

SAT-Based Learning of Compact Binary Decision Diagrams for Classification

POUYA SHATI, ELDAN COHEN, SHEILA MCILRAITH

UNIVERSITY OF TORONTO

VECTOR INSTITUTE FOR ARTIFICIAL INTELLIGENCE

CP-23

AUGUST 2023



UNIVERSITY OF
TORONTO



VECTOR
INSTITUTE

Motivation

- **Decision trees as interpretable classifiers**
 - However, number of splits **exponential** in depth
- **Binary Decision Diagrams** as more **compact** alternatives
 - Same split across each level

Our Contribution

- Propose **SAT-based encoding** for learning max-accuracy BDDs
- Model the **size** of the BDD as a **secondary objective**
- Introduce and model **Multi-Dimensional BDDs** as more expressive alternatives
- Demonstrate **improved compactness** with **maintained accuracy** in experiments

Background

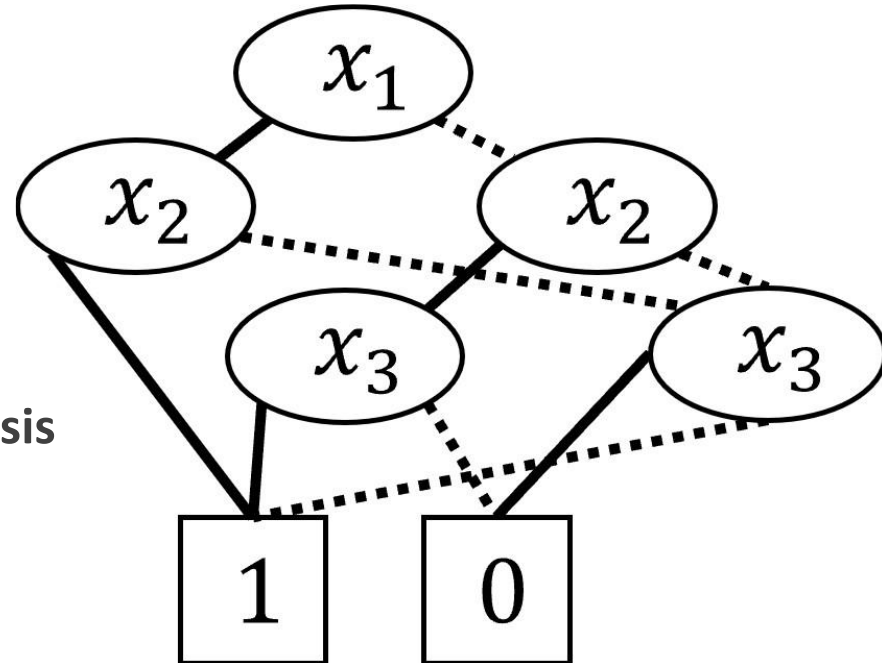
Encodings

Experiments

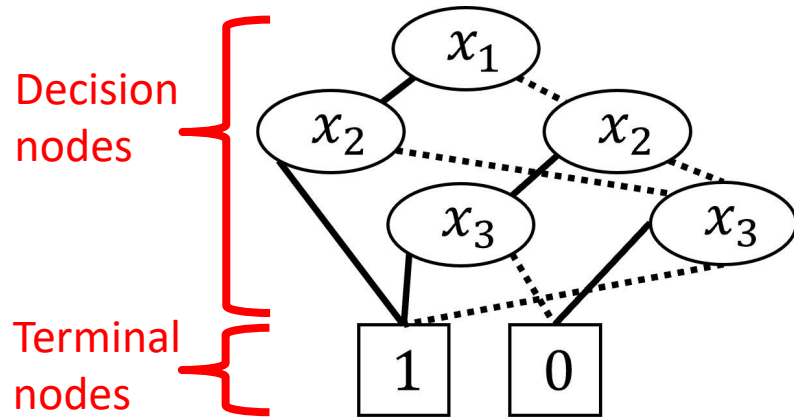
- Binary Decision Diagrams
- MaxSAT

Binary Decision Diagrams (BDD)

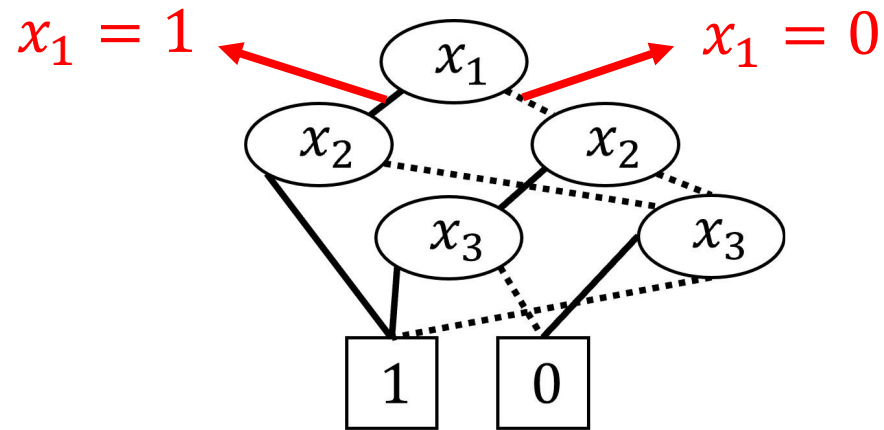
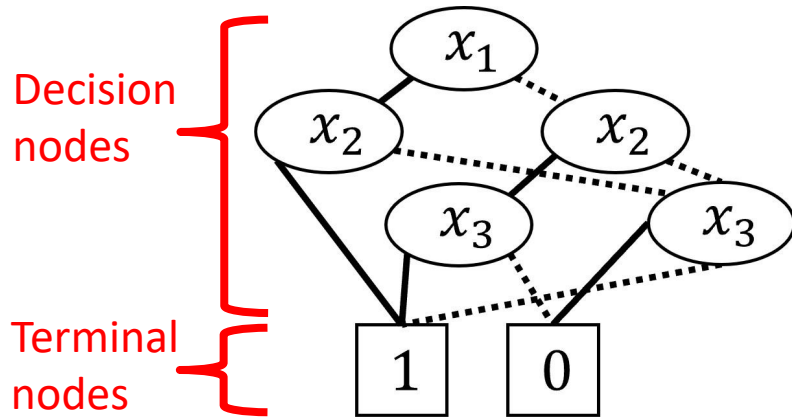
- **Rooted, directed, acyclic** graph
- Representation of a **Boolean** function
- Historically utilized towards **hardware synthesis**
- Recent focus on **BDD classifiers**



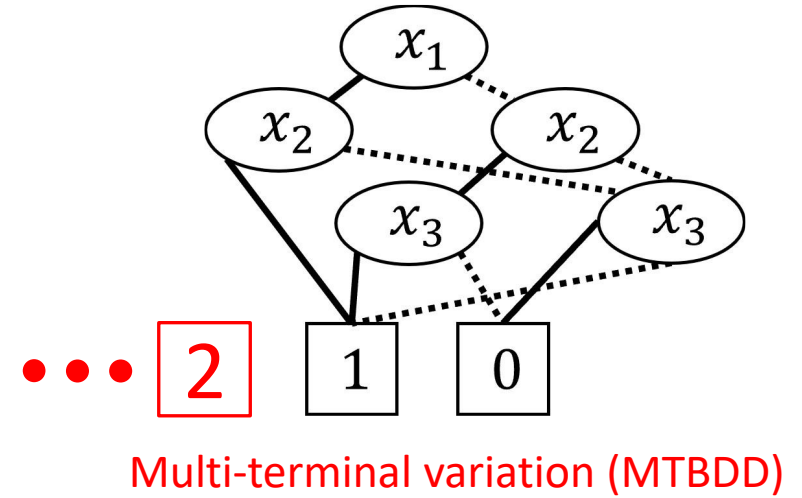
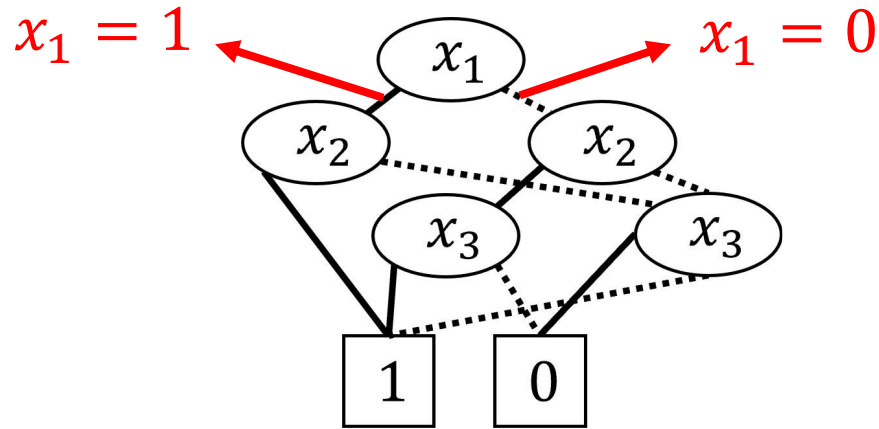
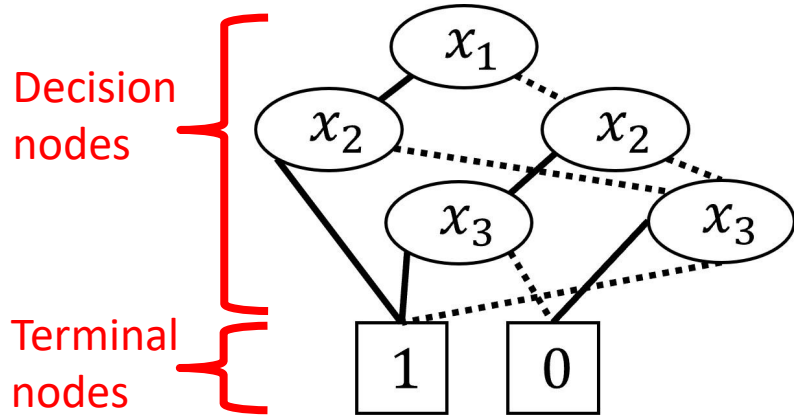
Binary Decision Diagrams (BDD)



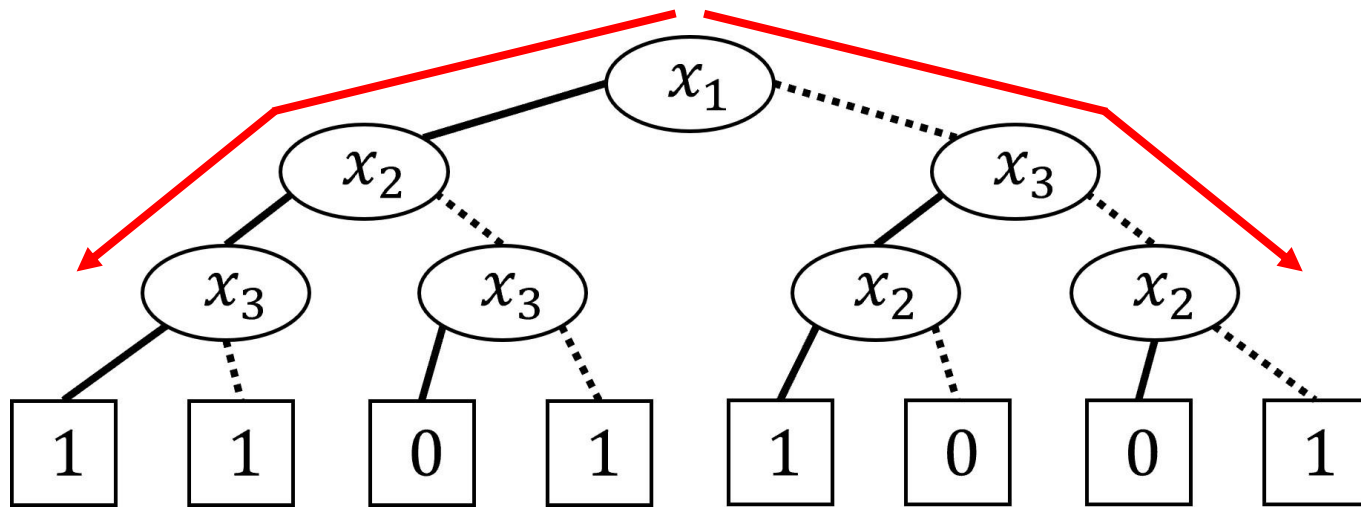
Binary Decision Diagrams (BDD)



