SAT-based Approach for Learning Optimal Decision Trees with Non-Binary Features

POUYA SHATI, ELDAN COHEN, SHEILA MCILRAITH

UNIVERSITY OF TORONTO

CP 2021

SEPTEMBER 13, 2021

Overview

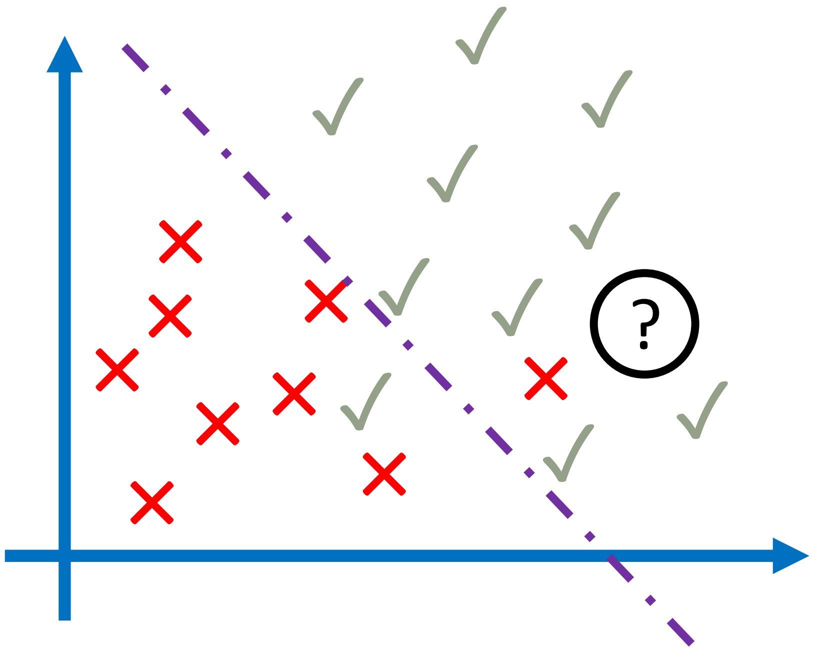
- **Decision trees** are popular classification models
  - provide **interpretability** and **accuracy**
  - constructed via **greedy heuristics** or **exact methods**
  - exact optimization methods largely focus on **binary features**

- **Our contribution:** an approach to handle **non-binary** features effectively
  - outperforms the state of the art on **non-binary** datasets with two popular objectives

Background

Classification

- A popular application of **machine learning**
- Labelling function learned from labelled data set
- The goal is to achieve high accuracy on unseen data points



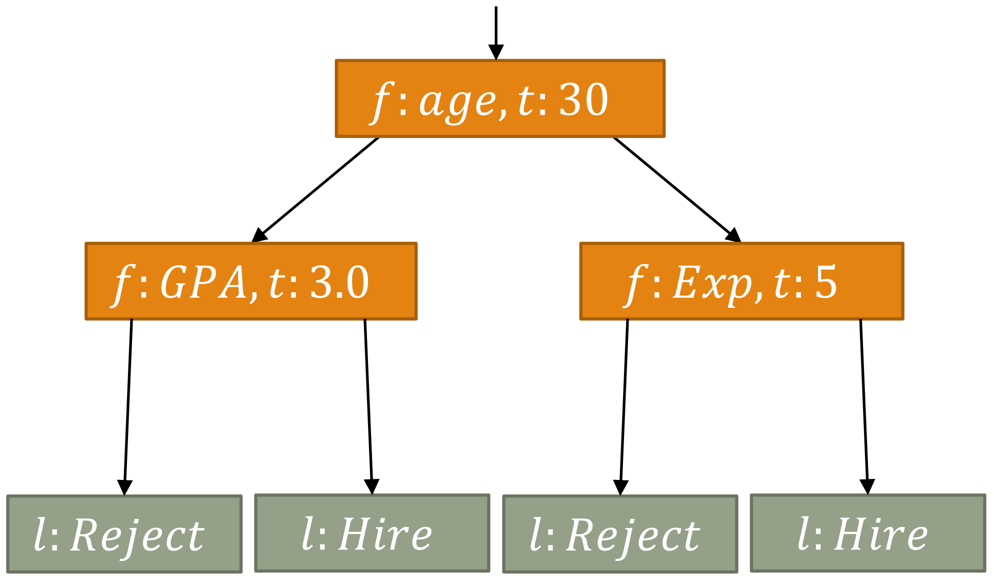
Decision Trees

- **Decision trees** are **interpretable**:
  - human-readable
  - amenable to further (logical) reasoning

