cr_{jq} \right) \qquad e_q \in \mathcal{E}^+$$

- The model uses $\mathcal{O}(N \times M \times K)$ clauses and literals

Experimental setup & initial results

- 49 datasets from the PMLB repository
- Assessment of MinDS_1, MinDS_2 and MP92, w/ and w/o SBPs [IPNM18]
  - A basic model MP92 developed in the 90s [KKRR92]
  - We devised SBPs for the MinDS and the MP92 models
- Comparison with (state of the art) IDS [LBL16]
  - Heuristic approach, using smooth local search
  - Default settings & additional settings
- All experiments on an Intel Xeon E5-2630 2.60GHz processor with 64GB of memory, running Ubuntu Linux
  - Timeout of 600s and memout of 10GB

Experimental setup & initial results

- 49 datasets from the PMLB repository
- Assessment of $MinDS_1$, $MinDS_2$ and MP92, w/ and w/o SBPs [IPNM18]
  - A basic model MP92 developed in the 90s [KKRR92]
  - We devised SBPs for the MinDS and the MP92 models
- Comparison with (state of the art) IDS [LBL16]
  - Heuristic approach, using smooth local search
  - Default settings & additional settings
- All experiments on an Intel Xeon E5-2630 2.60GHz processor with 64GB of memory, running Ubuntu Linux
  - Timeout of 600s and memout of 10GB

MP92	MP92+SBP	$MinDS_2$	$MinDS_2$+SBP	$MinDS_1$	$MinDS_1$+SBP	IDS-supp0.2	IDS-supp0.5
42	45	42	45	6	6	0	2

Experimental setup & initial results

- 49 datasets from the PMLB repository
- Assessment of MinDS_1, MinDS_2 and MP92, w/ and w/o SBPs [IPNM18]
  - A basic model MP92 developed in the 90s [KKRR92]
  - We devised SBPs for the MinDS and the MP92 models
- Comparison with (state of the art) IDS [LBL16]
  - Heuristic approach, using smooth local search
  - Default settings & additional settings
- All experiments on an Intel Xeon E5-2630 2.60GHz processor with 64GB of memory, running Ubuntu Linux
  - Timeout of 600s and memout of 10GB

MP92	MP92+SBP	MinDS_2	MinDS_2+SBP	MinDS_1	MinDS_1+SBP	IDS-supp0.2	IDS-supp0.5
42	45	42	45	6	6	0	2

- There are recent improvements [YISB20]

Outline

Propositional encodings for DTs

- Proposed tight encoding for computing smallest decision tree [NIPM18]
  - Encoding also serves to **pick** the structure of the binary tree

- Encoding much tighter (and more general) than earlier work [BHO09]

SAT	Weather	Mouse	Cancer	Car	Income
DT2*	27K	3.5M	92G	842M	354G
DT1	190K	1.2M	5.2M	4.1M	1.2G

- Several recent alternative proposals [ANS20b, VNP+20, HSHH20, JM20, ANS20a, Ave20, HRS19, VZ19]
  - Several approaches outperform our work

Questions for part 4?

Part 5

(Comments on) Robustness

Validating robustness in NNs

[HKWW17, KBD+17, SGM+18, KHI+19]

- Goal: prove properties of ML models
  - Some target objective is satisfied
  - Some bad state is not reached
  - Small-distance adversarial examples are not observed

Validating robustness in NNs

[HKWW17, KBD+17, SGM+18, KHI+19]

- Goal: prove properties of ML models
  - Some target objective is satisfied
  - Some bad state is not reached
  - Small-distance adversarial examples are not observed
- Tradeoffs: soundness vs. completeness vs. both

Validating robustness in NNs

[HKWW17, KBD+17, SGM+18, KHI+19]

- Goal: prove properties of ML models
    - Some target objective is satisfied
    - Some bad state is not reached
    - Small-distance adversarial examples are not observed
- Tradeoffs: soundness vs. completeness vs. both
- Example approach: [KBD+17, KHI+19]
    - Logic/constraint-based encoding of ML models
    - Dedicated engine to reason about NNs: Reluplex

Conclusions

- Overview of (our) work at intersection of AR & ML
    1. Explainability
    2. Learning (interpretable models)
    3. Fairness
    4. Robustness