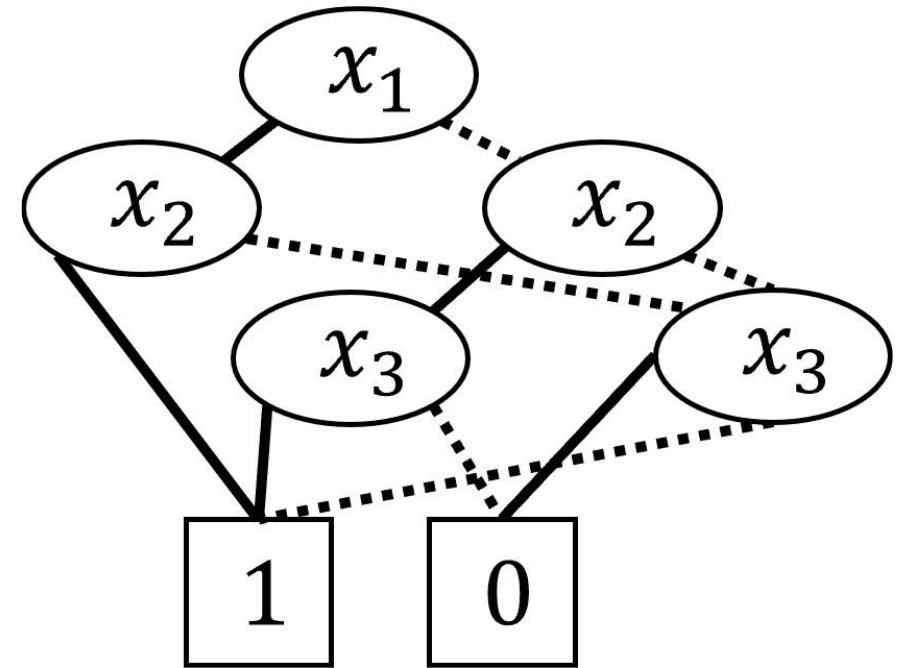
Binary Decision Diagrams (BDD)



Binary Decision Diagrams (BDD)



Unreduced and unordered



Reduced and ordered

MaxSAT

- Set of binary variables $\mathcal{X} = \{x_0, x_1, \dots, x_n\}$
- Set of clauses
 - Each clause C_i is a subset of literals $\mathcal{X} \cup \neg\mathcal{X}$
- Find an assignment $\mathcal{M}: \mathcal{X} \rightarrow \{false, true\}$
- Satisfy all **hard** clauses \mathcal{C}_h
- Maximize the number of satisfied **soft** clauses \mathcal{C}_s

Background

Encodings

Experiments

- BDD Encoding
- Size Optimization
- Multi-dimensional BDDs
- Expressiveness relations

BDD Encoding

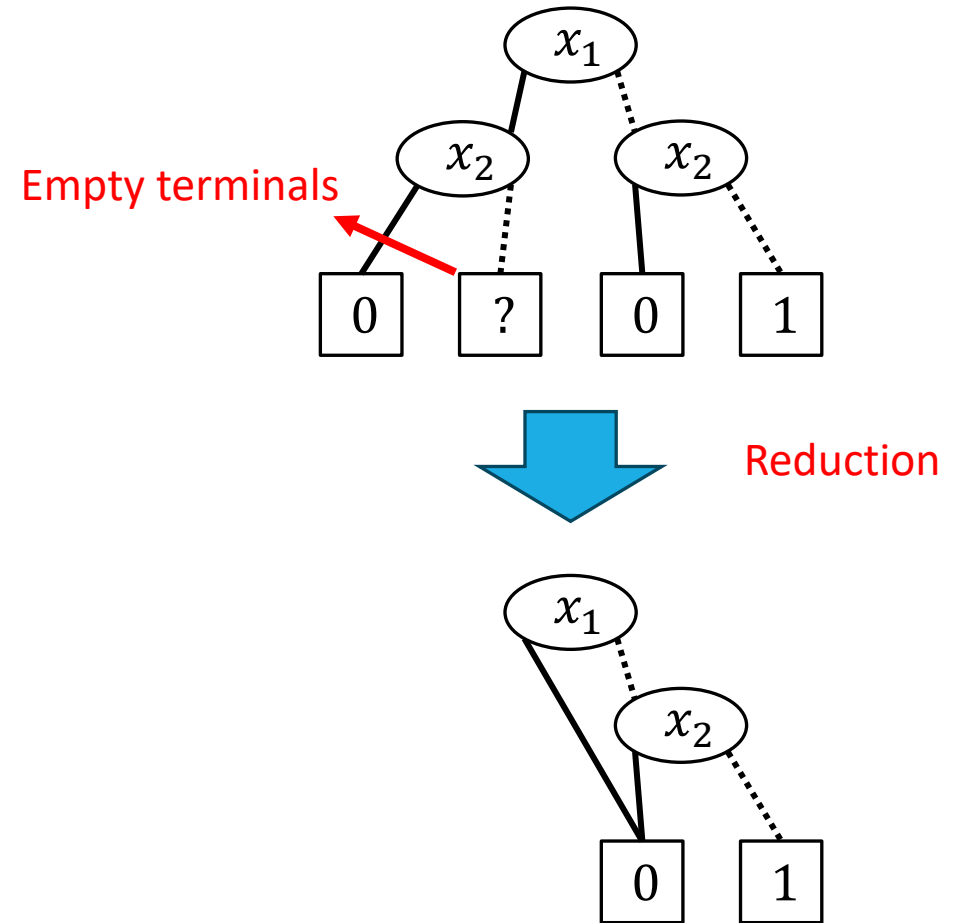
- **Direct encoding** of numerical features
 - Based on [Shati, Cohen, McIlraith, CP2021]

- Employ **splits** at each node
 - (feature, threshold)
- The alternative: **binarize features in advance**
 - **excessive** number of features

BDD Encoding

- **Direct encoding** of numerical features
 - Based on [Shati, Cohen, McClraith, CP2021]

- Learn ordered but **unreduced** BDDs
 - Reduction in a **second stage**



BDD Encoding

- **Hard clauses:**

$$(a_{s,j}, \neg a_{s,j+1}) \quad s < s_{max}, j \in F$$

$$(a_{s,0}) \quad s < s_{max}$$

$$(\neg a_{s,j}, a_{s,j+1}, d_{s,i_1}, \neg d_{s,i_2}) \quad s < s_{max}, j \in F, (i_1, i_2) \in O_j(X)$$

$$(\neg a_{s,j}, a_{s,j+1}, \neg d_{s,i_1}, d_{s,i_2}) \quad s < s_{max}, j \in F, (i_1, i_2) \in O_j^-(X)$$

$$(\neg a_{s,j}, a_{s,j+1}, d_{s,\#_j^1}) \quad s < s_{max}, j \in F$$

$$(\neg c_{t,l_1}, \neg c_{t,l_2}) \quad t \in \mathcal{N}_T, l_1, l_2 \in K$$

$$\left(\bigvee_{l \in L} c_{t,l} \right) \quad t \in \mathcal{N}_T$$

$$\left(\bigvee_{s \in A_L(t)} \neg d_{s,i}, \bigvee_{s \in A_R(t)} d_{s,i}, c_{t,y_i}, \neg o_i \right) \quad t \in \mathcal{N}_T, x_i \in X$$

- **Soft clauses:**

$$(o_i) \quad x_i \in X$$

- **Variables:**

- $a_{s,j}$: The feature chosen at split s is or comes before j .

- $d_{s,i}$: Point x_i is directed to the left child at split s .

- $c_{t,l}$: Output label l is assigned to terminal node t .

- o_i : Point x_i is classified correctly.

BDD Encoding

- **Hard clauses:**

$$(a_{s,j}, \neg a_{s,j+1}) \quad s < s_{max}, j \in F$$

$$(a_{s,0}) \quad s < s_{max}$$

$$(\neg a_{s,j}, a_{s,j+1}, d_{s,i_1}, \neg d_{s,i_2}) \quad s < s_{max}, j \in F, (i_1, i_2) \in O_j(X)$$

$$(\neg a_{s,j}, a_{s,j+1}, \neg d_{s,i_1}, d_{s,i_2}) \quad s < s_{max}, j \in F, (i_1, i_2) \in O_j^-(X)$$

$$(\neg a_{s,j}, a_{s,j+1}, d_{s,\#_j^1}) \quad s < s_{max}, j \in F$$

$$(\neg c_{t,l_1}, \neg c_{t,l_2}) \quad t \in \mathcal{N}_T, l_1, l_2 \in K$$

$$\left(\bigvee_{l \in L} c_{t,l} \right) \quad t \in \mathcal{N}_T$$

$$\left(\bigvee_{s \in A_L(t)} \neg d_{s,i}, \bigvee_{s \in A_R(t)} d_{s,i}, c_{t,y_i}, \neg o_i \right) \quad t \in \mathcal{N}_T, x_i \in X$$

- **Soft clauses:**

$$(o_i) \quad x_i \in X$$

- **Variables:**

- $a_{s,j}$: The feature chosen at split s is or comes before j .

- $d_{s,i}$: Point x_i is directed to the left child at split s .

- $c_{t,l}$: Output label l is assigned to terminal node t .

- o_i : Point x_i is classified correctly.

BDD Encoding

- **Hard clauses:**

$$(a_{s,j}, \neg a_{s,j+1}) \quad s < s_{max}, j \in F$$

$$(a_{s,0}) \quad s < s_{max}$$

$$(\neg a_{s,j}, a_{s,j+1}, d_{s,i_1}, \neg d_{s,i_2}) \quad s < s_{max}, j \in F, (i_1, i_2) \in O_j(X)$$

$$(\neg a_{s,j}, a_{s,j+1}, \neg d_{s,i_1}, d_{s,i_2}) \quad s < s_{max}, j \in F, (i_1, i_2) \in O_j^-(X)$$

$$(\neg a_{s,j}, a_{s,j+1}, d_{s,\#_j^1}) \quad s < s_{max}, j \in F$$

$$(\neg c_{t,l_1}, \neg c_{t,l_2}) \quad t \in \mathcal{N}_T, l_1, l_2 \in K$$

$$\left(\bigvee_{l \in L} c_{t,l} \right) \quad t \in \mathcal{N}_T$$

$$\left(\bigvee_{s \in A_L(t)} \neg d_{s,i}, \bigvee_{s \in A_R(t)} d_{s,i}, c_{t,y_i}, \neg o_i \right) \quad t \in \mathcal{N}_T, x_i \in X$$

- **Soft clauses:**

$$(o_i) \quad x_i \in X$$

- **Variables:**

- $a_{s,j}$: The feature chosen at split s is or comes before j .

- $d_{s,i}$: Point x_i is directed to the left child at split s .

- $c_{t,l}$: Output label l is assigned to terminal node t .

- o_i : Point x_i is classified correctly.

BDD Encoding

- **Hard clauses:**

$$(a_{s,j}, \neg a_{s,j+1}) \quad s < s_{max}, j \in F$$

$$(a_{s,0}) \quad s < s_{max}$$

$$(\neg a_{s,j}, a_{s,j+1}, d_{s,i_1}, \neg d_{s,i_2}) \quad s < s_{max}, j \in F, (i_1, i_2) \in O_j(X)$$

$$(\neg a_{s,j}, a_{s,j+1}, \neg d_{s,i_1}, d_{s,i_2}) \quad s < s_{max}, j \in F, (i_1, i_2) \in O_j^-(X)$$

$$(\neg a_{s,j}, a_{s,j+1}, d_{s,\#_j^1}) \quad s < s_{max}, j \in F$$

$$(\neg c_{t,l_1}, \neg c_{t,l_2}) \quad t \in \mathcal{N}_T, l_1, l_2 \in K$$

$$\left(\bigvee_{l \in L} c_{t,l} \right) \quad t \in \mathcal{N}_T$$

$$\left(\bigvee_{s \in A_L(t)} \neg d_{s,i}, \bigvee_{s \in A_R(t)} d_{s,i}, c_{t,y_i}, \neg o_i \right) \quad t \in \mathcal{N}_T, x_i \in X$$

- **Soft clauses:**

$$(o_i) \quad x_i \in X$$

- **Variables:**

- $a_{s,j}$: The feature chosen at split s is or comes before j .

- $d_{s,i}$: Point x_i is directed to the left child at split s .

- $c_{t,l}$: Output label l is assigned to terminal node t .

- o_i : Point x_i is classified correctly.