- Prime candidates for **safety-critical** applications

Decision Trees

- **Branching nodes** perform a **split** on a given feature and threshold
- **Leaf nodes** assign a label



Decision Trees

- A set of features F and integer labels C
- A decision tree: $\mathcal{D} = (\mathcal{T}, \beta, \alpha, \theta)$:
  - $\mathcal{T}$ tree structure $(\mathcal{T}_\beta, \mathcal{T}_L, \delta, p, l, r)$
  - β feature selection function
  - α threshold selection function
  - θ leaf labelling function

- Recursive prediction for point x_i:

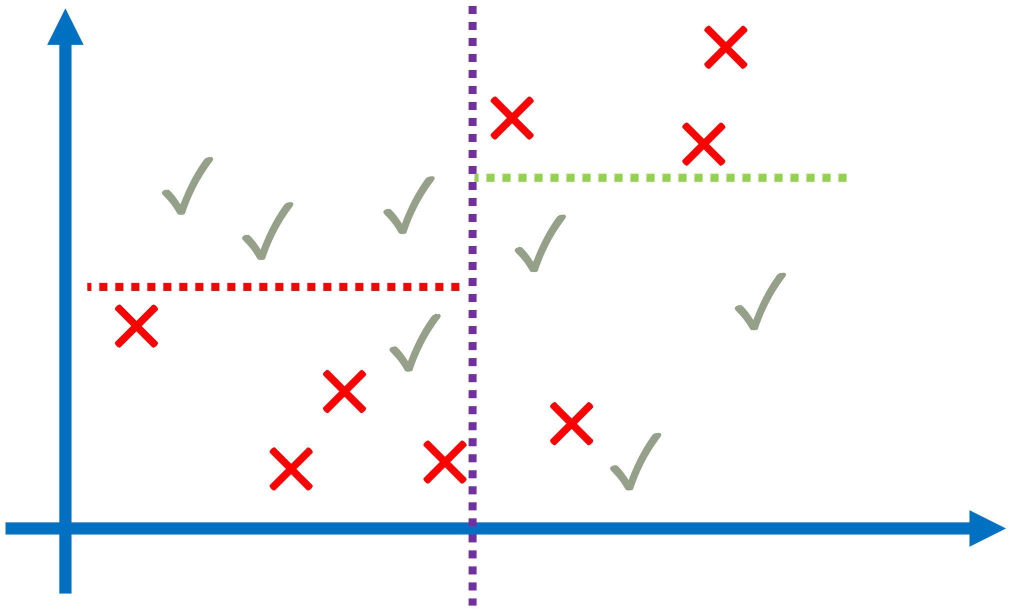
- $$\Theta(t, x_i) = \begin{cases} \theta(t) & \text{if } t \in \mathcal{T}_L \\ \Theta(l(t), x_i) & \text{else if } x_i[\beta(t)] \leq \alpha(t) \\ \Theta(r(t), x_i) & \text{else} \end{cases}$$

Decision Trees

Ways to construct decision trees:

1. **Local search** and **heuristics**

2. **Combinatorial optimization**:
   - optimality guarantees
   - additional constraints



Optimization Problem

SAT:

- A set of variables $\mathcal{X} = \{x_0, x_1, \ldots, x_n\}$ and a set of clauses $\mathcal{C} = \{C_1, C_2, \ldots, C_k\}$
- Find an assignment $\mathcal{M}: \mathcal{X} \rightarrow \{false, true\}$ that satisfies **all** clauses

MaxSAT:

- All **hard** clauses $\mathcal{C}_h$ should be satisfied
- The number of satisfied **soft** clauses $\mathcal{C}_s$ needs to be **maximized**

Encoding

Encoding Components

- It is straight-forward to encode:
  - feature selection
  - leaf labelling
  - presence at leaves
- The challenging component is the **split**
- How can we model a numerical **threshold**?

