From Explanations to Queries: Rethinking Explainable AI with Databases

Marcelo Arenas

International Research and Industry Symposium on AI - 2025

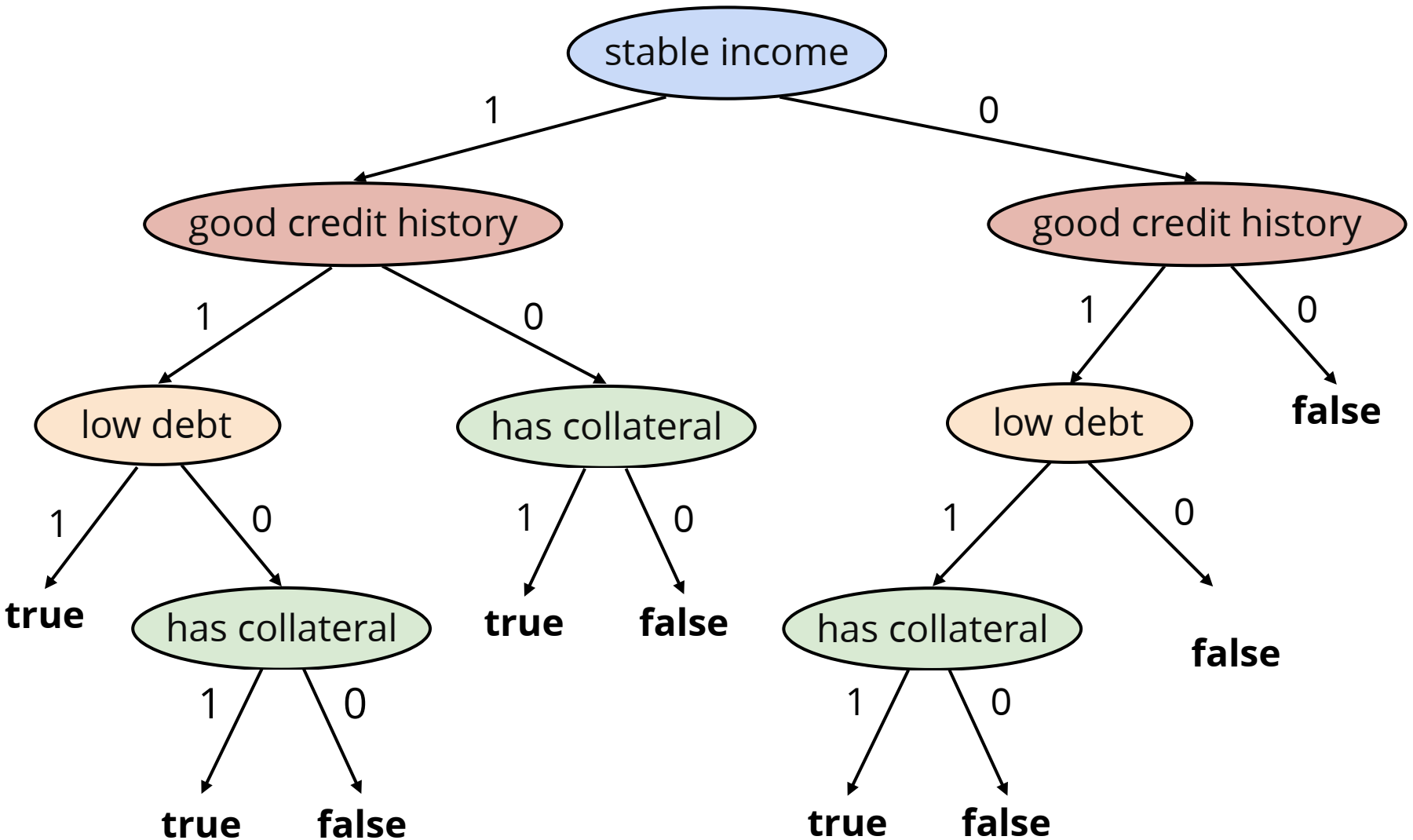
Explainable AI

- There is a great interest in developing methods to explain predictions made by ML models
- This has led to the introduction of numerous queries and scores that aim to explain the predictions of ML models

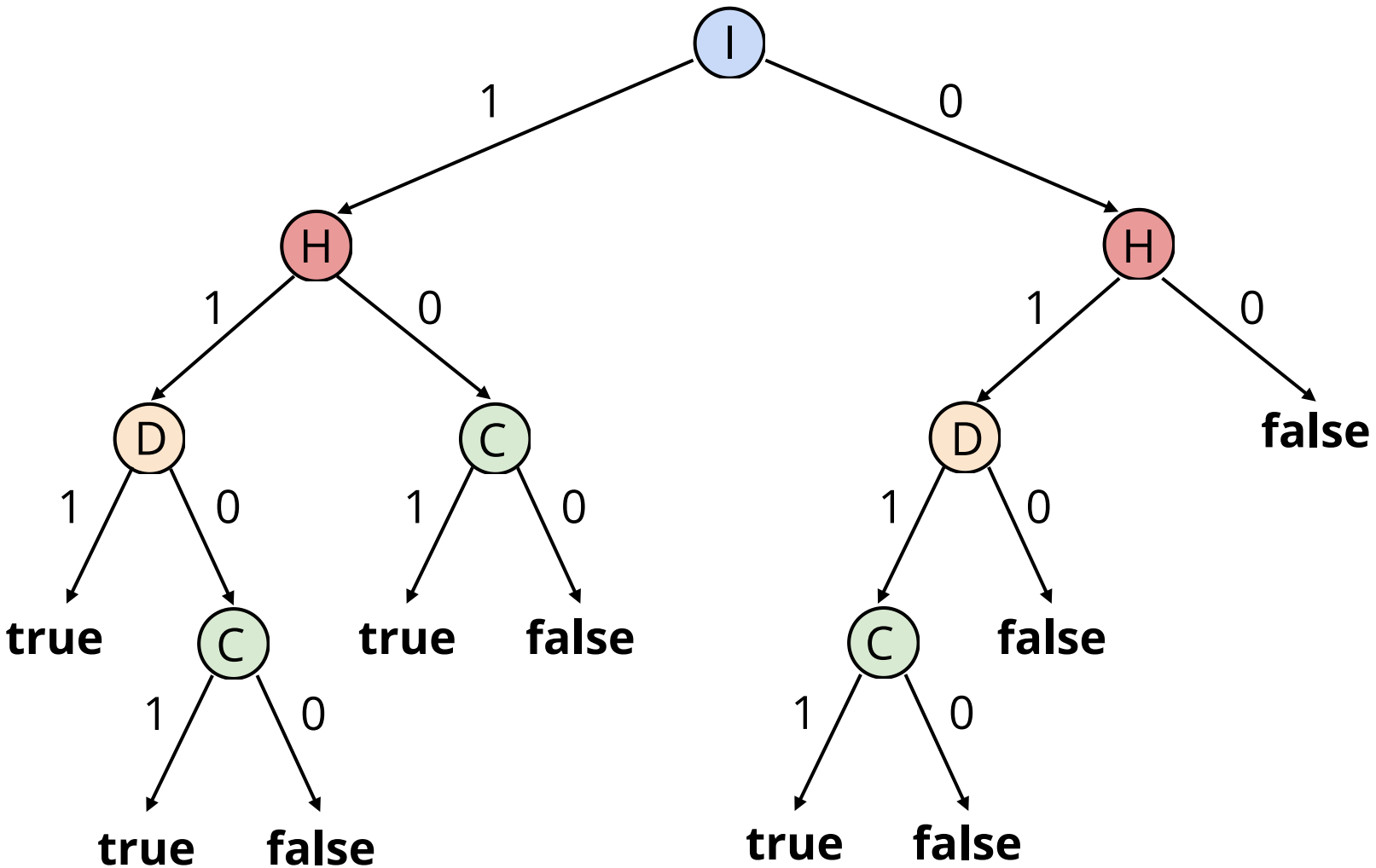
We focus here on formal explainable AI

- A growing area that focuses on computing explanations with mathematical guarantees for the predictions made by ML models
- In particular, we focus on a logic-based approach to formal explainable AI