BDD Encoding

- **Hard clauses:**

$$(a_{s,j}, \neg a_{s,j+1}) \quad s < s_{max}, j \in F$$

$$(a_{s,0}) \quad s < s_{max}$$

$$(\neg a_{s,j}, a_{s,j+1}, d_{s,i_1}, \neg d_{s,i_2}) \quad s < s_{max}, j \in F, (i_1, i_2) \in O_j(X)$$

$$(\neg a_{s,j}, a_{s,j+1}, \neg d_{s,i_1}, d_{s,i_2}) \quad s < s_{max}, j \in F, (i_1, i_2) \in O_j^-(X)$$

$$(\neg a_{s,j}, a_{s,j+1}, d_{s,\#_j^1}) \quad s < s_{max}, j \in F$$

$$(\neg c_{t,l_1}, \neg c_{t,l_2}) \quad t \in \mathcal{N}_T, l_1, l_2 \in K$$

$$\left(\bigvee_{l \in L} c_{t,l} \right) \quad t \in \mathcal{N}_T$$

$$\left(\bigvee_{s \in A_L(t)} \neg d_{s,i}, \bigvee_{s \in A_R(t)} d_{s,i}, c_{t,y_i}, \neg o_i \right) \quad t \in \mathcal{N}_T, x_i \in X$$

- **Soft clauses:**

$$(o_i) \quad x_i \in X$$

- **Variables:**

- $a_{s,j}$: The feature chosen at split s is or comes before j .

- $d_{s,i}$: Point x_i is directed to the left child at split s .

- $c_{t,l}$: Output label l is assigned to terminal node t .

- O_i : Point x_i is classified correctly.

BDD Encoding

- **Hard clauses:**

$$(a_{s,j}, \neg a_{s,j+1}) \quad s < s_{max}, j \in F$$

$$(a_{s,0}) \quad s < s_{max}$$

$$(\neg a_{s,j}, a_{s,j+1}, d_{s,i_1}, \neg d_{s,i_2}) \quad s < s_{max}, j \in F, (i_1, i_2) \in O_j(X)$$

$$(\neg a_{s,j}, a_{s,j+1}, \neg d_{s,i_1}, d_{s,i_2}) \quad s < s_{max}, j \in F, (i_1, i_2) \in O_j^-(X)$$

$$(\neg a_{s,j}, a_{s,j+1}, d_{s,\#_j^1}) \quad s < s_{max}, j \in F$$

$$(\neg c_{t,l_1}, \neg c_{t,l_2}) \quad t \in \mathcal{N}_T, l_1, l_2 \in K$$

$$\left(\bigvee_{l \in L} c_{t,l} \right) \quad t \in \mathcal{N}_T$$

$$\left(\bigvee_{s \in A_L(t)} \neg d_{s,i}, \bigvee_{s \in A_R(t)} d_{s,i}, c_{t,y_i}, \neg o_i \right) \quad t \in \mathcal{N}_T, x_i \in X$$

- **Soft clauses:**

$$(o_i) \quad x_i \in X$$

- **Variables:**

- $a_{s,j}$: The feature chosen at split s is or comes before j .
- $d_{s,i}$: Point x_i is directed to the left child at split s .
- $c_{t,l}$: Output label l is assigned to terminal node t .
- o_i : Point x_i is classified correctly.

◦ Exactly one feature is selected at each split.

BDD Encoding

- **Hard clauses:**

$$(a_{s,j}, \neg a_{s,j+1}) \quad s < s_{max}, j \in F$$

$$(a_{s,0}) \quad s < s_{max}$$

$$(\neg a_{s,j}, a_{s,j+1}, d_{s,i_1}, \neg d_{s,i_2}) \quad s < s_{max}, j \in F, (i_1, i_2) \in O_j(X)$$

$$(\neg a_{s,j}, a_{s,j+1}, \neg d_{s,i_1}, d_{s,i_2}) \quad s < s_{max}, j \in F, (i_1, i_2) \in O_j^-(X)$$

$$(\neg a_{s,j}, a_{s,j+1}, d_{s,\#_j^1}) \quad s < s_{max}, j \in F$$

$$(\neg c_{t,l_1}, \neg c_{t,l_2}) \quad t \in \mathcal{N}_T, l_1, l_2 \in K$$

$$\left(\bigvee_{l \in L} c_{t,l} \right) \quad t \in \mathcal{N}_T$$

$$\left(\bigvee_{s \in A_L(t)} \neg d_{s,i}, \bigvee_{s \in A_R(t)} d_{s,i}, c_{t,y_i}, \neg o_i \right) \quad t \in \mathcal{N}_T, x_i \in X$$

- **Soft clauses:**

$$(o_i) \quad x_i \in X$$

- **Variables:**

- $a_{s,j}$: The feature chosen at split s is or comes before j .

- $d_{s,i}$: Point x_i is directed to the left child at split s .

- $c_{t,l}$: Output label l is assigned to terminal node t .

- o_i : Point x_i is classified correctly.

- Exactly one feature is selected at each split.

- Selected feature enforces order of values.

BDD Encoding

- **Hard clauses:**

$$(a_{s,j}, \neg a_{s,j+1}) \quad s < s_{max}, j \in F$$

$$(a_{s,0}) \quad s < s_{max}$$

$$(\neg a_{s,j}, a_{s,j+1}, d_{s,i_1}, \neg d_{s,i_2}) \quad s < s_{max}, j \in F, (i_1, i_2) \in O_j(X)$$

$$(\neg a_{s,j}, a_{s,j+1}, \neg d_{s,i_1}, d_{s,i_2}) \quad s < s_{max}, j \in F, (i_1, i_2) \in O_j^-(X)$$

$$(\neg a_{s,j}, a_{s,j+1}, d_{s,\#_j^1}) \quad s < s_{max}, j \in F$$

$$(\neg c_{t,l_1}, \neg c_{t,l_2}) \quad t \in \mathcal{N}_T, l_1, l_2 \in K$$

$$\left(\bigvee_{l \in L} c_{t,l} \right) \quad t \in \mathcal{N}_T$$

$$\left(\bigvee_{s \in A_L(t)} \neg d_{s,i}, \bigvee_{s \in A_R(t)} d_{s,i}, c_{t,y_i}, \neg o_i \right) \quad t \in \mathcal{N}_T, x_i \in X$$

- **Soft clauses:**

$$(o_i) \quad x_i \in X$$

- **Variables:**

- $a_{s,j}$: The feature chosen at split s is or comes before j .

- $d_{s,i}$: Point x_i is directed to the left child at split s .

- $c_{t,l}$: Output label l is assigned to terminal node t .

- O_i : Point x_i is classified correctly.

- Exactly one feature is selected at each split.

- Selected feature enforces order of values.

- Exactly one label is selected at each leaf.

BDD Encoding

- **Hard clauses:**

$$(a_{s,j}, \neg a_{s,j+1}) \quad s < s_{max}, j \in F$$

$$(a_{s,0}) \quad s < s_{max}$$

$$(\neg a_{s,j}, a_{s,j+1}, d_{s,i_1}, \neg d_{s,i_2}) \quad s < s_{max}, j \in F, (i_1, i_2) \in O_j(X)$$

$$(\neg a_{s,j}, a_{s,j+1}, \neg d_{s,i_1}, d_{s,i_2}) \quad s < s_{max}, j \in F, (i_1, i_2) \in O_j^-(X)$$

$$(\neg a_{s,j}, a_{s,j+1}, d_{s,\#_j^1}) \quad s < s_{max}, j \in F$$

$$(\neg c_{t,l_1}, \neg c_{t,l_2}) \quad t \in \mathcal{N}_T, l_1, l_2 \in K$$

$$\left(\bigvee_{l \in L} c_{t,l} \right) \quad t \in \mathcal{N}_T$$

$$\left(\bigvee_{s \in A_L(t)} \neg d_{s,i}, \bigvee_{s \in A_R(t)} d_{s,i}, c_{t,y_i}, \neg o_i \right) \quad t \in \mathcal{N}_T, x_i \in X$$

- **Soft clauses:**

$$(o_i) \quad x_i \in X$$

- **Variables:**

- $a_{s,j}$: The feature chosen at split s is or comes before j .

- $d_{s,i}$: Point x_i is directed to the left child at split s .

- $c_{t,l}$: Output label l is assigned to terminal node t .

- O_i : Point x_i is classified correctly.

- Exactly one feature is selected at each split.

- Selected feature enforces order of values.

- Exactly one label is selected at each leaf.

- The leaf and point labels match for a correctly classified point.

BDD Encoding

- **Hard clauses:**

$$(a_{s,j}, \neg a_{s,j+1}) \quad s < s_{max}, j \in F$$

$$(a_{s,0}) \quad s < s_{max}$$

$$(\neg a_{s,j}, a_{s,j+1}, d_{s,i_1}, \neg d_{s,i_2}) \quad s < s_{max}, j \in F, (i_1, i_2) \in O_j(X)$$

$$(\neg a_{s,j}, a_{s,j+1}, \neg d_{s,i_1}, d_{s,i_2}) \quad s < s_{max}, j \in F, (i_1, i_2) \in O_j^-(X)$$

$$(\neg a_{s,j}, a_{s,j+1}, d_{s,\#_j^1}) \quad s < s_{max}, j \in F$$

$$(\neg c_{t,l_1}, \neg c_{t,l_2}) \quad t \in \mathcal{N}_T, l_1, l_2 \in K$$

$$\left(\bigvee_{l \in L} c_{t,l} \right) \quad t \in \mathcal{N}_T$$

$$\left(\bigvee_{s \in A_L(t)} \neg d_{s,i}, \bigvee_{s \in A_R(t)} d_{s,i}, c_{t,y_i}, \neg o_i \right) \quad t \in \mathcal{N}_T, x_i \in X$$

- **Soft clauses:**

$$(o_i) \quad x_i \in X$$

- **Variables:**

- $a_{s,j}$: The feature chosen at split s is or comes before j .
- $d_{s,i}$: Point x_i is directed to the left child at split s .
- $c_{t,l}$: Output label l is assigned to terminal node t .
- o_i : Point x_i is classified correctly.

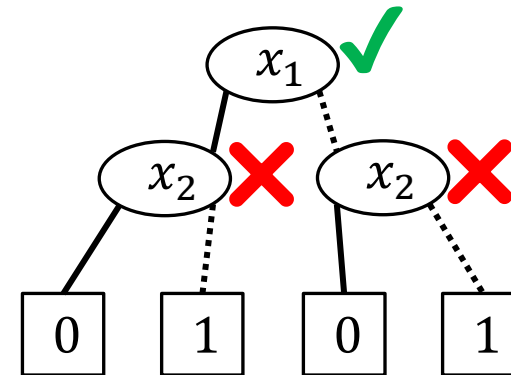
- Exactly one feature is selected at each split.
- Selected feature enforces order of values.
- Exactly one label is selected at each leaf.
- The leaf and point labels match for a correctly classified point.
- Number of correctly classified points are maximized.

Size Optimization

- Model the size of BDD's **reduced version**

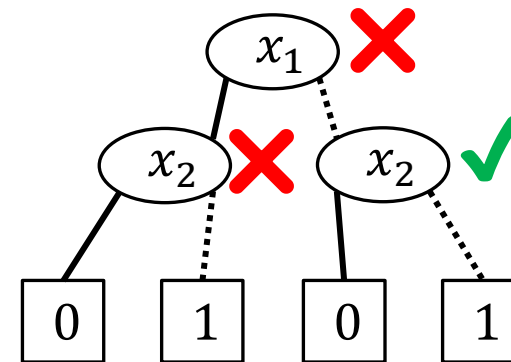
Size Optimization

- Model the size of BDD's **reduced version**
- For each decision node:
 - Can it be **replaced** by one of its **children**?



Size Optimization

- Model the size of BDD's **reduced version**
- For each decision node:
 - Can it be **replaced** by one of its **children**?
 - Can it be **merged** with one of the **previous** decision nodes in the **same level**?