Split Encoding

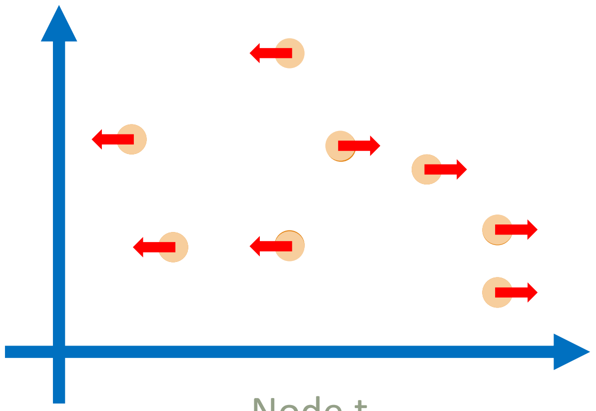
- Existing approach:
  - only support **binary** features!
  - transform **numerical** and **categorical** features into a set of **binary** ones
  - can lead to a huge number of features
  - Avellaneda's [2020], Hu et al.'s [2020], and Verhaeghe et al.'s [2020] employ this approach

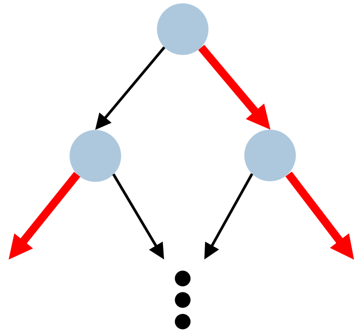
Numerical	Binary
4	001
9	011
1	000
4	001
12	111

Categorical	Binary
A	001
B	010
A	001
C	100

Split Encoding

- New idea:
  - encode the **direction** for each Point instead
  - validate the directions according to the **order** of values
  - the directions for **absent** points are encoded as well





Our Encoding

Variables:

- $a_{t,j}$: feature j is chosen at node t
- $s_{i,t}$: point i is directed left at node t
- $z_{i,t}$: point i ends up at leaf t
- $g_{t,c}$: label c is assigned to leaf t

Parameters:

- set of features F and integer labels C
- set of training examples $\mathcal{X}$
- labelling $\gamma: X \rightarrow C$
- tree structure $T = \{\delta, T_B, T_L, p, l, r\}$

Our Encoding

Variables:

- $a_{t,j}$: feature j is chosen at node t
- $s_{i,t}$: point i is directed left at node t
- $z_{i,t}$: point i ends up at leaf t
- $g_{t,c}$: label c is assigned to leaf t

Clauses:

- Exactly **one feature** is chosen at each **branching node**

$$(\neg a_{t,j}, \neg a_{t,j'}) \qquad t \in \mathcal{T}_B, j \neq j' \in F$$

$$\left(\bigvee_{j \in F} a_{t,j}\right) \qquad t \in \mathcal{T}_B$$

Our Encoding

Variables:

- $a_{t,j}$: feature j is chosen at node t
- $s_{i,t}$: point i is directed left at node t
- $z_{i,t}$: point i ends up at leaf t
- $g_{t,c}$: label c is assigned to leaf t

Clauses:

- The **directions** for splits are in **order**

$$\left(\neg a_{t,j}, s_{i,t}, \neg s_{i',t}\right) \qquad t \in \mathcal{T}_B, j \in F, (i, i') \in O_j(\mathcal{X})$$

$$\left(\neg a_{t,j}, \neg s_{i,t}, s_{i',t}\right) \quad t \in \mathcal{T}_B, j \in F, (i, i') \in O_j(\mathcal{X}), x_i[j] = x_{i'}[j]$$