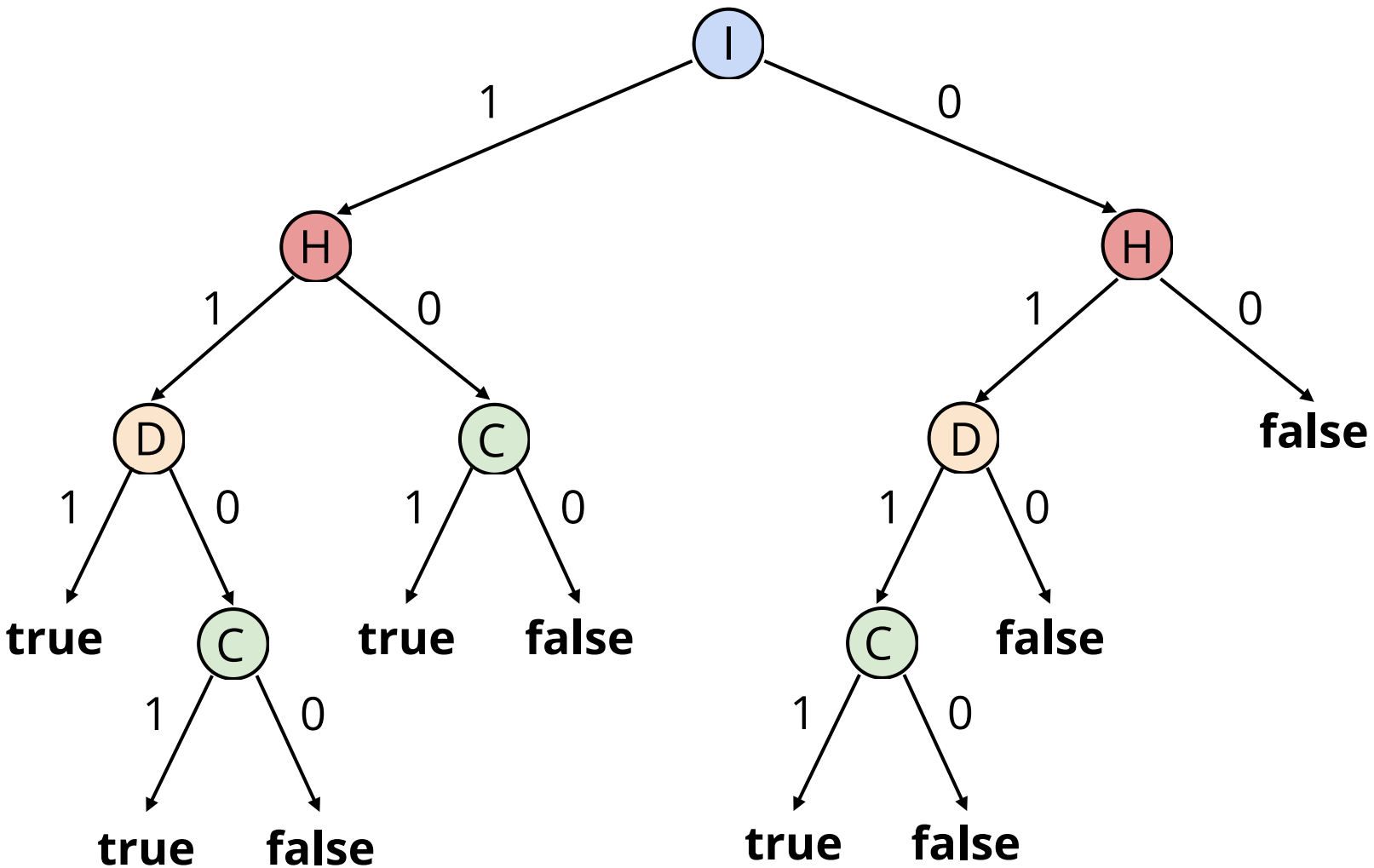
Abductive explanations



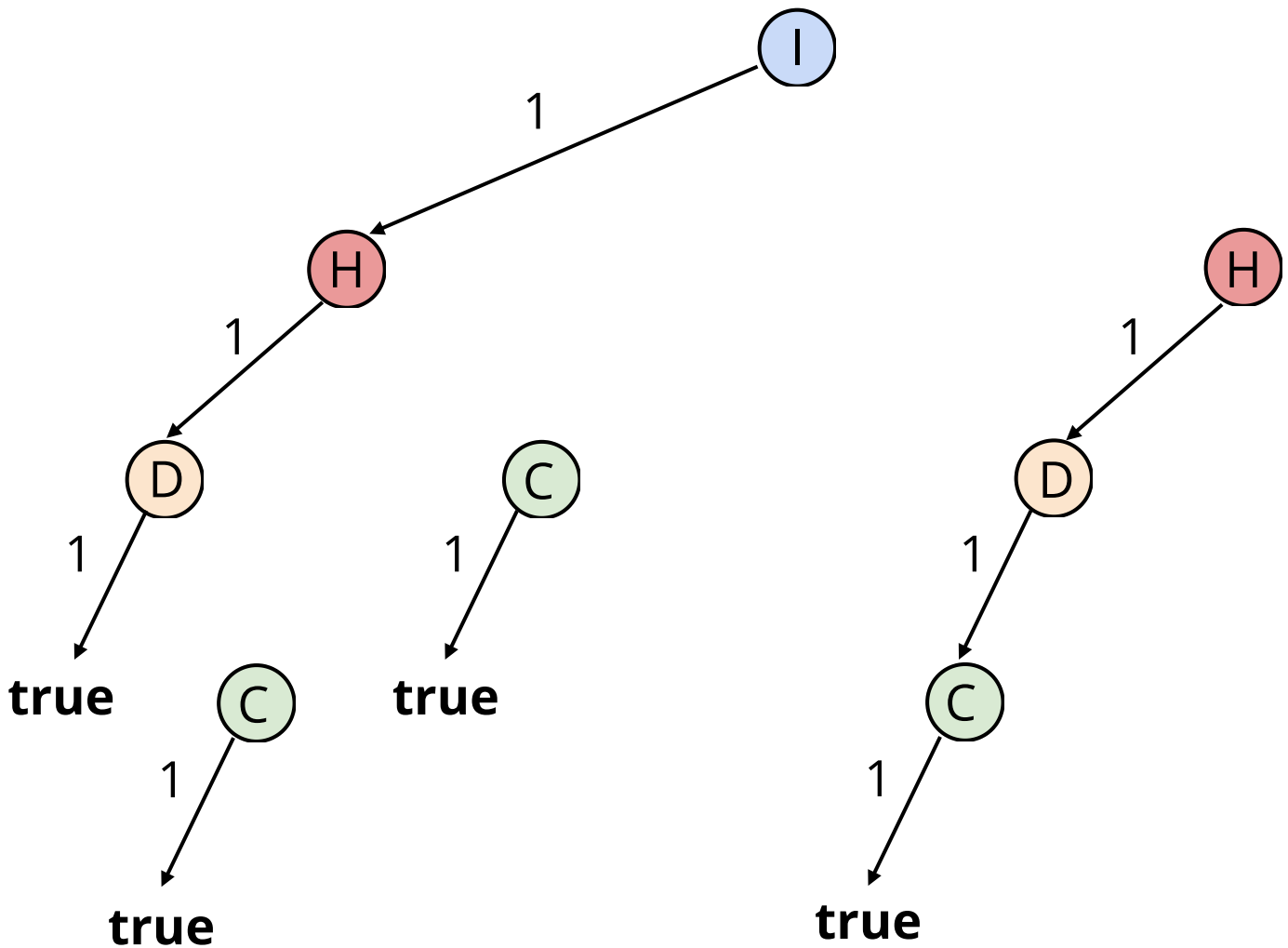
Abductive explanations



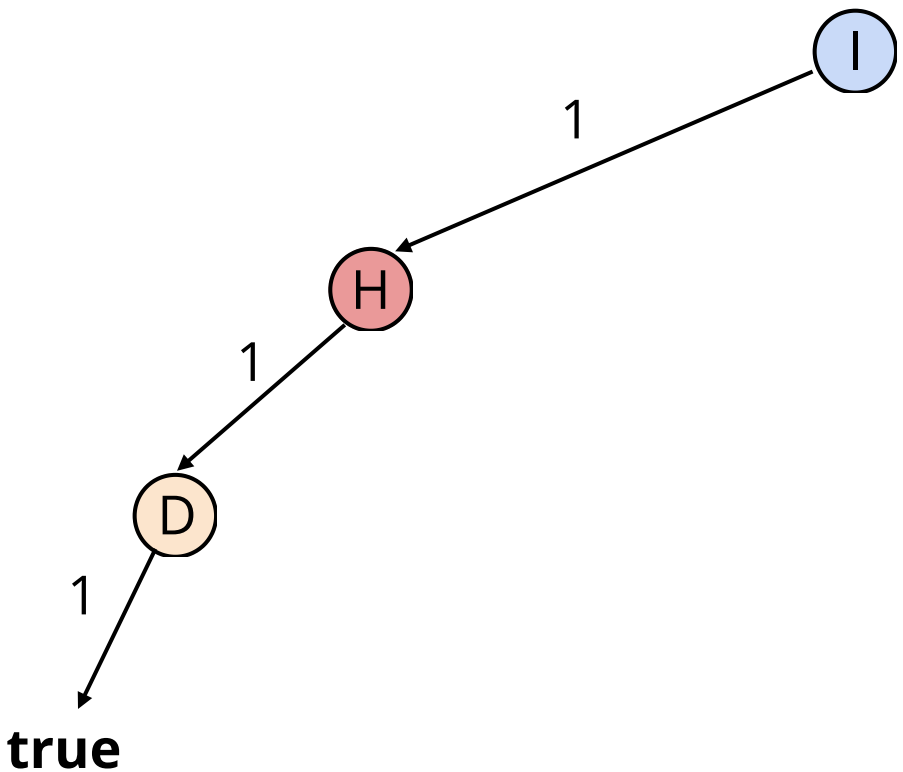
$I \to 1 \quad H \to 1 \quad D \to 1 \quad C \to 1$