Size Optimization

- Model the size of BDD's **reduced version**
- For each decision node:
 - Can it be **replaced** by one of its **children**?
 - Can it be **merged** with one of the **previous** decision nodes in the **same level**?
- Two approaches:
 - **1-stage**: add as a **secondary objective** to accuracy
 - **2-stage**: use as a **post-processing step** to choose **empty** node labels

Size Optimization

- **Hard clauses:**

$$\begin{aligned}
 (\neg c_{t_1,l}, \neg c_{t_2,l}, \neg \sigma_{t_1,t_2}) & \quad (t_1, t_2, \Delta) \in G(s_{max}), l \in K \\
 (\neg c_{t_1,l}, c_{t_2,l}, \sigma_{t_1,t_2}) & \quad (t_1, t_2, \Delta) \in G(s_{max}), l \in K \\
 (c_{t_1,l}, \neg c_{t_2,l}, \sigma_{t_1,t_2}) & \quad (t_1, t_2, \Delta) \in G(s_{max}), l \in K \\
 (b_{\Delta \lfloor t_1/\Delta \rfloor, \Delta}, \neg \sigma_{t_1,t_2}) & \quad (t_1, t_2, \Delta) \in G(s_{max}) \\
 (\neg r_{t_1\Delta, t_2\Delta, \Delta}, \neg c_{t_1\Delta+\delta, l}, c_{t_2\Delta+\delta, l}) & \quad \Delta \in P(s_{max}), t_1 < t_2 < 2^{s_{max}}/\Delta, \delta < \Delta, l \in K \\
 (\neg r_{t_1\Delta, t_2\Delta, \Delta}, c_{t_1\Delta+\delta, l}, \neg c_{t_2\Delta+\delta, l}) & \quad \Delta \in P(s_{max}), t_1 < t_2 < 2^{s_{max}}/\Delta, \delta < \Delta, l \in K
 \end{aligned}$$

- **Soft clauses:**

$$\left(\bigvee_{0 \leq t_2 < t} r_{t_2\Delta, t\Delta, \Delta}, \neg b_{t\Delta, \Delta} \right) \quad \Delta \in P(s_{max}), t < 2^{s_{max}}/\Delta$$

- **Definitions:**

$$\begin{aligned}
 G(1) &= \{(0, 1, 2)\} \\
 G(p) &= G(p-1) \cup \{(t_1 + 2^{p-1}, t_2 + 2^{p-1}, \Delta) \mid (t_1, t_2, \Delta) \in G(p-1)\} \\
 &\quad \cup \{(t, t + 2^{p-1}, 2^p) \mid 0 \leq t < 2^{p-1}\}
 \end{aligned}$$

- **Variables:**

- α_{t_1, t_2} : Terminals t_1 and t_2 have been assigned different output labels.
- $b_{t, \Delta}$: The sequence of Δ labels starting from terminal node t (inclusive) cannot be divided into two equal sub-sequences.
- $r_{t_1, t_2, \Delta}$: The sequence of Δ labels starting from terminal node t_1 is equal to the sequence of Δ labels starting from terminal node t_2 (both inclusive).

Size Optimization

- **Hard clauses:**

$$\begin{aligned}
 (\neg c_{t_1,l}, \neg c_{t_2,l}, \neg \sigma_{t_1,t_2}) & \quad (t_1, t_2, \Delta) \in G(s_{max}), l \in K \\
 (\neg c_{t_1,l}, c_{t_2,l}, \sigma_{t_1,t_2}) & \quad (t_1, t_2, \Delta) \in G(s_{max}), l \in K \\
 (c_{t_1,l}, \neg c_{t_2,l}, \sigma_{t_1,t_2}) & \quad (t_1, t_2, \Delta) \in G(s_{max}), l \in K \\
 (b_{\Delta \lfloor t_1/\Delta \rfloor, \Delta}, \neg \sigma_{t_1,t_2}) & \quad (t_1, t_2, \Delta) \in G(s_{max}) \\
 (\neg r_{t_1\Delta, t_2\Delta, \Delta}, \neg c_{t_1\Delta+\delta, l}, c_{t_2\Delta+\delta, l}) & \quad \Delta \in P(s_{max}), t_1 < t_2 < 2^{s_{max}}/\Delta, \delta < \Delta, l \in K \\
 (\neg r_{t_1\Delta, t_2\Delta, \Delta}, c_{t_1\Delta+\delta, l}, \neg c_{t_2\Delta+\delta, l}) & \quad \Delta \in P(s_{max}), t_1 < t_2 < 2^{s_{max}}/\Delta, \delta < \Delta, l \in K
 \end{aligned}$$

- **Soft clauses:**

$$\left(\bigvee_{0 \leq t_2 < t} r_{t_2\Delta, t\Delta, \Delta}, \neg b_{t\Delta, \Delta} \right) \quad \Delta \in P(s_{max}), t < 2^{s_{max}}/\Delta$$

- **Definitions:**

$$\begin{aligned}
 G(1) &= \{(0, 1, 2)\} \\
 G(p) &= G(p-1) \cup \{(t_1 + 2^{p-1}, t_2 + 2^{p-1}, \Delta) \mid (t_1, t_2, \Delta) \in G(p-1)\} \\
 &\quad \cup \{(t, t + 2^{p-1}, 2^p) \mid 0 \leq t < 2^{p-1}\}
 \end{aligned}$$

- **Variables:**

- α_{t_1, t_2} : Terminals t_1 and t_2 have been assigned different output labels.
- $b_{t, \Delta}$: The sequence of Δ labels starting from terminal node t (inclusive) cannot be divided into two equal sub-sequences.
- $r_{t_1, t_2, \Delta}$: The sequence of Δ labels starting from terminal node t_1 is equal to the sequence of Δ labels starting from terminal node t_2 (both inclusive).

Size Optimization

- **Hard clauses:**

$$\begin{aligned}
 (\neg c_{t_1,l}, \neg c_{t_2,l}, \neg \sigma_{t_1,t_2}) & \quad (t_1, t_2, \Delta) \in G(s_{max}), l \in K \\
 (\neg c_{t_1,l}, c_{t_2,l}, \sigma_{t_1,t_2}) & \quad (t_1, t_2, \Delta) \in G(s_{max}), l \in K \\
 (c_{t_1,l}, \neg c_{t_2,l}, \sigma_{t_1,t_2}) & \quad (t_1, t_2, \Delta) \in G(s_{max}), l \in K \\
 (b_{\Delta \lfloor t_1/\Delta \rfloor, \Delta}, \neg \sigma_{t_1,t_2}) & \quad (t_1, t_2, \Delta) \in G(s_{max}) \\
 (\neg r_{t_1\Delta, t_2\Delta, \Delta}, \neg c_{t_1\Delta+\delta, l}, c_{t_2\Delta+\delta, l}) & \quad \Delta \in P(s_{max}), t_1 < t_2 < 2^{s_{max}}/\Delta, \delta < \Delta, l \in K \\
 (\neg r_{t_1\Delta, t_2\Delta, \Delta}, c_{t_1\Delta+\delta, l}, \neg c_{t_2\Delta+\delta, l}) & \quad \Delta \in P(s_{max}), t_1 < t_2 < 2^{s_{max}}/\Delta, \delta < \Delta, l \in K
 \end{aligned}$$

- **Soft clauses:**

$$\left(\bigvee_{0 \leq t_2 < t} r_{t_2\Delta, t\Delta, \Delta}, \neg b_{t\Delta, \Delta} \right) \quad \Delta \in P(s_{max}), t < 2^{s_{max}}/\Delta$$

- **Definitions:**

$$\begin{aligned}
 G(1) &= \{(0, 1, 2)\} \\
 G(p) &= G(p-1) \cup \{(t_1 + 2^{p-1}, t_2 + 2^{p-1}, \Delta) \mid (t_1, t_2, \Delta) \in G(p-1)\} \\
 &\quad \cup \{(t, t + 2^{p-1}, 2^p) \mid 0 \leq t < 2^{p-1}\}
 \end{aligned}$$

- **Variables:**

- α_{t_1, t_2} : Terminals t_1 and t_2 have been assigned different output labels.
- $b_{t, \Delta}$: The sequence of Δ labels starting from terminal node t (inclusive) cannot be divided into two equal sub-sequences.
- $r_{t_1, t_2, \Delta}$: The sequence of Δ labels starting from terminal node t_1 is equal to the sequence of Δ labels starting from terminal node t_2 (both inclusive).

Size Optimization

- **Hard clauses:**

$$\begin{aligned}
 (\neg c_{t_1,l}, \neg c_{t_2,l}, \neg \sigma_{t_1,t_2}) & \quad (t_1, t_2, \Delta) \in G(s_{max}), l \in K \\
 (\neg c_{t_1,l}, c_{t_2,l}, \sigma_{t_1,t_2}) & \quad (t_1, t_2, \Delta) \in G(s_{max}), l \in K \\
 (c_{t_1,l}, \neg c_{t_2,l}, \sigma_{t_1,t_2}) & \quad (t_1, t_2, \Delta) \in G(s_{max}), l \in K \\
 (b_{\Delta \lfloor t_1/\Delta \rfloor, \Delta}, \neg \sigma_{t_1,t_2}) & \quad (t_1, t_2, \Delta) \in G(s_{max}) \\
 (\neg r_{t_1\Delta, t_2\Delta, \Delta}, \neg c_{t_1\Delta+\delta, l}, c_{t_2\Delta+\delta, l}) & \quad \Delta \in P(s_{max}), t_1 < t_2 < 2^{s_{max}}/\Delta, \delta < \Delta, l \in K \\
 (\neg r_{t_1\Delta, t_2\Delta, \Delta}, c_{t_1\Delta+\delta, l}, \neg c_{t_2\Delta+\delta, l}) & \quad \Delta \in P(s_{max}), t_1 < t_2 < 2^{s_{max}}/\Delta, \delta < \Delta, l \in K
 \end{aligned}$$

- **Soft clauses:**

$$\left(\bigvee_{0 \leq t_2 < t} r_{t_2\Delta, t\Delta, \Delta}, \neg b_{t\Delta, \Delta} \right) \quad \Delta \in P(s_{max}), t < 2^{s_{max}}/\Delta$$

- **Definitions:**

$$\begin{aligned}
 G(1) &= \{(0, 1, 2)\} \\
 G(p) &= G(p-1) \cup \{(t_1 + 2^{p-1}, t_2 + 2^{p-1}, \Delta) \mid (t_1, t_2, \Delta) \in G(p-1)\} \\
 &\quad \cup \{(t, t + 2^{p-1}, 2^p) \mid 0 \leq t < 2^{p-1}\}
 \end{aligned}$$

- **Variables:**

- α_{t_1, t_2} : Terminals t_1 and t_2 have been assigned different output labels.
- $b_{t, \Delta}$: The sequence of Δ labels starting from terminal node t (inclusive) cannot be divided into two equal sub-sequences.
- $r_{t_1, t_2, \Delta}$: The sequence of Δ labels starting from terminal node t_1 is equal to the sequence of Δ labels starting from terminal node t_2 (both inclusive).

Size Optimization

- **Hard clauses:**

$$\begin{aligned}
 (\neg c_{t_1,l}, \neg c_{t_2,l}, \neg \sigma_{t_1,t_2}) & \quad (t_1, t_2, \Delta) \in G(s_{max}), l \in K \\
 (\neg c_{t_1,l}, c_{t_2,l}, \sigma_{t_1,t_2}) & \quad (t_1, t_2, \Delta) \in G(s_{max}), l \in K \\
 (c_{t_1,l}, \neg c_{t_2,l}, \sigma_{t_1,t_2}) & \quad (t_1, t_2, \Delta) \in G(s_{max}), l \in K \\
 (b_{\Delta \lfloor t_1/\Delta \rfloor, \Delta}, \neg \sigma_{t_1,t_2}) & \quad (t_1, t_2, \Delta) \in G(s_{max}) \\
 (\neg r_{t_1\Delta, t_2\Delta, \Delta}, \neg c_{t_1\Delta+\delta, l}, c_{t_2\Delta+\delta, l}) & \quad \Delta \in P(s_{max}), t_1 < t_2 < 2^{s_{max}}/\Delta, \delta < \Delta, l \in K \\
 (\neg r_{t_1\Delta, t_2\Delta, \Delta}, c_{t_1\Delta+\delta, l}, \neg c_{t_2\Delta+\delta, l}) & \quad \Delta \in P(s_{max}), t_1 < t_2 < 2^{s_{max}}/\Delta, \delta < \Delta, l \in K
 \end{aligned}$$

- **Soft clauses:**

$$\left(\bigvee_{0 \leq t_2 < t} r_{t_2\Delta, t\Delta, \Delta}, \neg b_{t\Delta, \Delta} \right) \quad \Delta \in P(s_{max}), t < 2^{s_{max}}/\Delta$$

- **Definitions:**

$$\begin{aligned}
 G(1) &= \{(0, 1, 2)\} \\
 G(p) &= G(p-1) \cup \{(t_1 + 2^{p-1}, t_2 + 2^{p-1}, \Delta) \mid (t_1, t_2, \Delta) \in G(p-1)\} \\
 &\quad \cup \{(t, t + 2^{p-1}, 2^p) \mid 0 \leq t < 2^{p-1}\}
 \end{aligned}$$

- **Variables:**

- α_{t_1, t_2} : Terminals t_1 and t_2 have been assigned different output labels.
- $b_{t, \Delta}$: The sequence of Δ labels starting from terminal node t (inclusive) cannot be divided into two equal sub-sequences.
- $r_{t_1, t_2, \Delta}$: The sequence of Δ labels starting from terminal node t_1 is equal to the sequence of Δ labels starting from terminal node t_2 (both inclusive).