Our Encoding

Variables:

- $a_{t,j}$: feature j is chosen at node t
- $s_{i,t}$: point i is directed left at node t
- $z_{i,t}$: point i ends up at leaf t
- $g_{t,c}$: label c is assigned to leaf t

Clauses:

- The splits are **non-trivial**

$$\left(\neg a_{t,j}, s_{\#_j^1,t}\right) \qquad t \in \mathcal{T}_B, j \in F$$

$$\left(\neg a_{t,j}, s_{\#_j^{|X|},t}\right) \qquad t \in \mathcal{T}_B, j \in F$$

Our Encoding

Variables:

- $a_{t,j}$: feature j is chosen at node t
- $s_{i,t}$: point i is directed left at node t
- $z_{i,t}$: point i ends up at leaf t
- $g_{t,c}$: label c is assigned to leaf t

Clauses:

- **Presence** at **leaves** matches the split directions

$$(\neg z_{i,t}, s_{i,t'}) \qquad t \in \mathcal{T}_L, x_i \in \mathcal{X}, t' \in A_l(t)$$

$$(\neg z_{i,t}, \neg s_{i,t'}) \qquad t \in \mathcal{T}_L, x_i \in \mathcal{X}, t' \in A_r(t)$$

$$\left(z_{i.t}, \bigvee_{t' \in A_l(t)} \neg s_{i,t'}, \bigvee_{t' \in A_r(t)} s_{i,t'}\right) \qquad t \in \mathcal{T}_L, x_i \in \mathcal{X}$$

Our Encoding

Variables:

- $a_{t,j}$: feature j is chosen at node t
- $s_{i,t}$: point i is directed left at node t
- $z_{i,t}$: point i ends up at leaf t
- $g_{t,c}$: label c is assigned to leaf t

Clauses:

- At most one **label** is chosen at each leaf

$$(\neg g_{t,c}, \neg g_{t,c'}) \qquad t \in \mathcal{T}_L, c \neq c' \in C$$

Learning Decision Trees

Two main **objectives:**

- **Min-depth:**
  - correctly classify **all** of the training points
  - find the **lowest depth** possible
  - solved by iterative **SAT** instances

- **Max-accuracy:**
  - **maximize** the number of correct classifications
  - use a fixed **depth**
  - solved via **MaxSAT**

Extension to Categorical Features

- Use the same idea for **categorical splits:**
  - no need to validate **order** in directions, checking **equality** is enough
  - enables **power set** branching:
    - **min-depth**: potentially more **shallow** solution
    - **max-accuracy**: potentially more **accurate** solution

Experimental

Experimental Setup

Objective	Language	Solver	Baseline	Baseline's Solver
Min-Depth	C++	MiniSAT	**Avellaneda**'s [2020] SAT-based approach	MiniSAT
Max-Accuracy	Java	Loandra	**Hu et al.**'s [2020] MaxSAT approach	Loandra
			Verhaeghe et al.'s [2020] Constraint Programming approach	Oscar

- The chosen baselines are the state-of-the-art algorithms for their respective objectives

Goals

- The benefits and applications of the two **objectives** are well-studied

- Focus on optimization performance:
  - find the solutions **faster**
  - find **near-optimal** solutions in **time-out** scenarios

Datasets

- Three types of datasets:
  - mostly **numerical** features
  - mostly **categorical** features
  - mostly **binary** features

Type	Name	$\|X\|$	$\|F_N\|$	$\|F_B\|$	$\|F_C\|$	$\tilde{f}$	$\|C\|$
N	Banknote	1372	4	0	0	5016	2
	Breast Cancer	116	9	0	0	891	2
	Cryotherapy	90	5	1	0	93	2
	Immunotherapy	91	6	1	0	166	2
	Ionosphere	351	32	2	0	8114	2
	Iris	150	4	0	0	119	3
	User Knowledge	258	5	0	0	431	4
	Vertebral Column	310	6	0	0	1741	2
	Wine	178	13	0	0	1263	3
B	Car‡	1728	6	0	0	15	2
	Monk2	169	4	2	0	11	2