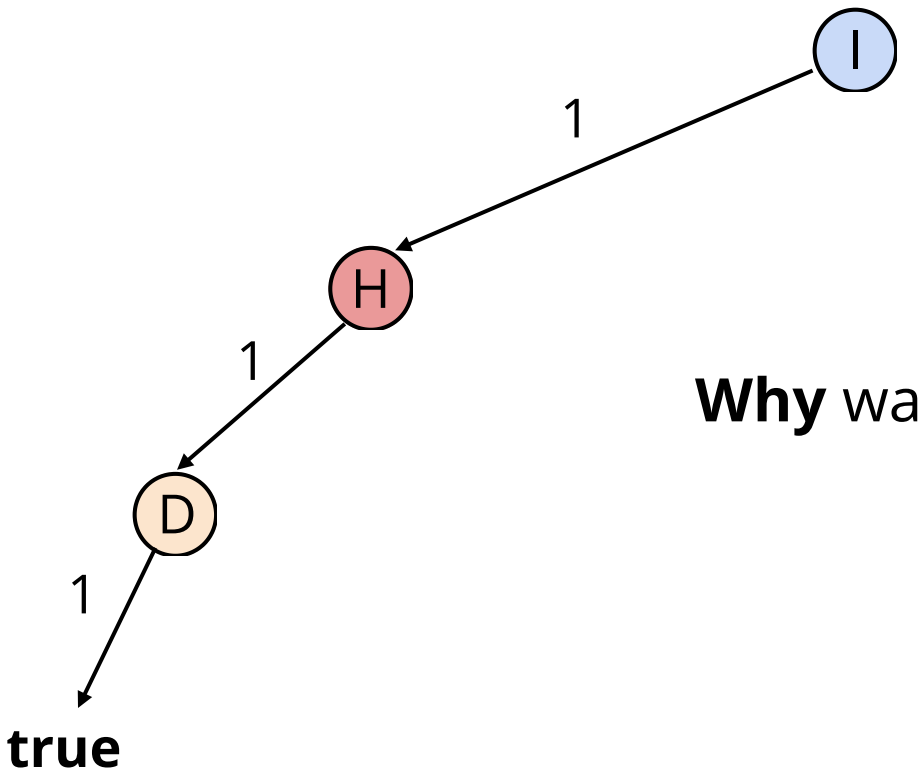
$I \to 1 \quad H \to 1 \quad D \to 1 \quad C \to 1$



$I \rightarrow 1 \quad H \rightarrow 1 \quad D \rightarrow 1 \quad C \rightarrow 1$



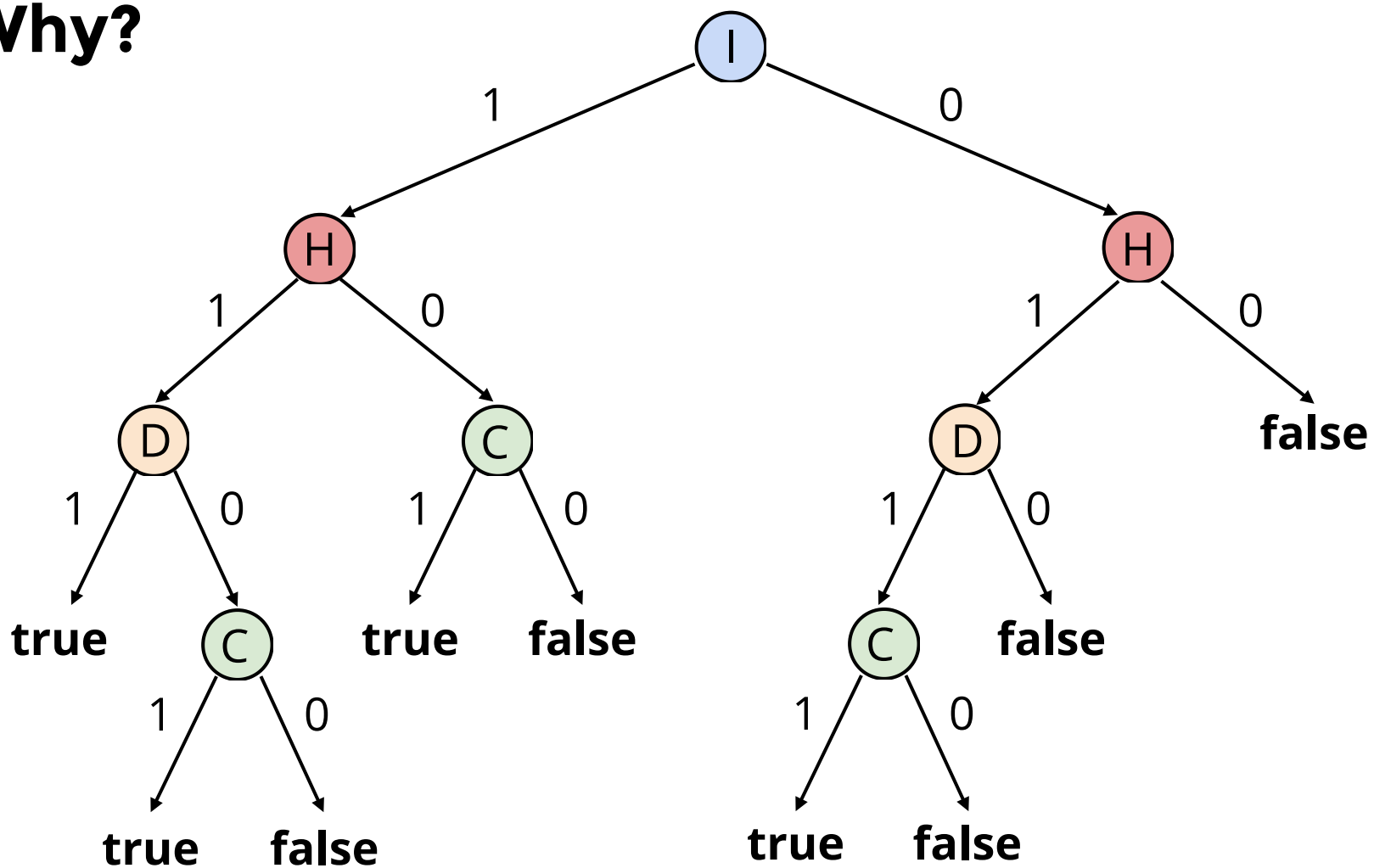
$I \rightarrow 1 \quad H \rightarrow 1 \quad D \rightarrow 1 \quad C \rightarrow 1$



Why was the credit approved?

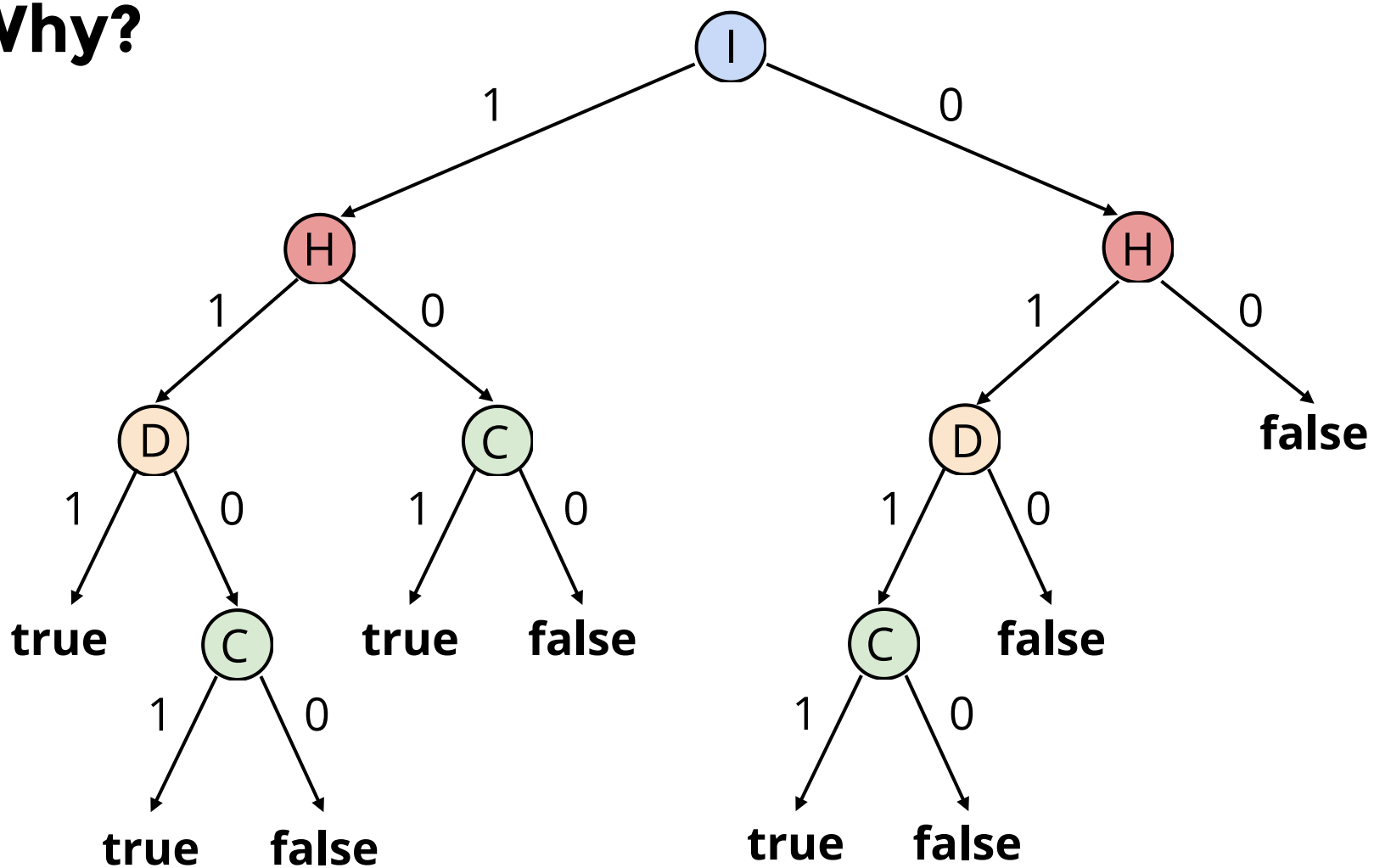
$I \to 1 \quad H \to 1 \quad D \to 1 \quad C \to 1$

Why?