Conclusions

- Overview of (our) work at intersection of AR & ML
    1. Explainability
    2. Learning (interpretable models)
    3. Fairness
    4. Robustness
- Work offers (often viable) alternative to heuristic-based solutions
    - Fascinating range of research topics
    - Exploiting formal methods in offering much-needed rigor to the emerging field of ML

Conclusions

- Overview of (our) work at intersection of AR & ML
    1. Explainability
    2. Learning (interpretable models)
    3. Fairness
    4. Robustness
- Work offers (often viable) alternative to heuristic-based solutions
    - Fascinating range of research topics
    - Exploiting formal methods in offering much-needed rigor to the emerging field of ML
- Many challenges lie ahead:
    - Scalability, scalability, ... (often a perception, but ...)
    - Adoption, adoption, ... (evidence suggests no alternative, but ...)

Conclusions

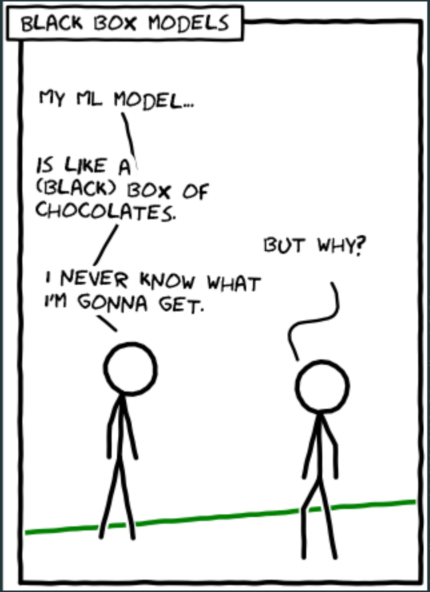
- Overview of (our) work at intersection of AR & ML
  1. Explainability
  2. Learning (interpretable models)
  3. Fairness
  4. Robustness
- Work offers (often viable) alternative to heuristic-based solutions
  - Fascinating range of research topics
  - Exploiting formal methods in offering much-needed rigor to the emerging field of ML
- Many challenges lie ahead:
  - Scalability, scalability, ... (often a perception, but ...)
  - Adoption, adoption, ... (evidence suggests no alternative, but ...)
- Our remit @ ANITI:

  To explain, to verify & to learn ML models

  with guarantees of rigor, by using AR tools & techniques

Questions?

Acknowledgment: joint work with M. Cooper, T. Gerspacher, E. Hebrard, A. Ignatiev, I. Izza, N. Narodytska, N. Asher, F. Pereira, M. Siala, et al.



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

References i

[Alp14] Ethem Alpaydin.
Introduction to machine learning.
MIT press, 2014.

[ANS20a] Gaël Aglin, Siegfried Nijssen, and Pierre Schaus.
Learning optimal decision trees using caching branch-and-bound search.
In *AAAI*, pages 3146–3153, 2020.

[ANS20b] Gaël Aglin, Siegfried Nijssen, and Pierre Schaus.
PyDL8.5: a library for learning optimal decision trees.
In *IJCAI*, pages 5222–5224, 2020.

[Ave20] Florent Avellaneda.
Efficient inference of optimal decision trees.
In *AAAI*, pages 3195–3202, 2020.

[BHO09] Christian Bessiere, Emmanuel Hebrard, and Barry O'Sullivan.
Minimising decision tree size as combinatorial optimisation.
In *CP*, pages 173–187, 2009.

[Dar20] Adnan Darwiche.
Three modern roles for logic in AI.
In *PODS*, pages 229–243, 2020.

[dBLSS20] Guy Van den Broeck, Anton Lykov, Maximilian Schleich, and Dan Suciu.
On the tractability of SHAP explanations.
CoRR, abs/2009.08634, 2020.

[DH20] Adnan Darwiche and Auguste Hirth.
On the reasons behind decisions.
In *ECAI*, pages 712–720, 2020.

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

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

[HKWW17] Xiaowei Huang, Marta Kwiatkowska, Sen Wang, and Min Wu.
Safety verification of deep neural networks.
In *CAV*, pages 3–29, 2017.