◦ The differences in labels are correctly represented.

Size Optimization

- **Hard clauses:**

$$(\neg c_{t_1,l}, \neg c_{t_2,l}, \neg \sigma_{t_1,t_2}) \quad (t_1, t_2, \Delta) \in G(s_{max}), l \in K$$

$$(\neg c_{t_1,l}, c_{t_2,l}, \sigma_{t_1,t_2}) \quad (t_1, t_2, \Delta) \in G(s_{max}), l \in K$$

$$(c_{t_1,l}, \neg c_{t_2,l}, \sigma_{t_1,t_2}) \quad (t_1, t_2, \Delta) \in G(s_{max}), l \in K$$

$$(b_{\Delta \lfloor t_1/\Delta \rfloor, \Delta}, \neg \sigma_{t_1,t_2}) \quad (t_1, t_2, \Delta) \in G(s_{max})$$

$$(\neg r_{t_1\Delta, t_2\Delta, \Delta}, \neg c_{t_1\Delta+\delta, l}, c_{t_2\Delta+\delta, l}) \quad \Delta \in P(s_{max}), t_1 < t_2 < 2^{s_{max}}/\Delta, \delta < \Delta, l \in K$$

$$(\neg r_{t_1\Delta, t_2\Delta, \Delta}, c_{t_1\Delta+\delta, l}, \neg c_{t_2\Delta+\delta, l}) \quad \Delta \in P(s_{max}), t_1 < t_2 < 2^{s_{max}}/\Delta, \delta < \Delta, l \in K$$

- **Soft clauses:**

$$\left(\bigvee_{0 \leq t_2 < t} r_{t_2\Delta, t\Delta, \Delta}, \neg b_{t\Delta, \Delta} \right) \quad \Delta \in P(s_{max}), t < 2^{s_{max}}/\Delta$$

- **Definitions:**

$$G(1) = \{(0, 1, 2)\}$$

$$G(p) = G(p-1) \cup \{(t_1 + 2^{p-1}, t_2 + 2^{p-1}, \Delta) \mid (t_1, t_2, \Delta) \in G(p-1)\} \\ \cup \{(t, t + 2^{p-1}, 2^p) \mid 0 \leq t < 2^{p-1}\}$$

- **Variables:**

- α_{t_1, t_2} : Terminals t_1 and t_2 have been assigned different output labels.
- $b_{t, \Delta}$: The sequence of Δ labels starting from terminal node t (inclusive) cannot be divided into two equal sub-sequences.
- $r_{t_1, t_2, \Delta}$: The sequence of Δ labels starting from terminal node t_1 is equal to the sequence of Δ labels starting from terminal node t_2 (both inclusive).

- The differences in labels are correctly represented.

- A node cannot be replaced by one of its children if their descendants are labelled differently.

Size Optimization

- **Hard clauses:**

$$(\neg c_{t_1,l}, \neg c_{t_2,l}, \neg \sigma_{t_1,t_2}) \quad (t_1, t_2, \Delta) \in G(s_{max}), l \in K$$

$$(\neg c_{t_1,l}, c_{t_2,l}, \sigma_{t_1,t_2}) \quad (t_1, t_2, \Delta) \in G(s_{max}), l \in K$$

$$(c_{t_1,l}, \neg c_{t_2,l}, \sigma_{t_1,t_2}) \quad (t_1, t_2, \Delta) \in G(s_{max}), l \in K$$

$$(b_{\Delta \lfloor t_1/\Delta \rfloor, \Delta}, \neg \sigma_{t_1,t_2}) \quad (t_1, t_2, \Delta) \in G(s_{max})$$

$$(\neg r_{t_1\Delta, t_2\Delta, \Delta}, \neg c_{t_1\Delta+\delta, l}, c_{t_2\Delta+\delta, l}) \quad \Delta \in P(s_{max}), t_1 < t_2 < 2^{s_{max}}/\Delta, \delta < \Delta, l \in K$$

$$(\neg r_{t_1\Delta, t_2\Delta, \Delta}, c_{t_1\Delta+\delta, l}, \neg c_{t_2\Delta+\delta, l}) \quad \Delta \in P(s_{max}), t_1 < t_2 < 2^{s_{max}}/\Delta, \delta < \Delta, l \in K$$

- **Soft clauses:**

$$\left(\bigvee_{0 \leq t_2 < t} r_{t_2\Delta, t\Delta, \Delta}, \neg b_{t\Delta, \Delta} \right) \quad \Delta \in P(s_{max}), t < 2^{s_{max}}/\Delta$$

- **Definitions:**

$$G(1) = \{(0, 1, 2)\}$$

$$G(p) = G(p-1) \cup \{(t_1 + 2^{p-1}, t_2 + 2^{p-1}, \Delta) \mid (t_1, t_2, \Delta) \in G(p-1)\} \\ \cup \{(t, t + 2^{p-1}, 2^p) \mid 0 \leq t < 2^{p-1}\}$$

- **Variables:**

- α_{t_1, t_2} : Terminals t_1 and t_2 have been assigned different output labels.
- $b_{t, \Delta}$: The sequence of Δ labels starting from terminal node t (inclusive) cannot be divided into two equal sub-sequences.
- $r_{t_1, t_2, \Delta}$: The sequence of Δ labels starting from terminal node t_1 is equal to the sequence of Δ labels starting from terminal node t_2 (both inclusive).

- The differences in labels are correctly represented.

- A node cannot be replaced by one of its children if their descendants are labelled differently.

- A node cannot be merged with one of the previous ones if their descendants are labelled differently.

Size Optimization

- **Hard clauses:**

$$\begin{aligned} (\neg c_{t_1,l}, \neg c_{t_2,l}, \neg \sigma_{t_1,t_2}) & \quad (t_1, t_2, \Delta) \in G(s_{max}), l \in K \\ (\neg c_{t_1,l}, c_{t_2,l}, \sigma_{t_1,t_2}) & \quad (t_1, t_2, \Delta) \in G(s_{max}), l \in K \\ (c_{t_1,l}, \neg c_{t_2,l}, \sigma_{t_1,t_2}) & \quad (t_1, t_2, \Delta) \in G(s_{max}), l \in K \\ (b_{\Delta \lfloor t_1/\Delta \rfloor, \Delta}, \neg \sigma_{t_1,t_2}) & \quad (t_1, t_2, \Delta) \in G(s_{max}) \\ (\neg r_{t_1\Delta, t_2\Delta, \Delta}, \neg c_{t_1\Delta+\delta, l}, c_{t_2\Delta+\delta, l}) & \quad \Delta \in P(s_{max}), t_1 < t_2 < 2^{s_{max}}/\Delta, \delta < \Delta, l \in K \\ (\neg r_{t_1\Delta, t_2\Delta, \Delta}, c_{t_1\Delta+\delta, l}, \neg c_{t_2\Delta+\delta, l}) & \quad \Delta \in P(s_{max}), t_1 < t_2 < 2^{s_{max}}/\Delta, \delta < \Delta, l \in K \end{aligned}$$

- **Soft clauses:**

$$\left(\bigvee_{0 \leq t_2 < t} r_{t_2\Delta, t\Delta, \Delta}, \neg b_{t\Delta, \Delta} \right) \quad \Delta \in P(s_{max}), t < 2^{s_{max}}/\Delta$$

- **Definitions:**

$$\begin{aligned} G(1) &= \{(0, 1, 2)\} \\ G(p) &= G(p-1) \cup \{(t_1 + 2^{p-1}, t_2 + 2^{p-1}, \Delta) \mid (t_1, t_2, \Delta) \in G(p-1)\} \\ &\quad \cup \{(t, t + 2^{p-1}, 2^p) \mid 0 \leq t < 2^{p-1}\} \end{aligned}$$

- **Variables:**

- α_{t_1, t_2} : Terminals t_1 and t_2 have been assigned different output labels.
- $b_{t, \Delta}$: The sequence of Δ labels starting from terminal node t (inclusive) cannot be divided into two equal sub-sequences.
- $r_{t_1, t_2, \Delta}$: The sequence of Δ labels starting from terminal node t_1 is equal to the sequence of Δ labels starting from terminal node t_2 (both inclusive).

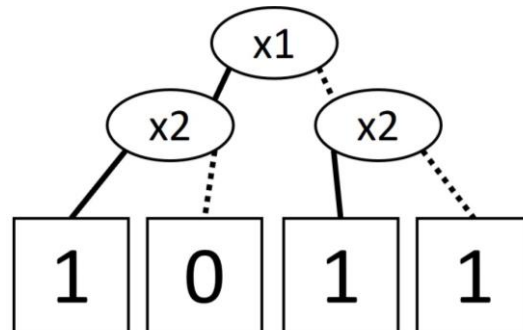
- The differences in labels are correctly represented.
- A node cannot be replaced by one of its children if their descendants are labelled differently.
- A node cannot be merged with one of the previous ones if their descendants are labelled differently.
- The number of nodes that can be replaced or merged are maximized.

Multi-dimensional BDD

- **Multi-dimensional splits:**
 - Specified by **directional inner BDDs (DIBDD)** rather feature threshold pairs

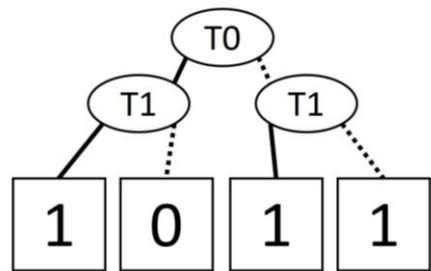
Multi-dimensional BDD

- **Multi-dimensional splits:**
 - Specified by **directional inner BDDs (DIBDD)** rather feature threshold pairs
- **DIBDD** with dimension D :
 - Operates on D **ordinary splits**
 - Has two terminals, representing **left (1)** and **right (0)** directions



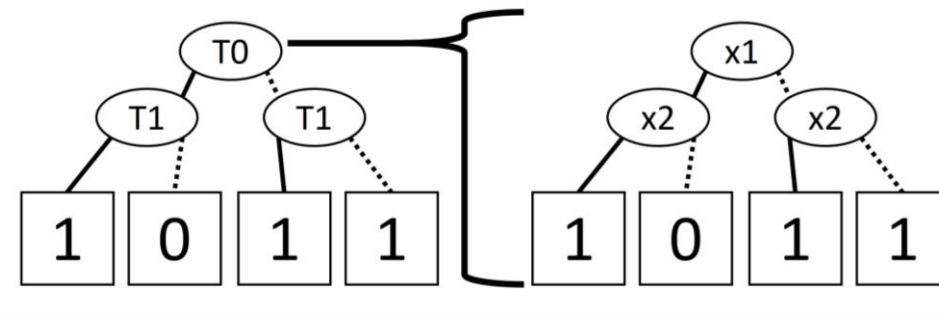
Multi-dimensional BDD

- **Multi-dimensional BDDs (MDBDD):**
 - Operates on multi-dimensional splits rather than ordinary ones



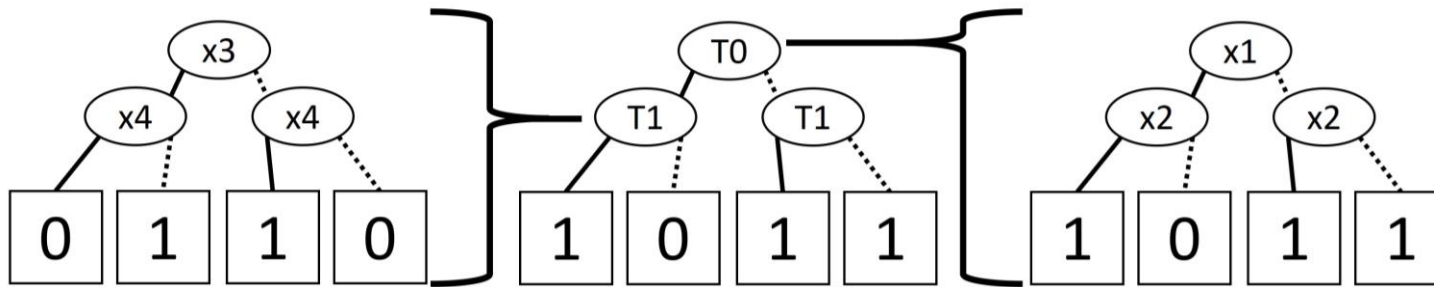
Multi-dimensional BDD

- **Multi-dimensional BDDs (MDBDD):**
 - Operates on multi-dimensional splits rather than ordinary ones