$I \to 1$ $C \to 1$

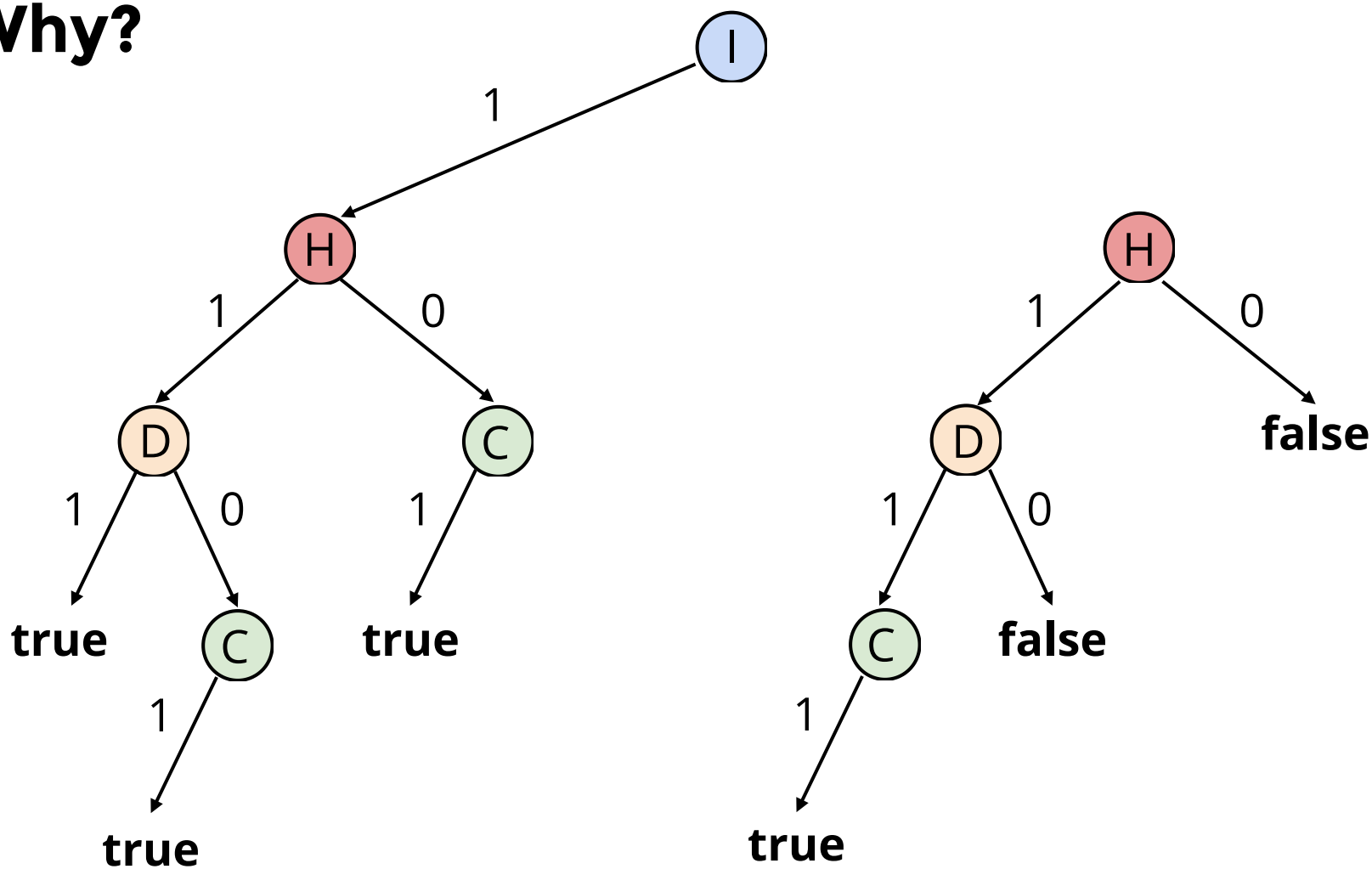
Why?



$I \to 1$

$C \to 1$

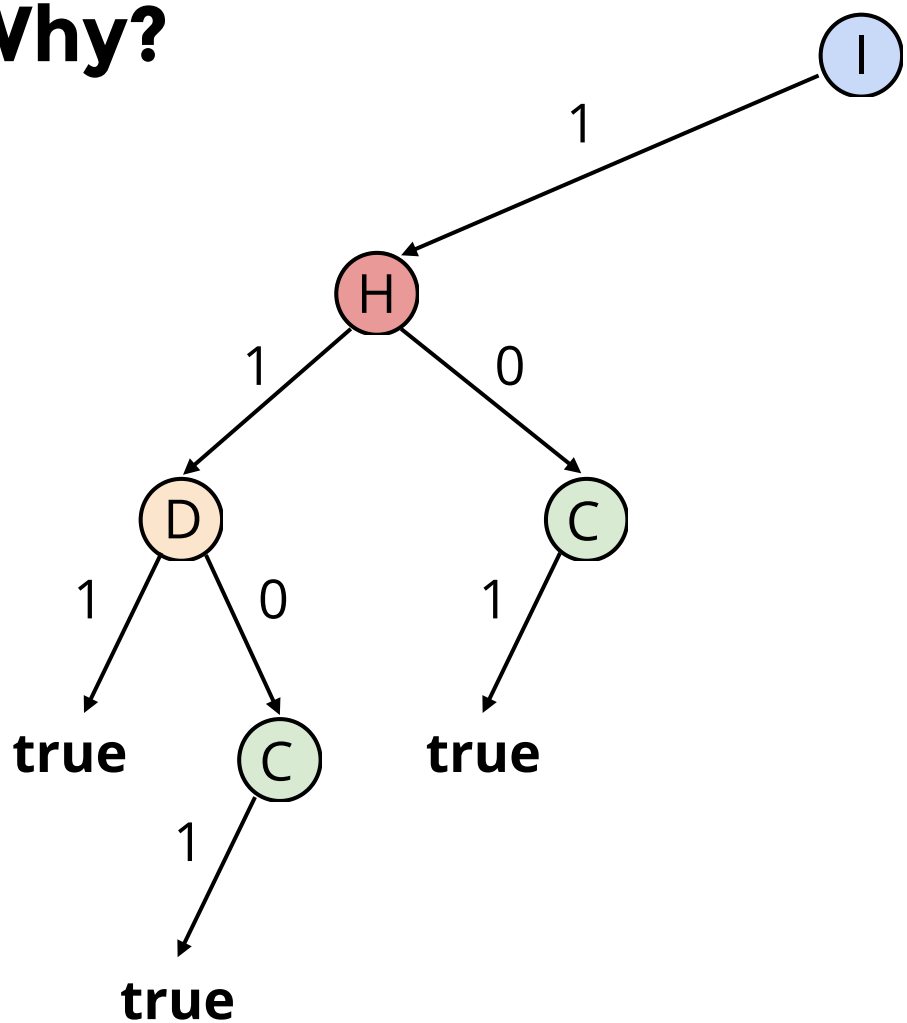
Why?