Min-Depth Results

- On **non-binary** datasets, our approach is significantly faster than the baseline
- As expected, the existing approach works better on **binary** datasets

Dataset	Min Depth	Time (s)	
		Ours	SAT [3]
Banknote	4	**5.82**	T/O [4]
Breast Cancer	4	**6.59**	T/O [4]
Cryotherapy	4	**0.08**	0.24
Immunotherapy	4	**0.18**	1.3
Ionosphere	?	T/O [4]	T/O [3]
Iris	4	**0.04**	0.17
User Knowledge	5	**1.31**	59.44
Vertebral Column	5	**87.35**	T/O [5]
Wine	3	**0.11**	14.75
Car	8	T/O [8]	**89.1**
Monk2	6	2.73	**0.28**

Max-Accuracy Results

- On **non-binary** datasets, our approach is significantly faster than the baselines
- As expected, existing approaches work better on **binary** datasets
- Our approach still finds **optimal solutions** for **binary** datasets most of the time

Dataset	Depth	Solution Cost			Time (s)		
		Ours	MaxSAT [15]	CP [23]	Ours	MaxSAT [15]	CP [23]
Banknote	2	**100**	176	**100**	**16.83**	T/O	512.21
	3	**23**	550	100	**105.79**	T/O	T/O
	4	**0**	88	100	**18.98**	T/O	T/O
Breast Cancer	2	**19**	24	**19**	**5.07**	T/O	22.19
	3	**9**	25	12	**242.16**	T/O	T/O
	4	**0**	18	11	**20.79**	T/O	T/O
Cryotherapy	2	**5**	**5**	**5**	**0.57**	3.68	4.21
	3	**1**	**1**	**1**	**0.73**	17.57	27.39
	4	**0**	**0**	**0**	**0.75**	24.14	7.61
Immunotherapy	2	**8**	**8**	**8**	**0.99**	10.53	5.22
	3	**4**	**4**	**4**	**3.81**	T/O	146.45
	4	**0**	1	**0**	**1.27**	T/O	18.53
Ionosphere	2	**29**	41	**29**	**155.06**	T/O	T/O
	3	**21**	186	29	T/O	T/O	T/O
	4	**10**	76	28	T/O	T/O	T/O
Iris	2	**6**	-	-	**0.6**	-	-
	3	**1**	-	-	**0.77**	-	-
	4	**0**	-	-	**0.82**	-	-
User Knowledge	2	**35**	-	-	**1.94**	-	-
	3	**10**	-	-	**3.29**	-	-
	4	**1**	-	-	**3.86**	-	-
Vertebral Column	2	**45**	46	**45**	**15.79**	T/O	67.91
	3	**32**	44	42	T/O	T/O	T/O
	4	**15**	39	42	T/O	T/O	T/O
Wine	2	**6**	-	-	**1.25**	-	-
	3	**0**	-	-	**1.62**	-	-
Car	2	**250**	**250**	**250**	12.67	9.2	**2.16**
	3	**182**	**182**	**182**	T/O	T/O	**5.99**
	4	**122**	**122**	**122**	T/O	T/O	**14.09**
Monk2	2	**57**	**57**	**57**	2.74	4.38	**1.38**
	3	**42**	**42**	**42**	T/O	826.31	**3.6**
	4	32	**31**	**31**	T/O	T/O	**8.12**

Summary

- Novel MaxSAT-based encoding for constructing **optimal decision trees** for datasets with **numerical** and **categorical** Features
- Can be employed by both **min-depth** and **max-accuracy** objectives
- Supports **power set** splitting on **categorical** features to achieve **compactness**
- Significantly outperforms the state of the art for **non-binary** datasets

Thank You!

Questions & Answers

POUYA SHATI, ELDAN COHEN, SHEILA MCILRAITH

UNIVERSITY OF TORONTO

CP 2021

SEPTEMBER 13, 2021