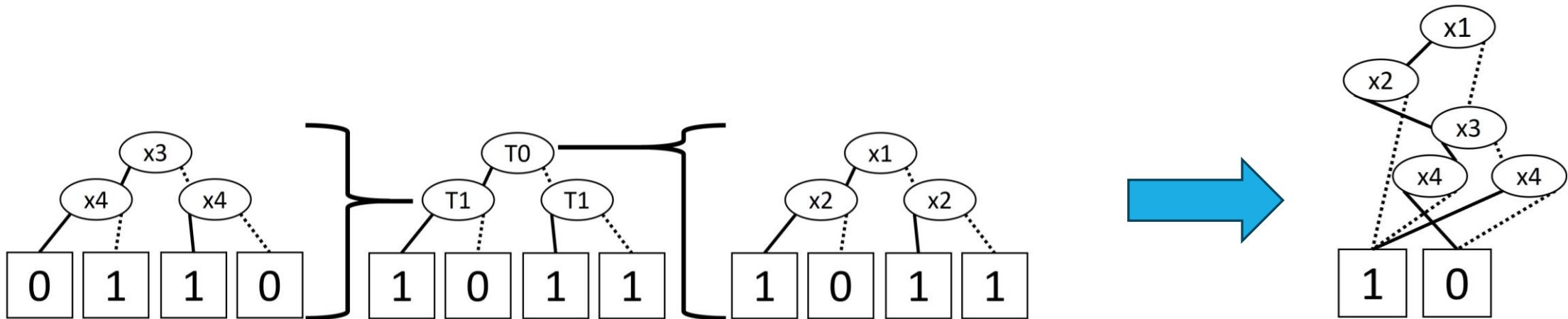
Multi-dimensional BDD

- **Multi-dimensional BDDs (MDBDD):**
 - Operates on multi-dimensional splits rather than ordinary ones



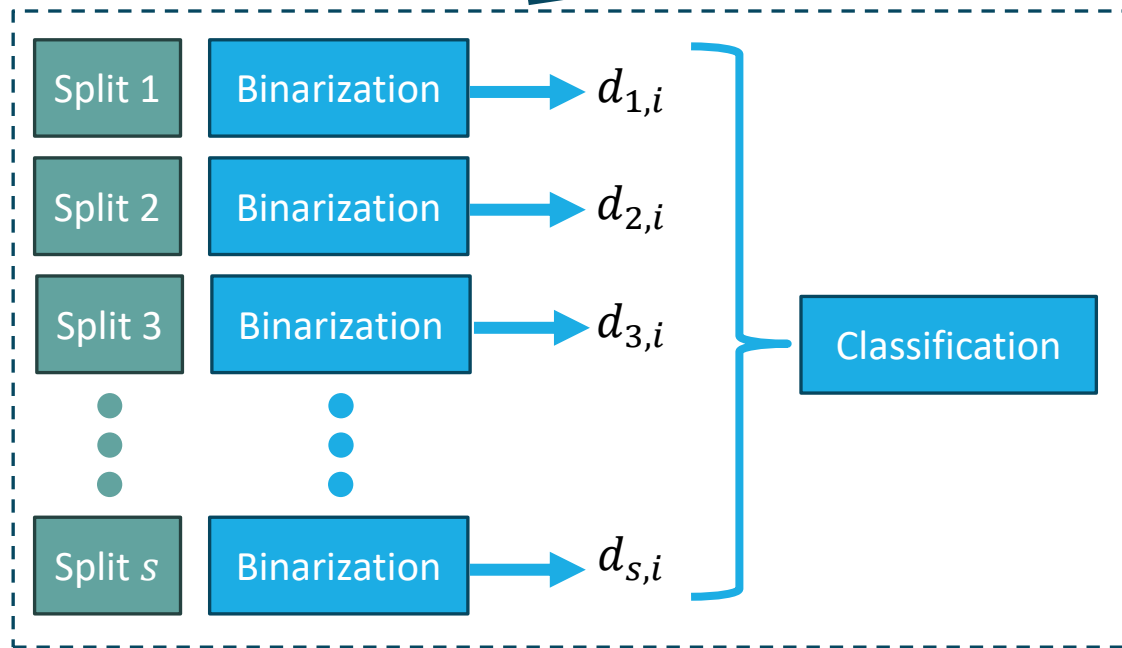
Multi-dimensional BDD

- **Multi-dimensional BDDs (MDBDD):**
 - Operates on multi-dimensional splits rather than ordinary ones
 - Can be **transformed** into ordinary BDDs

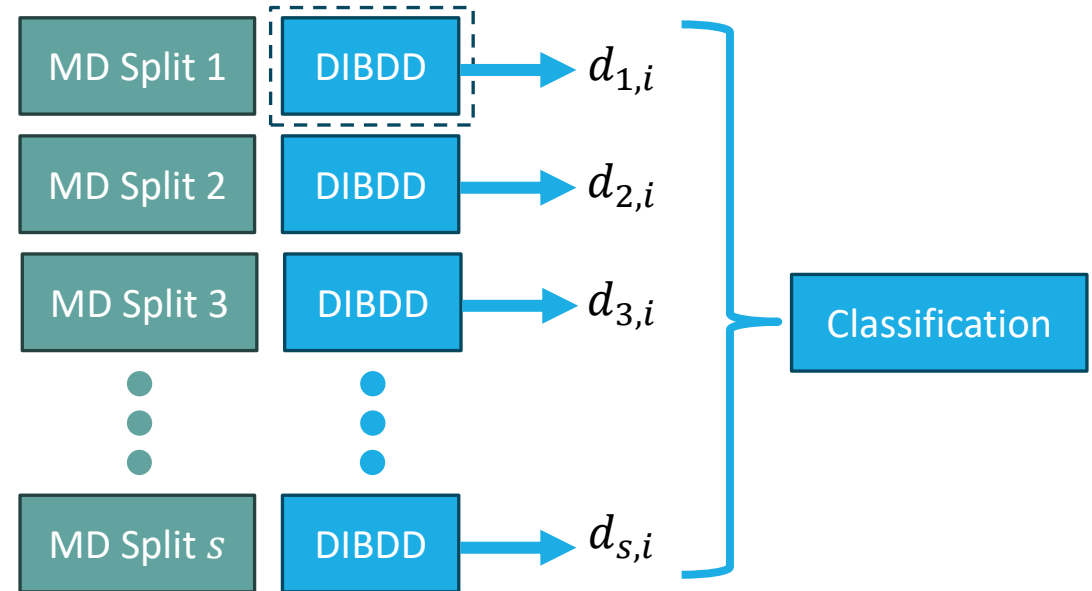


Multi-dimensional BDD

◦ BDD encoding:



◦ MDBDD encoding:



Multi-dimensional BDD

- Kept **hard clauses**: enforce labels and correct classification
- Kept **soft clauses**: maximize correct classification
- New **hard clauses**:

$$(\hat{a}_{(s,h),j}, \neg \hat{a}_{(s,h),j+1}) \quad (s,h) \in S_H, j \in F$$

$$(\hat{a}_{(s,h),0}) \quad (s,h) \in S_H$$

$$(\neg \hat{a}_{(s,h),j}, \hat{a}_{(s,h),j+1}, \hat{d}_{(s,h),i_1}, \neg \hat{d}_{(s,h),i_2}) \quad (s,h) \in S_H, j \in F, (i_1, i_2) \in O_j(X)$$

$$(\neg \hat{a}_{(s,h),j}, \hat{a}_{(s,h),j+1}, \neg \hat{d}_{(s,h),i_1}, \hat{d}_{(s,h),i_2}) \quad (s,h) \in S_H, j \in F, (i_1, i_2) \in O_j^-(X)$$

$$(\neg \hat{a}_{(s,h),j}, \hat{a}_{(s,h),j+1}, \hat{d}_{(s,h),\#j^1}) \quad (s,h) \in S_H, j \in F$$

$$\left(\bigvee_{h \in A_R(s,t)} \hat{d}_{(s,h),i}, \bigvee_{h \in A_L(s,t)} \neg \hat{d}_{(s,h),i}, d_{s,i}, \neg \hat{c}_{s,t} \right) \quad s \in S, x_i \in X, t \in \mathcal{N}_T^s$$

$$\left(\bigvee_{h \in A_R(s,t)} \hat{d}_{(s,h),i}, \bigvee_{h \in A_L(s,t)} \neg \hat{d}_{(s,h),i}, \neg d_{s,i}, \hat{c}_{s,t} \right) \quad s \in S, x_i \in X, t \in \mathcal{N}_T^s$$

$$(\hat{c}_{s,0}) \quad s \in S$$

Variables:

- $\hat{a}_{(s,h),j}$: The feature chosen at split h of directional inner BDD s is or comes before j .
- $\hat{d}_{(s,h),i}$: Point x_i is directed to left at split h of directional inner BDD s .
- $\hat{c}_{s,t}$: Terminal t of directional inner BDD s is assigned the label 1.

Multi-dimensional BDD

- Kept **hard clauses**: enforce labels and correct classification
- Kept **soft clauses**: maximize correct classification
- New **hard clauses**:

$$(\hat{a}_{(s,h),j}, \neg \hat{a}_{(s,h),j+1}) \quad (s,h) \in S_H, j \in F$$

$$(\hat{a}_{(s,h),0}) \quad (s,h) \in S_H$$

$$(\neg \hat{a}_{(s,h),j}, \hat{a}_{(s,h),j+1}, \hat{d}_{(s,h),i_1}, \neg \hat{d}_{(s,h),i_2}) \quad (s,h) \in S_H, j \in F, (i_1, i_2) \in O_j(X)$$

$$(\neg \hat{a}_{(s,h),j}, \hat{a}_{(s,h),j+1}, \neg \hat{d}_{(s,h),i_1}, \hat{d}_{(s,h),i_2}) \quad (s,h) \in S_H, j \in F, (i_1, i_2) \in O_j^-(X)$$

$$(\neg \hat{a}_{(s,h),j}, \hat{a}_{(s,h),j+1}, \hat{d}_{(s,h),\#j^1}) \quad (s,h) \in S_H, j \in F$$

$$\left(\bigvee_{h \in A_R(s,t)} \hat{d}_{(s,h),i}, \bigvee_{h \in A_L(s,t)} \neg \hat{d}_{(s,h),i}, d_{s,i}, \neg \hat{c}_{s,t} \right) \quad s \in S, x_i \in X, t \in \mathcal{N}_T^s$$

$$\left(\bigvee_{h \in A_R(s,t)} \hat{d}_{(s,h),i}, \bigvee_{h \in A_L(s,t)} \neg \hat{d}_{(s,h),i}, \neg d_{s,i}, \hat{c}_{s,t} \right) \quad s \in S, x_i \in X, t \in \mathcal{N}_T^s$$

$$(\hat{c}_{s,0}) \quad s \in S$$

Variables:

- $\hat{a}_{(s,h),j}$: The feature chosen at split h of directional inner BDD s is or comes before j .
- $\hat{d}_{(s,h),i}$: Point x_i is directed to left at split h of directional inner BDD s .
- $\hat{c}_{s,t}$: Terminal t of directional inner BDD s is assigned the label 1.

Multi-dimensional BDD

- Kept **hard clauses**: enforce labels and correct classification
- Kept **soft clauses**: maximize correct classification
- New **hard clauses**:

$$(\hat{a}_{(s,h),j}, \neg \hat{a}_{(s,h),j+1}) \quad (s,h) \in S_H, j \in F$$

$$(\hat{a}_{(s,h),0}) \quad (s,h) \in S_H$$

$$(\neg \hat{a}_{(s,h),j}, \hat{a}_{(s,h),j+1}, \hat{d}_{(s,h),i_1}, \neg \hat{d}_{(s,h),i_2}) \quad (s,h) \in S_H, j \in F, (i_1, i_2) \in O_j(X)$$

$$(\neg \hat{a}_{(s,h),j}, \hat{a}_{(s,h),j+1}, \neg \hat{d}_{(s,h),i_1}, \hat{d}_{(s,h),i_2}) \quad (s,h) \in S_H, j \in F, (i_1, i_2) \in O_j^-(X)$$

$$(\neg \hat{a}_{(s,h),j}, \hat{a}_{(s,h),j+1}, \hat{d}_{(s,h),\#j^1}) \quad (s,h) \in S_H, j \in F$$

$$\left(\bigvee_{h \in A_R(s,t)} \hat{d}_{(s,h),i}, \bigvee_{h \in A_L(s,t)} \neg \hat{d}_{(s,h),i}, d_{s,i}, \neg \hat{c}_{s,t} \right) \quad s \in S, x_i \in X, t \in \mathcal{N}_T^s$$

$$\left(\bigvee_{h \in A_R(s,t)} \hat{d}_{(s,h),i}, \bigvee_{h \in A_L(s,t)} \neg \hat{d}_{(s,h),i}, \neg d_{s,i}, \hat{c}_{s,t} \right) \quad s \in S, x_i \in X, t \in \mathcal{N}_T^s$$

$$(\hat{c}_{s,0}) \quad s \in S$$

Variables:

- $\hat{a}_{(s,h),j}$: The feature chosen at split h of directional inner BDD s is or comes before j .