$I \rightarrow 1$

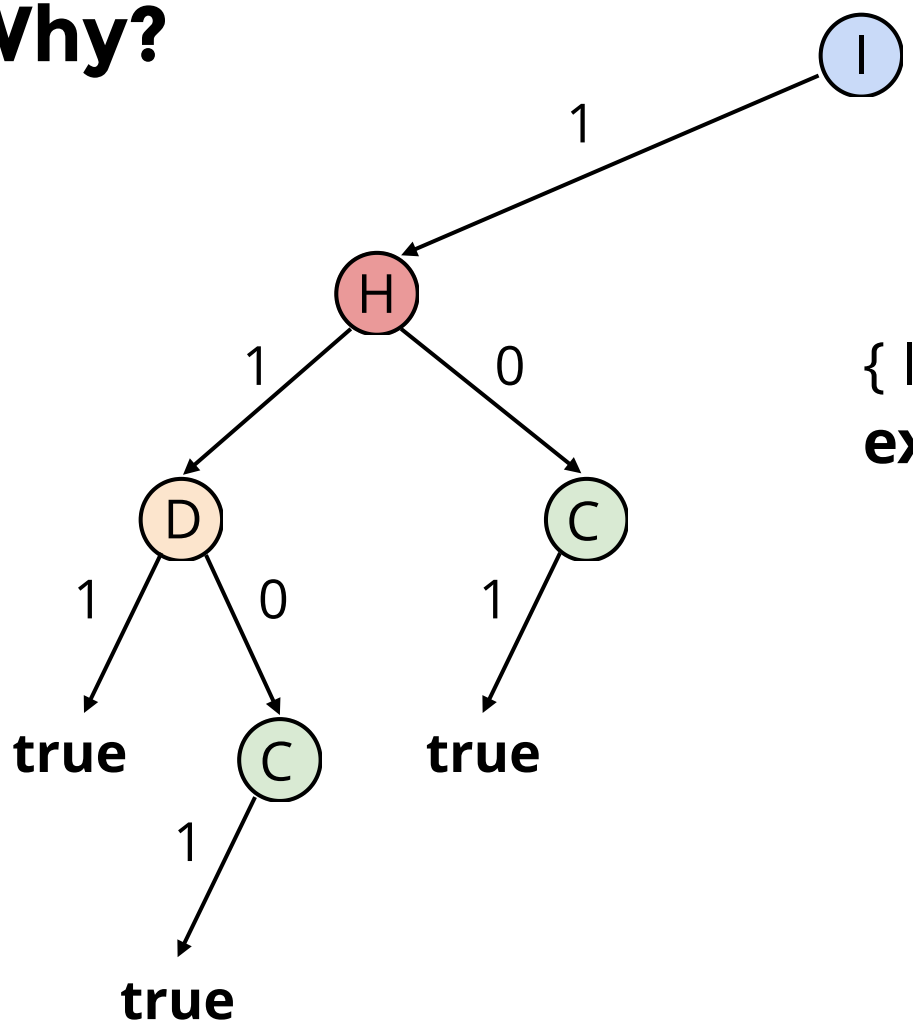
$C \rightarrow 1$

Why?



$I \rightarrow 1$ $C \rightarrow 1$

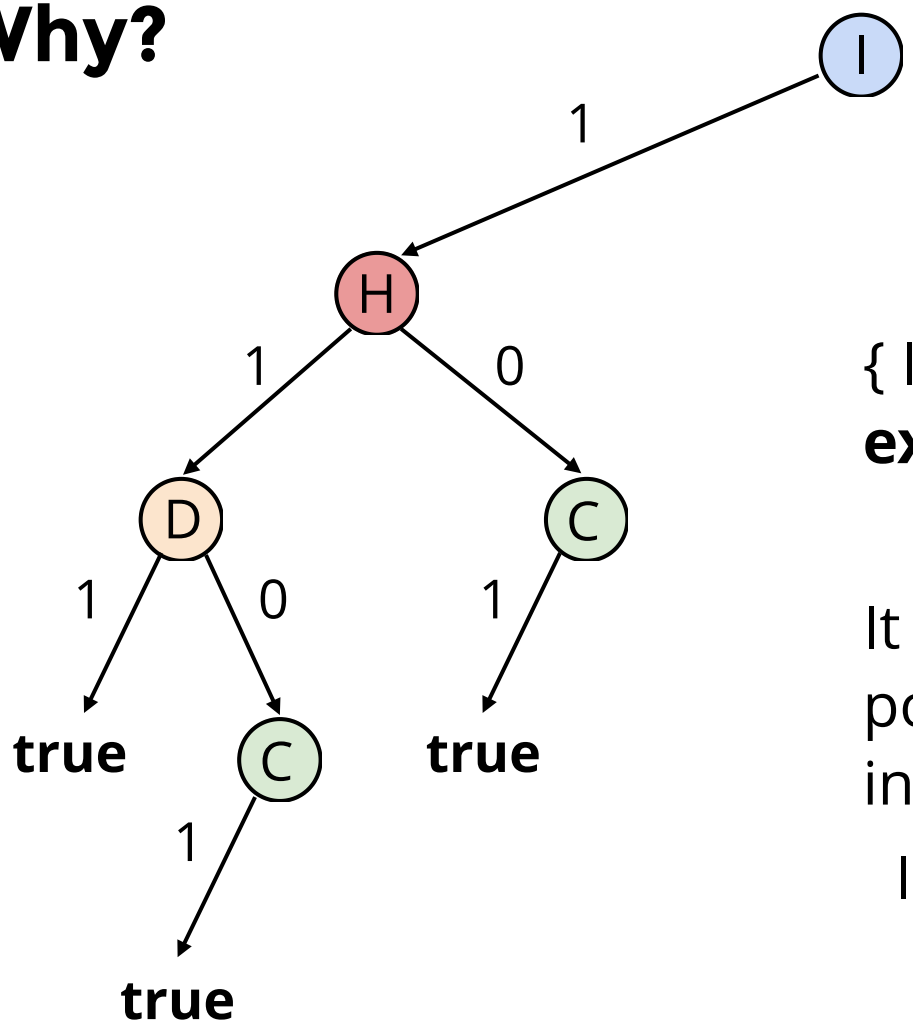
Why?



{ I, C } is an **abductive explanation**

$I \rightarrow 1$ $C \rightarrow 1$

Why?



{ I, C } is an **abductive explanation**

It is a **local** explanation for the positive classification of the instance

$I \rightarrow 1$ $H \rightarrow 1$ $D \rightarrow 1$ $C \rightarrow 1$

Formal explainability admits no silver bullet

- Explainability may require combining different notions of explanation
- It is better to think of explainability as an interactive process

Common abductive explanation

Is there a common abductive explanation for the positive classification of

$$I \to 1 \quad H \to 1 \quad D \to 1 \quad C \to 1$$

and

$$I \to 1 \quad H \to 1 \quad D \to 0 \quad C \to 1 \; ?$$

Common abductive explanation

Is there a common abductive explanation for the positive classification of

$$I \to 1 \quad H \to 1 \quad D \to 1 \quad C \to 1$$

and

$$I \to 1 \quad H \to 1 \quad D \to 0 \quad C \to 1 \; ?$$

Yes, { I, C } is an answer to the query

Distinctive abductive explanation

Is there an abductive explanation for the positive classification of

$$I \to 1 \quad H \to 1 \quad D \to 1 \quad C \to 1$$

that is not an abductive explanation for the positive classification of

$$I \to 1 \quad H \to 1 \quad D \to 0 \quad C \to 1 \; ?$$

Distinctive abductive explanation

Is there an abductive explanation for the positive classification of

$$I \to 1 \quad H \to 1 \quad D \to 1 \quad C \to 1$$

that is not an abductive explanation for the positive classification of

$$I \to 1 \quad H \to 1 \quad D \to 0 \quad C \to 1 \; ?$$

Yes, { I, H, D } is an answer to the query

Different orders

What is the abductive explanation with the **smallest** number of feature assignments for the positive classification of

$$\mathrm{I} \to 1 \quad \mathrm{H} \to 1 \quad \mathrm{D} \to 1 \quad \mathrm{C} \to 1 \; ?$$

Different orders

What is the abductive explanation with the **smallest** number of feature assignments for the positive classification of

$$\mathrm{I} \to 1 \quad \mathrm{H} \to 1 \quad \mathrm{D} \to 1 \quad \mathrm{C} \to 1 \; ?$$

What are the answers to all the previous queries if a feature is given preference over another feature?

A call for an explainability query language

- There should be a one-to-one correspondence between queries in the language and explanation notions
- The language should be declarative, with a simple syntax and semantics
- It should be possible to evaluate every query in the language efficiently

A call for an explainability query language

A desirable data complexity is P^{NP}

- Some ML models, such as decision trees, have a moderate size compared to a database
- Certain explanation tasks have an inherently high complexity
- This would enable the use of SAT solvers for query evaluation

Our goal is to develop an explainability query language that meets the previous criteria

FOIL [ABBPS21]

First-order logic defined on a suitable vocabulary to describe classification models

Representing a model

A classification model: $\mathcal{M} : \{0,1\}^n \rightarrow \{0,1\}$

Representing a model

A classification model: $\mathcal{M} : \{0,1\}^n \rightarrow \{0,1\}$

$\mathbf{e} \in \{0,1\}^n$ is an instance of dimensión n

Representing a model

A classification model: $\mathcal{M} : \{0,1\}^n \rightarrow \{0,1\}$

$\mathbf{e} \in \{0,1\}^n$ is an instance of dimensión n

$\mathcal{M}$ accepts $\mathbf{e}$ if $\mathcal{M}(\mathbf{e}) = 1$, otherwise $\mathcal{M}$ rejects $\mathbf{e}$

Representing a model

A classification model: $\mathcal{M} : \{0,1\}^n \rightarrow \{0,1\}$

$\mathbf{e} \in \{0,1\}^n$ is an instance of dimensión n

$\mathcal{M}$ accepts $\mathbf{e}$ if $\mathcal{M}(\mathbf{e}) = 1$, otherwise $\mathcal{M}$ rejects $\mathbf{e}$