[HRS19] Xiyang Hu, Cynthia Rudin, and Margo Seltzer.
Optimal sparse decision trees.
In *NeurIPS*, pages 7265–7273, 2019.

References iii

[HSHH20] Hao Hu, Mohamed Siala, Emmanuel Hebrard, and Marie-José Huguet.
Learning optimal decision trees with MaxSAT and its integration in adaboost.
In *IJCAI*, pages 1170–1176, 2020.

[ICS+20] Alexey Ignatiev, Martin C. Cooper, Mohamed Siala, Emmanuel Hebrard, and João Marques-Silva.
Towards formal fairness in machine learning.
In *CP*, pages 846–867, 2020.

[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.

[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.

[IPNM18] Alexey Ignatiev, Filipe Pereira, Nina Narodytska, and Joao Marques-Silva.
A SAT-based approach to learn explainable decision sets.
In *IJCAR*, pages 627–645, 2018.

[JM20] Mikolás Janota and António Morgado.
SAT-based encodings for optimal decision trees with explicit paths.
In *SAT*, pages 501–518, 2020.

[KBD+17] Guy Katz, Clark W. Barrett, David L. Dill, Kyle Julian, and Mykel J. Kochenderfer.
Reluplex: An efficient SMT solver for verifying deep neural networks.
In *CAV*, pages 97–117, 2017.

[KHI+19] Guy Katz, Derek A. Huang, Duligur Ibeling, Kyle Julian, Christopher Lazarus, Rachel Lim, Parth Shah, Shantanu Thakoor, Haoze Wu, Aleksandar Zeljic, David L. Dill, Mykel J. Kochenderfer, and Clark W. Barrett.
The marabou framework for verification and analysis of deep neural networks.
In *CAV*, pages 443–452, 2019.

[KKRR92] Anil P. Kamath, Narendra Karmarkar, K. G. Ramakrishnan, and Mauricio G. C. Resende.
A continuous approach to inductive inference.
Math. Program., 57:215–238, 1992.

[LBL16] Himabindu Lakkaraju, Stephen H. Bach, and Jure Leskovec.
Interpretable decision sets: A joint framework for description and prediction.
In *KDD*, pages 1675–1684, 2016.

[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.
CoRR, abs/2008.05803, 2020.
Accepted at NeurIPS'20.

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

[NIPM18] Nina Narodytska, Alexey Ignatiev, Filipe Pereira, and Joao Marques-Silva.
Learning optimal decision trees with SAT.
In *IJCAI*, pages 1362–1368, 2018.

[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.

[PM17] David Poole and Alan K. Mackworth.
Artificial Intelligence - Foundations of Computational Agents.
CUP, 2017.

[Rei87] Raymond Reiter.
A theory of diagnosis from first principles.
Artif. Intell., 32(1):57–95, 1987.

[RN10] Stuart J. Russell and Peter Norvig.
Artificial Intelligence - A Modern Approach.
Pearson Education, 2010.

[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.

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

[SCD19] Andy Shih, Arthur Choi, and Adnan Darwiche.
Compiling bayesian network classifiers into decision graphs.
In *AAAI*, pages 7966–7974, 2019.

[SGM+18] Gagandeep Singh, Timon Gehr, Matthew Mirman, Markus Püschel, and Martin T. Vechev.
Fast and effective robustness certification.
In *NeurIPS*, pages 10825–10836, 2018.

[VNP+20] Hélène Verhaeghe, Siegfried Nijssen, Gilles Pesant, Claude-Guy Quimper, and Pierre Schaus.
Learning optimal decision trees using constraint programming.
In *IJCAI*, pages 4765–4769, 2020.

[VZ19] Sicco Verwer and Yingqian Zhang.
Learning optimal classification trees using a binary linear program formulation.
In *AAAI*, pages 1625–1632, 2019.

References viii

[YISB20] Jinqiang Yu, Alexey Ignatiev, Peter J. Stuckey, and Pierre Le Bodic.
Computing optimal decision sets with SAT.
In *CP*, pages 952–970, 2020.

[Zho12] Zhi-Hua Zhou.
Ensemble methods: foundations and algorithms.
CRC press, 2012.