- $\hat{d}_{(s,h),i}$: Point x_i is directed to left at split h of directional inner BDD s .

- $\hat{c}_{s,t}$: Terminal t of directional inner BDD s is assigned the label 1.

Multi-dimensional BDD

- Kept **hard clauses**: enforce labels and correct classification
- Kept **soft clauses**: maximize correct classification
- New **hard clauses**:

$$(\hat{a}_{(s,h),j}, \neg \hat{a}_{(s,h),j+1}) \quad (s,h) \in S_H, j \in F$$

$$(\hat{a}_{(s,h),0}) \quad (s,h) \in S_H$$

$$(\neg \hat{a}_{(s,h),j}, \hat{a}_{(s,h),j+1}, \hat{d}_{(s,h),i_1}, \neg \hat{d}_{(s,h),i_2}) \quad (s,h) \in S_H, j \in F, (i_1, i_2) \in O_j(X)$$

$$(\neg \hat{a}_{(s,h),j}, \hat{a}_{(s,h),j+1}, \neg \hat{d}_{(s,h),i_1}, \hat{d}_{(s,h),i_2}) \quad (s,h) \in S_H, j \in F, (i_1, i_2) \in O_j^-(X)$$

$$(\neg \hat{a}_{(s,h),j}, \hat{a}_{(s,h),j+1}, \hat{d}_{(s,h),\#j^1}) \quad (s,h) \in S_H, j \in F$$

$$\left(\bigvee_{h \in A_R(s,t)} \hat{d}_{(s,h),i}, \bigvee_{h \in A_L(s,t)} \neg \hat{d}_{(s,h),i}, d_{s,i}, \neg \hat{c}_{s,t} \right) \quad s \in S, x_i \in X, t \in \mathcal{N}_T^s$$

$$\left(\bigvee_{h \in A_R(s,t)} \hat{d}_{(s,h),i}, \bigvee_{h \in A_L(s,t)} \neg \hat{d}_{(s,h),i}, \neg d_{s,i}, \hat{c}_{s,t} \right) \quad s \in S, x_i \in X, t \in \mathcal{N}_T^s$$

$$(\hat{c}_{s,0}) \quad s \in S$$

Variables:

- $\hat{a}_{(s,h),j}$: The feature chosen at split h of directional inner BDD s is or comes before j .
- $\hat{d}_{(s,h),i}$: Point x_i is directed to left at split h of directional inner BDD s .
- $\hat{c}_{s,t}$: Terminal t of directional inner BDD s is assigned the label 1.

Multi-dimensional BDD

- Kept **hard clauses**: enforce labels and correct classification
- Kept **soft clauses**: maximize correct classification
- New **hard clauses**:

$$(\hat{a}_{(s,h),j}, \neg \hat{a}_{(s,h),j+1}) \quad (s,h) \in S_H, j \in F$$

$$(\hat{a}_{(s,h),0}) \quad (s,h) \in S_H$$

$$(\neg \hat{a}_{(s,h),j}, \hat{a}_{(s,h),j+1}, \hat{d}_{(s,h),i_1}, \neg \hat{d}_{(s,h),i_2}) \quad (s,h) \in S_H, j \in F, (i_1, i_2) \in O_j(X)$$

$$(\neg \hat{a}_{(s,h),j}, \hat{a}_{(s,h),j+1}, \neg \hat{d}_{(s,h),i_1}, \hat{d}_{(s,h),i_2}) \quad (s,h) \in S_H, j \in F, (i_1, i_2) \in O_j^-(X)$$

$$(\neg \hat{a}_{(s,h),j}, \hat{a}_{(s,h),j+1}, \hat{d}_{(s,h),\#j^1}) \quad (s,h) \in S_H, j \in F$$

$$\left(\bigvee_{h \in A_R(s,t)} \hat{d}_{(s,h),i}, \bigvee_{h \in A_L(s,t)} \neg \hat{d}_{(s,h),i}, d_{s,i}, \neg \hat{c}_{s,t} \right) \quad s \in S, x_i \in X, t \in \mathcal{N}_T^s$$

$$\left(\bigvee_{h \in A_R(s,t)} \hat{d}_{(s,h),i}, \bigvee_{h \in A_L(s,t)} \neg \hat{d}_{(s,h),i}, \neg d_{s,i}, \hat{c}_{s,t} \right) \quad s \in S, x_i \in X, t \in \mathcal{N}_T^s$$

$$(\hat{c}_{s,0}) \quad s \in S$$

Variables:

- $\hat{a}_{(s,h),j}$: The feature chosen at split h of directional inner BDD s is or comes before j .
- $\hat{d}_{(s,h),i}$: Point x_i is directed to left at split h of directional inner BDD s .
- $\hat{c}_{s,t}$: Terminal t of directional inner BDD s is assigned the label 1.

- Exactly one feature is selected at each split of each DIBDD.

Multi-dimensional BDD

- Kept **hard clauses**: enforce labels and correct classification
- Kept **soft clauses**: maximize correct classification
- New **hard clauses**:

$$(\hat{a}_{(s,h),j}, \neg \hat{a}_{(s,h),j+1}) \quad (s,h) \in S_H, j \in F$$

$$(\hat{a}_{(s,h),0}) \quad (s,h) \in S_H$$

$$(\neg \hat{a}_{(s,h),j}, \hat{a}_{(s,h),j+1}, \hat{d}_{(s,h),i_1}, \neg \hat{d}_{(s,h),i_2}) \quad (s,h) \in S_H, j \in F, (i_1, i_2) \in O_j(X)$$

$$(\neg \hat{a}_{(s,h),j}, \hat{a}_{(s,h),j+1}, \neg \hat{d}_{(s,h),i_1}, \hat{d}_{(s,h),i_2}) \quad (s,h) \in S_H, j \in F, (i_1, i_2) \in O_j^-(X)$$

$$(\neg \hat{a}_{(s,h),j}, \hat{a}_{(s,h),j+1}, \hat{d}_{(s,h),\#_j^1}) \quad (s,h) \in S_H, j \in F$$

$$\left(\bigvee_{h \in A_R(s,t)} \hat{d}_{(s,h),i}, \bigvee_{h \in A_L(s,t)} \neg \hat{d}_{(s,h),i}, d_{s,i}, \neg \hat{c}_{s,t} \right) \quad s \in S, x_i \in X, t \in \mathcal{N}_T^s$$

$$\left(\bigvee_{h \in A_R(s,t)} \hat{d}_{(s,h),i}, \bigvee_{h \in A_L(s,t)} \neg \hat{d}_{(s,h),i}, \neg d_{s,i}, \hat{c}_{s,t} \right) \quad s \in S, x_i \in X, t \in \mathcal{N}_T^s$$

$$(\hat{c}_{s,0}) \quad s \in S$$

Variables:

- $\hat{a}_{(s,h),j}$: The feature chosen at split h of directional inner BDD s is or comes before j .
- $\hat{d}_{(s,h),i}$: Point x_i is directed to left at split h of directional inner BDD s .
- $\hat{c}_{s,t}$: Terminal t of directional inner BDD s is assigned the label 1.

- Exactly one feature is selected at each split of each DIBDD.

- Selected feature enforces order of values.

Multi-dimensional BDD

- Kept **hard clauses**: enforce labels and correct classification
- Kept **soft clauses**: maximize correct classification
- New **hard clauses**:

$$(\hat{a}_{(s,h),j}, \neg \hat{a}_{(s,h),j+1}) \quad (s,h) \in S_H, j \in F$$

$$(\hat{a}_{(s,h),0}) \quad (s,h) \in S_H$$

$$(\neg \hat{a}_{(s,h),j}, \hat{a}_{(s,h),j+1}, \hat{d}_{(s,h),i_1}, \neg \hat{d}_{(s,h),i_2}) \quad (s,h) \in S_H, j \in F, (i_1, i_2) \in O_j(X)$$

$$(\neg \hat{a}_{(s,h),j}, \hat{a}_{(s,h),j+1}, \neg \hat{d}_{(s,h),i_1}, \hat{d}_{(s,h),i_2}) \quad (s,h) \in S_H, j \in F, (i_1, i_2) \in O_j^-(X)$$

$$(\neg \hat{a}_{(s,h),j}, \hat{a}_{(s,h),j+1}, \hat{d}_{(s,h),\#j^1}) \quad (s,h) \in S_H, j \in F$$

$$\left(\bigvee_{h \in A_R(s,t)} \hat{d}_{(s,h),i}, \bigvee_{h \in A_L(s,t)} \neg \hat{d}_{(s,h),i}, d_{s,i}, \neg \hat{c}_{s,t} \right) \quad s \in S, x_i \in X, t \in \mathcal{N}_T^s$$

$$\left(\bigvee_{h \in A_R(s,t)} \hat{d}_{(s,h),i}, \bigvee_{h \in A_L(s,t)} \neg \hat{d}_{(s,h),i}, \neg d_{s,i}, \hat{c}_{s,t} \right) \quad s \in S, x_i \in X, t \in \mathcal{N}_T^s$$

$$(\hat{c}_{s,0}) \quad s \in S$$

Variables:

- $\hat{a}_{(s,h),j}$: The feature chosen at split h of directional inner BDD s is or comes before j .
- $\hat{d}_{(s,h),i}$: Point x_i is directed to left at split h of directional inner BDD s .
- $\hat{c}_{s,t}$: Terminal t of directional inner BDD s is assigned the label 1.

- Exactly one feature is selected at each split of each DIBDD.
- Selected feature enforces order of values.
- The value of a multi-dimensional split matches the label of the containing leaf in the DIBDD.

Multi-dimensional BDD

- Kept **hard clauses**: enforce labels and correct classification
- Kept **soft clauses**: maximize correct classification
- New hard clauses**:

$$(\hat{a}_{(s,h),j}, \neg \hat{a}_{(s,h),j+1}) \quad (s,h) \in S_H, j \in F$$

$$(\hat{a}_{(s,h),0}) \quad (s,h) \in S_H$$

$$(\neg \hat{a}_{(s,h),j}, \hat{a}_{(s,h),j+1}, \hat{d}_{(s,h),i_1}, \neg \hat{d}_{(s,h),i_2}) \quad (s,h) \in S_H, j \in F, (i_1, i_2) \in O_j(X)$$

$$(\neg \hat{a}_{(s,h),j}, \hat{a}_{(s,h),j+1}, \neg \hat{d}_{(s,h),i_1}, \hat{d}_{(s,h),i_2}) \quad (s,h) \in S_H, j \in F, (i_1, i_2) \in O_j^-(X)$$

$$(\neg \hat{a}_{(s,h),j}, \hat{a}_{(s,h),j+1}, \hat{d}_{(s,h),\#j^1}) \quad (s,h) \in S_H, j \in F$$

$$\left(\bigvee_{h \in A_R(s,t)} \hat{d}_{(s,h),i}, \bigvee_{h \in A_L(s,t)} \neg \hat{d}_{(s,h),i}, d_{s,i}, \neg \hat{c}_{s,t} \right) \quad s \in S, x_i \in X, t \in \mathcal{N}_T^s$$

$$\left(\bigvee_{h \in A_R(s,t)} \hat{d}_{(s,h),i}, \bigvee_{h \in A_L(s,t)} \neg \hat{d}_{(s,h),i}, \neg d_{s,i}, \hat{c}_{s,t} \right) \quad s \in S, x_i \in X, t \in \mathcal{N}_T^s$$

$$(\hat{c}_{s,0}) \quad s \in S$$

Variables:

- $\hat{a}_{(s,h),j}$: The feature chosen at split h of directional inner BDD s is or comes before j .
- $\hat{d}_{(s,h),i}$: Point x_i is directed to left at split h of directional inner BDD s .
- $\hat{c}_{s,t}$: Terminal t of directional inner BDD s is assigned the label 1.