$\mathbf{e} \in \{0,1,\perp\}^n$ is a partial instance of dimension n

- $\perp$ represents an unknown value

Representing a model

A model $\mathcal{M}$ of dimensión n is represented as a structure $\mathfrak{A}_{\mathcal{M}}$

Representing a model

A model $\mathcal{M}$ of dimensión n is represented as a structure $\mathfrak{A}_{\mathcal{M}}$

- The domain of $\mathfrak{A}_{\mathcal{M}}$ is $\{0, 1, \bot\}^n$
- $\mathbf{Pos(e)}$ holds if $\mathbf{e}$ is an instance and $\mathcal{M}$ accepts $\mathbf{e}$
- $\mathbf{e}_1 \subseteq \mathbf{e}_2$ holds if for every $i \in \{1, \ldots, n\}$ such that $\mathbf{e}_1[i] \neq \bot$, it holds that $\mathbf{e}_1[i] = \mathbf{e}_2[i]$

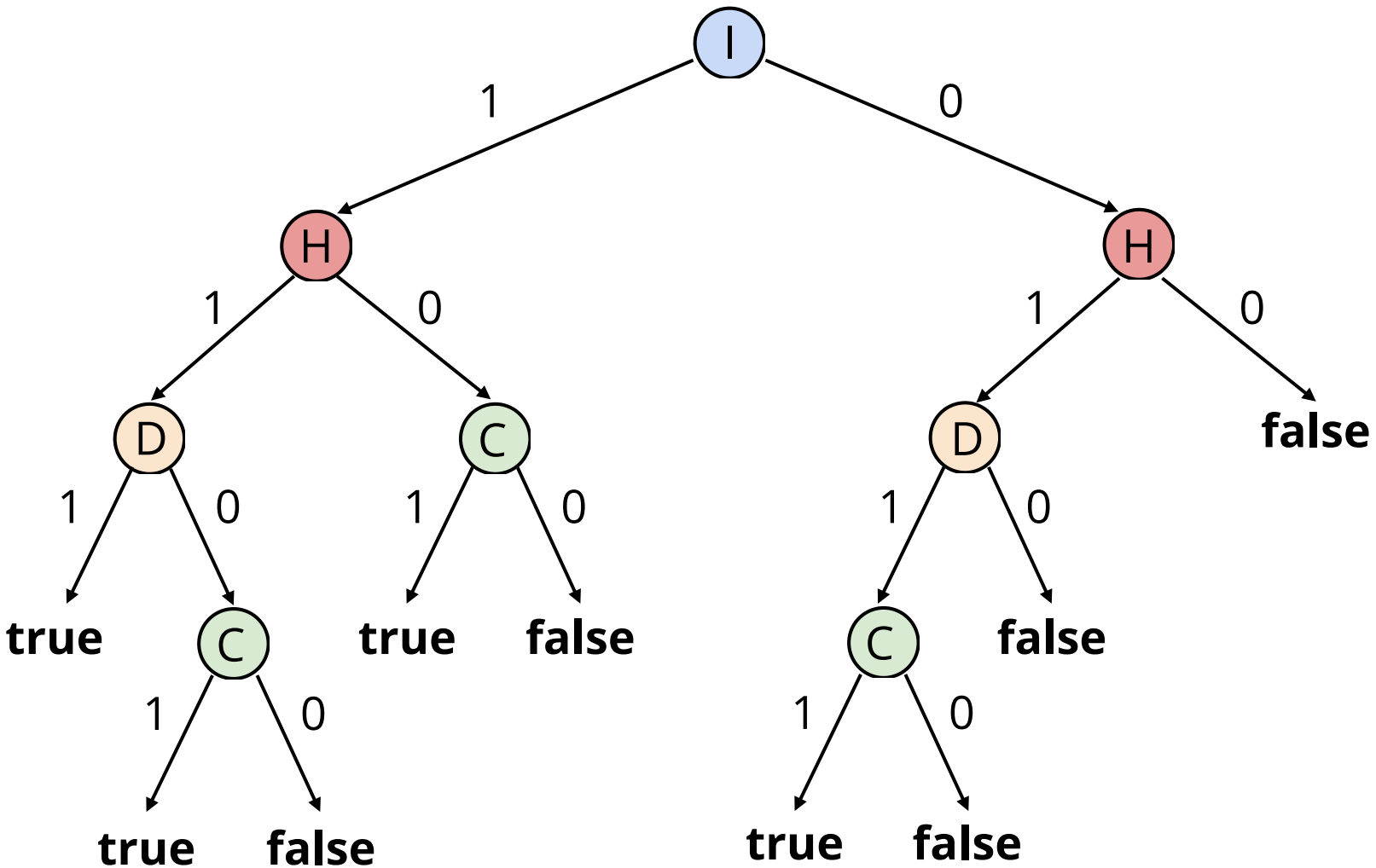
Representing a model

A model $\mathcal{M}$ of dimensión n is represented as a structure $\mathfrak{A}_{\mathcal{M}}$

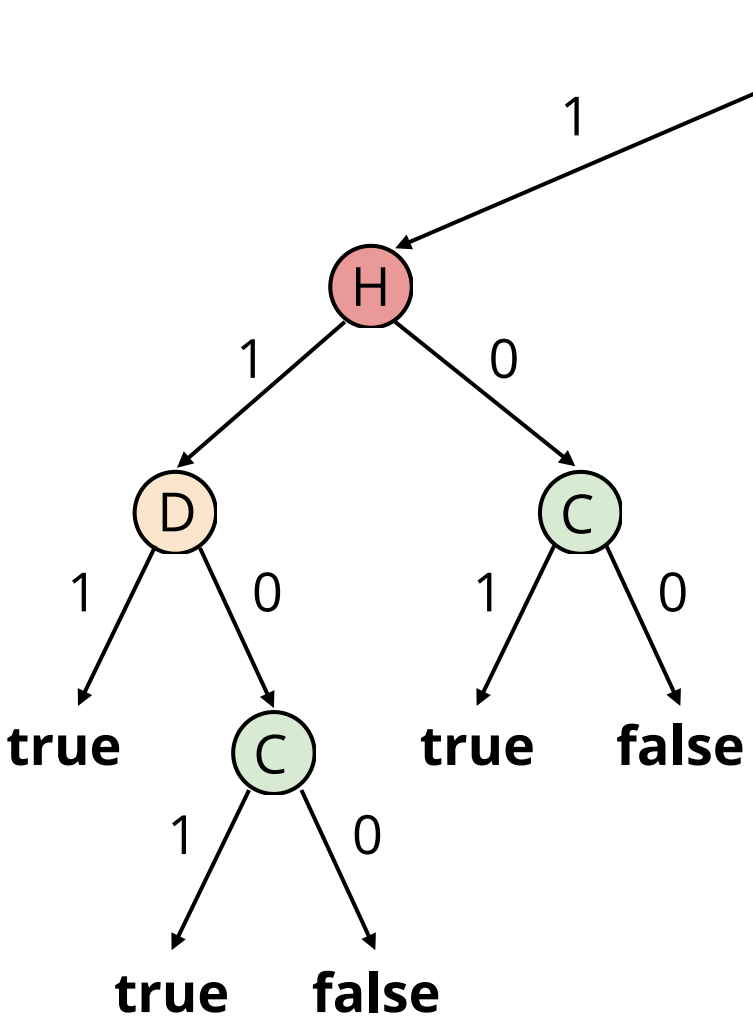
- The domain of $\mathfrak{A}_{\mathcal{M}}$ is $\{0, 1, \bot\}^n$
- $\mathbf{Pos(e)}$ holds if $\mathbf{e}$ is an instance and $\mathcal{M}$ accepts $\mathbf{e}$
- $\mathbf{e}_1 \subseteq \mathbf{e}_2$ holds if for every $i \in \{1, \ldots, n\}$ such that $\mathbf{e}_1[i] \neq \bot$, it holds that $\mathbf{e}_1[i] = \mathbf{e}_2[i]$

$$(1, \bot, 0, \bot) \subseteq (1, 0, 0, \bot) \subseteq (1, 0, 0, 1)$$

The predicate **Pos**



The predicate **Pos**



We assume some order on the features: (I, H, D, C)

Pos

$(1, 1, 1, 1)$
$(1, 1, 1, 0)$
$(1, 1, 0, 1)$
$(1, 0, 0, 1)$

. . .

Syntax and semantics of FOIL

First-order logic defined over the vocabulary $\{\mathrm{Pos}, \subseteq\}$

Syntax and semantics of FOIL

First-order logic defined over the vocabulary $\{\mathrm{Pos}, \subseteq\}$

Given a **FOIL** formula $\Phi(x_1, x_2, \ldots, x_k)$, a classification model $\mathcal{M}$ of dimensión n, and partial instances $\mathbf{e}_1$, $\mathbf{e}_2$, ..., $\mathbf{e}_k$ of dimension n

Syntax and semantics of FOIL

First-order logic defined over the vocabulary $\{\mathrm{Pos}, \subseteq\}$

Given a **FOIL** formula $\Phi(x_1, x_2, \ldots, x_k)$, a classification model $\mathcal{M}$ of dimensión n, and partial instances $\mathbf{e}_1$, $\mathbf{e}_2$, ..., $\mathbf{e}_k$ of dimension n

$$\mathcal{M} \models \Phi(\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_k)$$

$$\Longleftrightarrow$$

$$\mathfrak{A}_{\mathcal{M}} \models \Phi(\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_k)$$

(in the usual sense)

An example: abductive explanations

An example: abductive explanations

$$x \subset y \;=\; x \subseteq y \wedge \neg\, y \subseteq x$$

An example: abductive explanations

$$x \subset y \;=\; x \subseteq y \wedge \neg\, y \subseteq x$$

$$\mathrm{Inst}(x) \;=\; \forall y\,(x \subseteq y \rightarrow x = y)$$

An example: abductive explanations

$$x \subset y \;=\; x \subseteq y \wedge \neg\, y \subseteq x$$

$$\mathrm{Inst}(x) \;=\; \forall y\,(x \subseteq y \rightarrow x = y)$$

$$\mathrm{wAE}(x, y) \;=\; \mathrm{Inst}(x) \wedge y \subseteq x \;\wedge \\ \forall z\,[(\mathrm{Inst}(z) \wedge y \subseteq z) \rightarrow (\mathrm{Pos}(x) \leftrightarrow \mathrm{Pos}(z))]$$

An example: abductive explanations

$$x \subset y \;=\; x \subseteq y \wedge \neg\, y \subseteq x$$

$$\mathrm{Inst}(x) \;=\; \forall y\,(x \subseteq y \rightarrow x = y)$$

$$\mathrm{wAE}(x,y) \;=\; \mathrm{Inst}(x) \wedge y \subseteq x \;\wedge \\ \forall z\,[(\mathrm{Inst}(z) \wedge y \subseteq z) \rightarrow (\mathrm{Pos}(x) \leftrightarrow \mathrm{Pos}(z))]$$

$$\mathrm{AE}(x,y) \;=\; \mathrm{wAE}(x,y) \wedge \forall z(\mathrm{wAE}(x,z) \rightarrow \neg\, z \subset y)$$

An example: abductive explanations

Consider the order (I, H, D, C) on the features

$\mathcal{M}(\mathbf{e}) = 1$ for $\mathbf{e} = (1, 1, 1, 1)$, and $\mathbf{e}_1 = (1, \perp, \perp, 1)$ is an abductive explanation for this

An example: abductive explanations

Consider the order (I, H, D, C) on the features

$\mathcal{M}(\mathbf{e}) = 1$ for $\mathbf{e} = (1, 1, 1, 1)$, and $\mathbf{e}_1 = (1, \bot, \bot, 1)$ is an abductive explanation for this

$$\mathcal{M} \models \mathrm{AE}(\mathbf{e}, \mathbf{e}_1)$$

Common abductive explanation

Is there a common abductive explanation for the positive classification of instances $\mathbf{e} = (1, 1, 1, 1)$ and $\mathbf{e}' = (1, 1, 0, 1)$?

$$\mathrm{CAE}(x_1, x_2, y) \;=\; \mathrm{AE}(x_1, y) \wedge \mathrm{AE}(x_2, y)$$

Distinctive abductive explanation

Is there an abductive explanation for the positive classification of the instance $\mathbf{e} = (1, 1, 1, 1)$ that is not an abductive explanation for the positive classification of the instance $\mathbf{e}' = (1, 1, 0, 1)$?

$$\mathrm{DAE}(x_1, x_2, y) \;=\; \mathrm{AE}(x_1, y) \wedge \neg \mathrm{AE}(x_2, y)$$

The evaluation problem for FOIL

Assume $\Phi(x_1, \ldots, x_k)$ is a fixed **FOIL** formula and $\mathcal{C}$ is a class of classification models

$\mathbf{Eval(\Phi, \mathcal{C})}$:

- **Input:** a classification model $\mathcal{M} \in \mathcal{C}$ of dimension n and partial instances $\mathbf{e}_1, \ldots, \mathbf{e}_k$ of dimension n
- **Output:** yes if $\mathcal{M} \models \Phi(\mathbf{e}_1, \ldots, \mathbf{e}_k)$, and no otherwise

Evaluating FOIL is very hard

Theorem:

1. For every **FOIL** formula Φ, there exists $k \geq 1$ such that $\mathrm{Eval}(\Phi, \mathrm{DTree})$ is in Σ_k^{P}
2. For every $k \geq 1$, there exists a **FOIL** formula Φ such that $\mathrm{Eval}(\Phi, \mathrm{DTree})$ is Σ_k^{P}-hard

FOIL lacks counting capabilities

What is the abductive explanation with the smallest number of feature assignments for the positive classification of $(1, 1, 1, 1)$?

Such an explanation is referred to as a minimum abductive explanation

The expressiveness of FOIL

Theorem:

There is no **FOIL** formula $\mathrm{minAE}(x, y)$ such that, for every decision tree $\mathcal{T}$, instance $\mathbf{e}_1$ and partial instance $\mathbf{e}_2$:

$$\mathcal{T} \models \mathrm{minAE}(\mathbf{e}_1, \mathbf{e}_2)$$

$$\Longleftrightarrow$$

$\mathbf{e}_2$ is a minimum abductive explanation for $\mathbf{e}_1$ over $\mathcal{T}$

How can these limitations be overcome?

- Extend **FOIL** vocabulary to express missing notions of explanation
- Use restricted forms of quantification
- Identify fragments of **FOIL** that can be evaluated efficiently

How can these limitations be overcome?

- Extend **FOIL** vocabulary to express missing notions of explanation
- Use restricted forms of quantification
- Identify fragments of **FOIL** that can be evaluated efficiently

How can these limitations be overcome?

- Extend **FOIL** vocabulary to express missing notions of explanation
- Use restricted forms of quantification
- Identify fragments of **FOIL** that can be evaluated efficiently

We follow a principled approach

Extending the vocabulary

We need a predicate to encode orders based on cardinalities

Extending the vocabulary

We need a predicate to encode orders based on cardinalities

Given partial instances $\mathbf{e}_1, \mathbf{e}_2$ of dimension n:

$$\mathbf{e}_1 \preceq \mathbf{e}_2$$

$$\Longleftrightarrow$$

$$|\{i \in \{1, \ldots n\} \mid \mathbf{e}_1[i] = \bot\}| \geq |\{i \in \{1, \ldots n\} \mid \mathbf{e}_2[i] = \bot\}|$$

Minimum abductive explanations

$$x \prec y \;=\; x \preceq y \wedge \neg\, y \preceq x$$

$$\mathrm{Inst}(x) \;=\; \forall y\,(x \subseteq y \rightarrow x = y)$$

$$\mathrm{wAE}(x,y) \;=\; \mathrm{Inst}(x) \wedge y \subseteq x \;\wedge \\ \forall z\,[(\mathrm{Inst}(z) \wedge y \subseteq z) \rightarrow (\mathrm{Pos}(x) \leftrightarrow \mathrm{Pos}(z))]$$

$$\mathrm{minAE}(x,y) \;=\; \mathrm{wAE}(x,y) \wedge \forall z(\mathrm{wAE}(x,z) \rightarrow \neg\, z \prec y)$$

But how many more predicates do we need to include?

All the *order* predicates needed in our formalism can be expressed as first-order queries over $\{\subseteq, \preceq\}$

But how many more predicates do we need to include?

Theorem: For every first-order formula Φ defined over $\{\subseteq, \preceq\}$ and class $\mathcal{C}$ of classification models, $\mathrm{Eval}(\Phi, \mathcal{C})$ can be solved in polynomial time

Proof: The predicate **Pos** does not need to be considered, so there is a single model for each dimension n. Then we use Ehrenfeucht–Fraïssé games to prove the theorem

An extended notion of atomic formula

Atomic formulas: the set of first-order formulas defined over $\{\subseteq, \preceq\}$

A restricted form of quantification

We would like to use (partial) instances and features in the logic

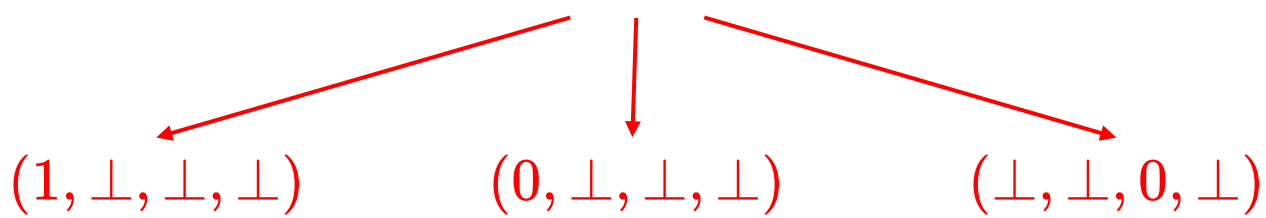
- In particular, we would like to quantify over features

A restricted form of quantification

Consider the following predicates:

$$L_0(x) \;=\; \forall y\,(x \subseteq y)$$

$$L_1(x) \;=\; \exists y\,\big(L_0(y) \wedge y \subset x \wedge \neg\exists z\,(y \subset z \wedge z \subset x)\big)$$



A restricted form of quantification

Consider the following predicates:

$$L_0(x) \;=\; \forall y\,(x \subseteq y)$$

$$L_1(x) \;=\; \exists y\,\big(L_0(y) \wedge y \subset x \wedge \neg\exists z\,(y \subset z \wedge z \subset x)\big)$$

Atomic formulas

Guarded quantification

An efficient form of quantification is obtained by considering the notion of *guard*:

$$\exists x\,(L_1(x) \wedge \Phi) \qquad\qquad \forall x\,(L_1(x) \rightarrow \Phi)$$

We are quantifying over features

A last ingredient

The predicate **Pos** was included to encode classification models

- It does not distinguish between partial instances with at least one undefined feature

A last ingredient

We replace **Pos** with **AllPos** and **AllNeg:**

$$\mathrm{AllPos}(x) = \forall y \left((x \subseteq y \wedge \mathrm{Inst}(y)) \rightarrow \mathrm{Pos}(y) \right)$$

$$\mathrm{AllNeg}(x) = \forall y \left((x \subseteq y \wedge \mathrm{Inst}(y)) \rightarrow \neg\mathrm{Pos}(y) \right)$$

These predicates coincide for an instance e:

$$\mathrm{Pos}(e) \Leftrightarrow \mathrm{AllPos}(e) \Leftrightarrow \neg\mathrm{AllNeg}(e)$$

$$\neg\mathrm{Pos}(e) \Leftrightarrow \mathrm{AllNeg}(e) \Leftrightarrow \neg\mathrm{AllPos}(e)$$

A last ingredient

Predicates **AllPos** and **AllNeg** are defined for every classification model, but can be computed in polynomial time for certain classes of models:

- Decision tress, OBDDs, FBDDs, d-DNNF circuits

The logic GD-FOIL

GD-FOIL is recursive defined as follows

1. Every atomic formula is a **GD-FOIL** formula
2. $\mathrm{AllPos}(x)$ and $\mathrm{AllNeg}(x)$ are **GD-FOIL** formulas
3. A Boolean combination of **GD-FOIL** formulas is a **GD-FOIL** formula
4. If Φ is a **GD-FOIL** formula, then $\exists x\,(L_1(x) \wedge \Phi)$ and $\forall x\,(L_1(x) \rightarrow \Phi)$ are **GD-FOIL** formulas

The logic GD-FOIL

Theorem: If **AllPos** and **AllNeg** can be computed in polynomial time for a class of models $\mathcal{C}$, then $\mathrm{Eval}(\Phi, \mathcal{C})$ can be solved in polynomial time for every **GD-FOIL** formula Φ

The logic Q-GD-FOIL

Q-GD-FOIL is recursively defined as:

- Every **GD-FOIL** formula is a **Q-GD-FOIL** formula
- If φ is a **GD-FOIL** formula, then $\exists x_1 \cdots \exists x_k \, \varphi$ and $\forall x_1 \cdots \forall x_k \, \varphi$ are **Q-GD-FOIL** formulas
- A Boolean combination of **Q-GD-FOIL** formulas is a **Q-GD-FOIL** formula

What is the expressiveness of Q-GD-FOIL?

Common notions of explanation are expressible in **Q-GD-FOIL**:

- Abductive, minimum abductive, contrastive, minimum contrastive, minimum change required, maximum changed allowed, global necessity, ...
- Distance-restricted abductive and contrastive explanations for the norm $\| \ \|_1$ [IHMPIM24]
- They can be defined for arbitrary notions of order that can consider preferences over features

Minimum change required

What is the smallest set of features that must be changed in an instance to change its classification?

Minimum change required

What is the smallest set of features that must be changed in an instance to change its classification?

$$\mathrm{GLB}(x, y, z) = (z \subseteq x \wedge z \subseteq y) \wedge \forall w \left((w \subseteq x \wedge w \subseteq y) \rightarrow w \subseteq z\right)$$

Minimum change required

What is the smallest set of features that must be changed in an instance to change its classification?

$$\mathrm{GLB}(x, y, z) = (z \subseteq x \wedge z \subseteq y) \wedge \forall w \left((w \subseteq x \wedge w \subseteq y) \rightarrow w \subseteq z\right)$$

$$\mathrm{LEH}(x, y, z) = \mathrm{Inst}(x) \wedge \mathrm{Inst}(y) \wedge \mathrm{Inst}(z) \wedge \exists w_1 \exists w_2 \left(\mathrm{GLB}(x, y, w_1) \wedge \mathrm{GLB}(x, z, w_2) \wedge w_2 \preceq w_1\right)$$

Minimum change required

What is the smallest set of features that must be changed in an instance to change its classification?

Atomic formulas

$$\mathrm{GLB}(x, y, z) = (z \subseteq x \wedge z \subseteq y) \wedge \forall w\,((w \subseteq x \wedge w \subseteq y) \rightarrow w \subseteq z)$$

$$\mathrm{LEH}(x, y, z) = \mathrm{Inst}(x) \wedge \mathrm{Inst}(y) \wedge \mathrm{Inst}(z) \wedge \exists w_1 \exists w_2\,(\mathrm{GLB}(x, y, w_1) \wedge \mathrm{GLB}(x, z, w_2) \wedge w_2 \preceq w_1)$$

Minimum change required

What is the smallest set of features that must be changed in an instance to change its classification?

$$\begin{aligned}\mathrm{MinCR}(x, y) = {} & \mathrm{Inst}(x) \wedge \mathrm{Inst}(y) \wedge \\ & \neg(\mathrm{AllPos}(x) \leftrightarrow \ \mathrm{AllPos}(y)) \wedge \\ & \forall z \left[\big(\mathrm{Inst}(z) \wedge \neg(\mathrm{AllPos}(x) \leftrightarrow \mathrm{AllPos}(z)) \big) \rightarrow \mathrm{LEH}(x, y, z) \right]\end{aligned}$$

Minimum change required

What is the smallest set of features that must be changed in an instance to change its classification?

$$\begin{aligned}\mathrm{MinCR}(x,y) = {} & \mathrm{Inst}(x) \wedge \mathrm{Inst}(y) \wedge \\ & \neg(\mathrm{AllPos}(x) \leftrightarrow \ \mathrm{AllPos}(y)) \wedge \\ & \forall z \left[\left(\mathrm{Inst}(z) \wedge \neg(\mathrm{AllPos}(x) \leftrightarrow \mathrm{AllPos}(z)) \right) \rightarrow \mathrm{LEH}(x,y,z) \right]\end{aligned}$$

Q-GD-FOIL formula

Global necessity of a feature

Is there a feature assignment that is necessary to obtain a positive classification?

$$L_0(x) \;=\; \forall y\,(x \subseteq y)$$

$$L_1(x) \;=\; \exists y\,\big(L_0(y) \wedge y \subset x \wedge \neg\exists z\,(y \subset z \wedge z \subset x)\big)$$

$$\mathrm{GN}(x) \;= L_1(x) \wedge \forall y\,\big((\mathrm{Inst}(y) \wedge \mathrm{AllPos}(y)) \rightarrow x \subseteq y\big)$$

The complexity of Q-GD-FOIL

Theorem:

1. For every **Q-GD-FOIL** formula Φ, there exists $k \geq 1$ such that $\mathrm{Eval}(\Phi, \mathrm{d\text{-}DNNF})$ is in BH_k
2. For every $k \geq 1$, there exists a **Q-GD-FOIL** formula Φ such that $\mathrm{Eval}(\Phi, \mathrm{DTree})$ is BH_k-hard

What about in practice?

We want to both **verify** and **compute** explanations

For the case of computation, we propose the logic **OPT-GD-FOIL** that extends of **GD-FOIL** with a min operator

- min is defined over an arbitrary partial order

What about in practice?

The complexity of the evaluation problem for **OPT-GD-FOIL** is $\mathrm{FP}^{\mathrm{NP}}$

What about in practice?

The complexity of the evaluation problem for **OPT-GD-FOIL** is $\mathrm{FP}^{\mathrm{NP}}$

Repository with implementation:

https://github.com/jtcaraball/goexpdt

Repository with experimental evaluations:

https://github.com/jtcaraball/goexpdt-experiments

Concluding remarks

What constitutes a user-friendly explainability query language?

- How should an explanation be presented to the user?
- What is the right level of detail that has to be provided to different users? How can this level of detail be specified?
- How can it be proven that such an explanation is trustworthy?

Concluding remarks

How can probabilities be incorporated into this framework?

- A probability distribution on the possible values of features, and a probabilistic classifier

Probabilistic circuits seem to be the right model for this

- A natural and robust generalization of Boolean circuits, with many well-understood properties

Thanks!

Bibliography

[ABBPS21] M. Arenas, D. Báez, P. Barceló, J. Pérez, B. Subercaseaux: Foundations of Symbolic Languages for Model Interpretability. NeurIPS 2021

[ABBCS24] M. Arenas, P. Barceló, D. Bustamante, J. Caraball, B. Subercaseaux: A Uniform Language to Explain Decision Trees. KR 2024

[IHMPIM24] Y. Izza, X. Huang, A. Morgado, J. Planes, A. Ignatiev, J. Marques-Silva: Distance-Restricted Explanations: Theoretical Underpinnings & Efficient Implementation. KR 2024

[MS22] J. Marques-Silva: Logic-Based Explainability in Machine Learning. RW 2022