- Exactly one feature is selected at each split of each DIBDD.
- Selected feature enforces order of values.
- The value of a multi-dimensional split matches the label of the containing leaf in the DIBDD.
- First leaf of each DIBDD is labelled 1 for breaking symmetry.

Expressiveness relations

- Given: $D_{total} = D_1 + D_2 + \dots + D_s$

Multi-dimension BDD
on s splits \leq BDDs on D_{total} splits

Expressiveness relations

- Given: $D_{total} = D_1 + D_2 + \dots + D_s$

BDDs on s splits \leq Multi-dimension BDD on s splits \leq BDDs on D_{total} splits

Background

Encodings

Experiments

- Objectives
- Setup
- Performance
- Size optimization
- Multi-dimensional BDD

Objectives

- Does our encoding outperform the state of the art in **runtime** and **training accuracy**?
- Do the 1-stage and 2-stage approaches **improve compactness** and how do they **affect testing accuracy**?
- Do multi-dimensional BDDs enable us to employ a **higher number of features** in our solutions?

Setup

- Baseline: **Hu et al.** [2022]
 - Learns **max-accuracy** BDDs with **binary classification** and **binary features**
- Datasets: **11** datasets from the **UCI** repository
- Solver: **Loandra** with 15 minutes time limit

Performance

- **Higher** training accuracies
- **Shorter** runtimes

Splits	Training Accuracy			Runtime / Timeout		
	Ours	Hu et al.	Tie	Ours	Hu et al.	Tie
4	2	0	6	5	0	3
5	3	1	4	3	0	5
6	3	0	5	3	0	5

Number of improvements compared to Hu et al.

Performance

- **Higher** training accuracies
- **Shorter** runtimes
- Improvements seen in both numerical and binary datasets

Splits	Training Accuracy			Runtime / Timeout		
	Ours	Hu et al.	Tie	Ours	Hu et al.	Tie
4	2	0	6	5	0	3
5	3	1	4	3	0	5
6	3	0	5	3	0	5

Number of improvements compared to Hu et al.

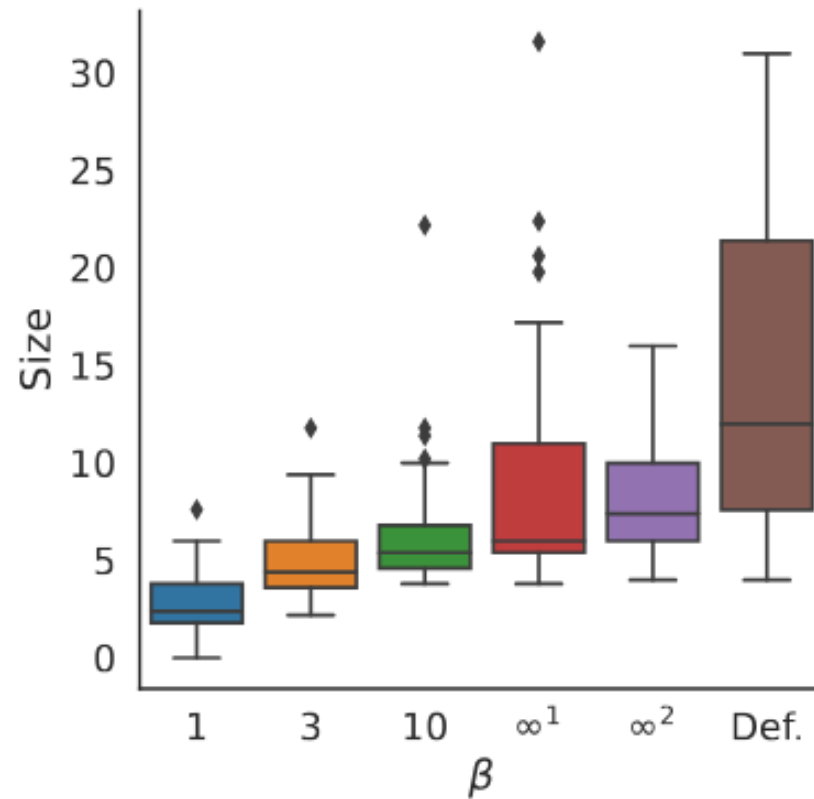
Dataset	Splits	Accuracy (%)		Time (s)	
		Ours	Hu et al. [16]	Ours	Hu et al. [16]
Ionosphere	4	94.9	90.6	TO	TO
$ X $ $ F $ $ K $	5	95.2	85.8	TO	TO
351 34 2	6	96.6	90.6	TO	TO
Monk2	4	74.6	74.6	85.49	473.43
$ X $ $ F $ $ K $	5	84.6	84.6	185.3	416.37
169 6 2	6	100	100	0.35	1.09

Size Optimization

- Use **5-fold cross validation** and study testing accuracy, training accuracy, and size
- Parameter β : the weight of accuracy against size in objective
 - $\beta \in \mathbb{R}$: Accuracy weighted (1-stage)
 - $\beta = \infty^1$: Accuracy prioritized (1-stage)
 - $\beta = \infty^2$: 2-stage approach
 - $\beta = \mathit{def}$: No size optimization

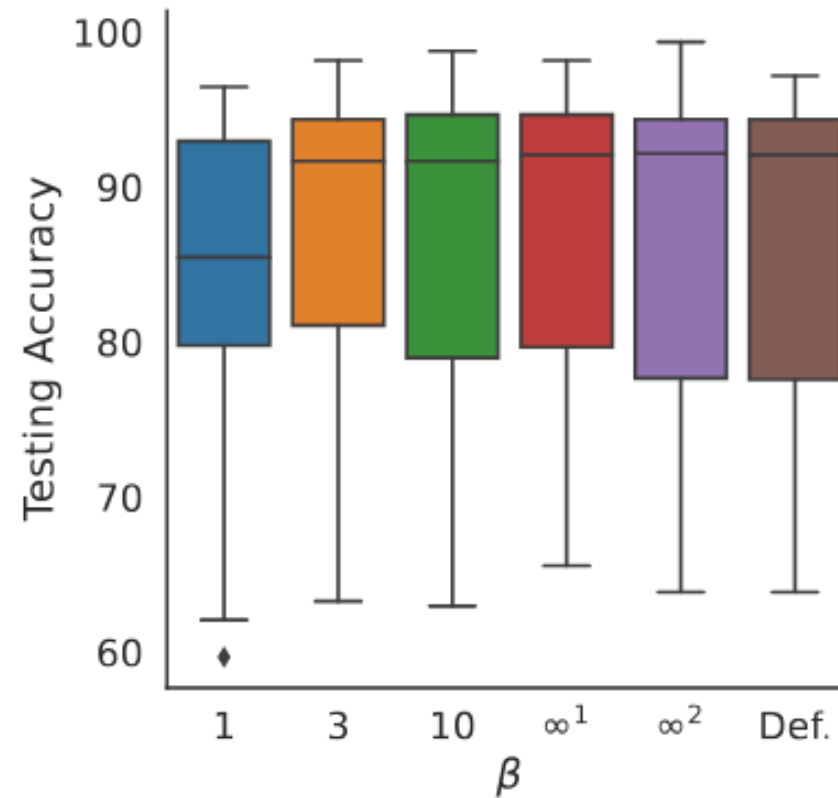
Size Optimization

- **1-stage** and **2-stage** approaches significantly improve compactness



Size Optimization

- Compactness helps with **testing accuracy** and **generalization**

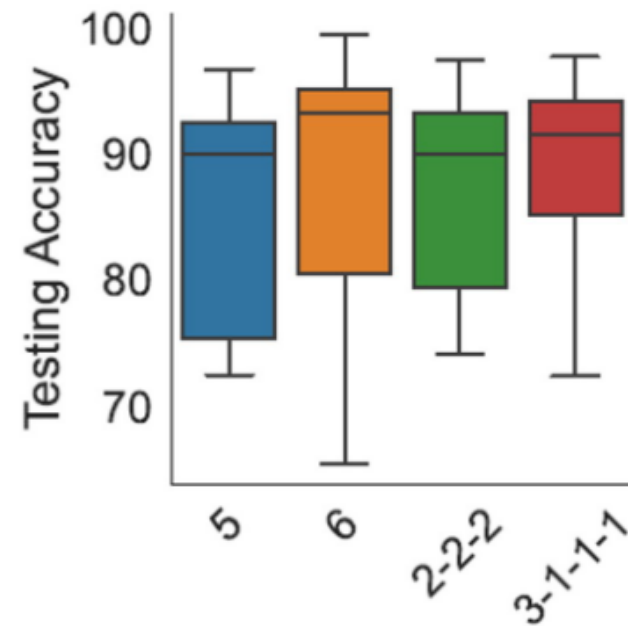


Multi-dimensional BDD

- **5-fold cross validation**
- **Sequence of dimensions** as parameter
- Utilize **2-stage** size optimization

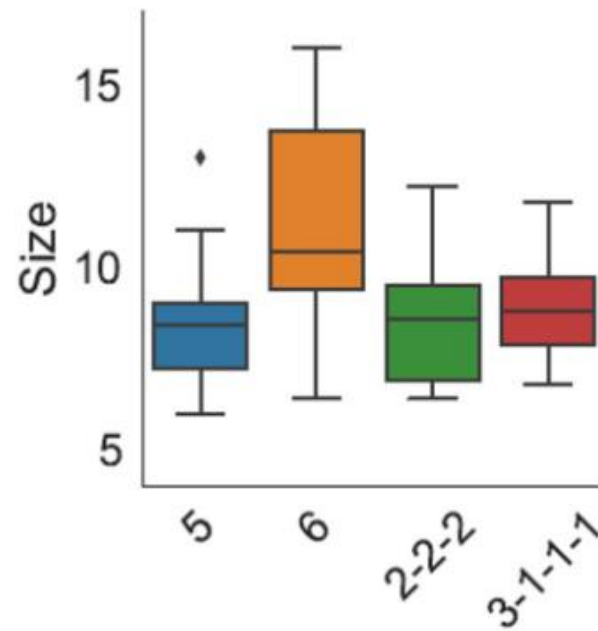
Multi-dimensional BDD

- Testing accuracy is close to **upper bound of expressiveness**



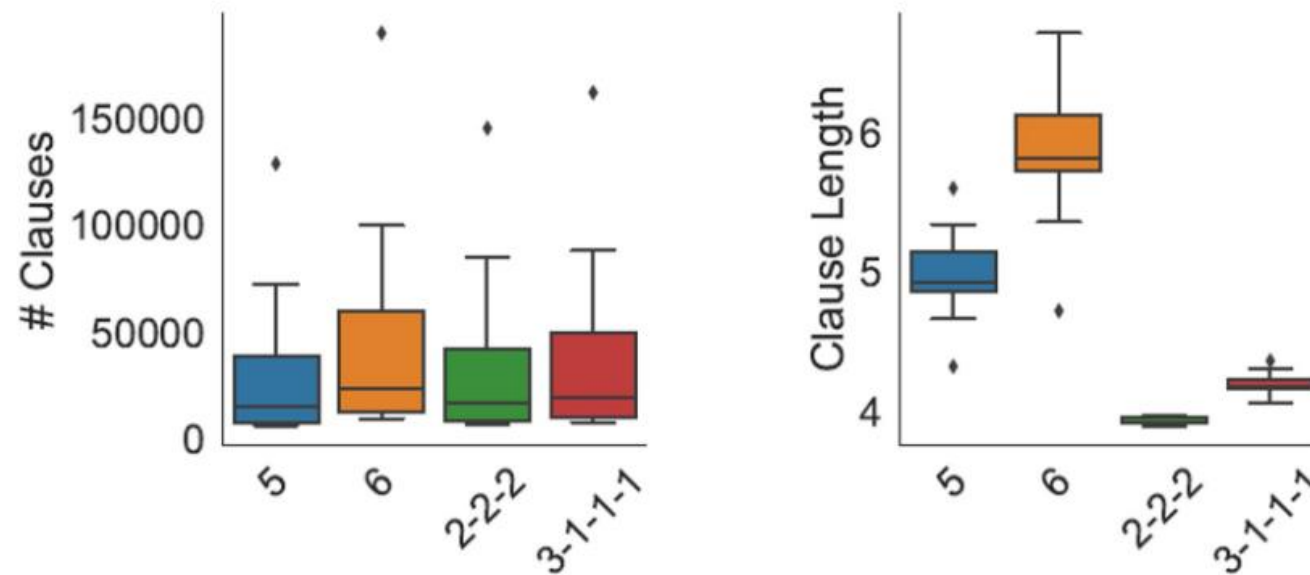
Multi-dimensional BDD

- Solutions are more compact than **upper bound of expressiveness**



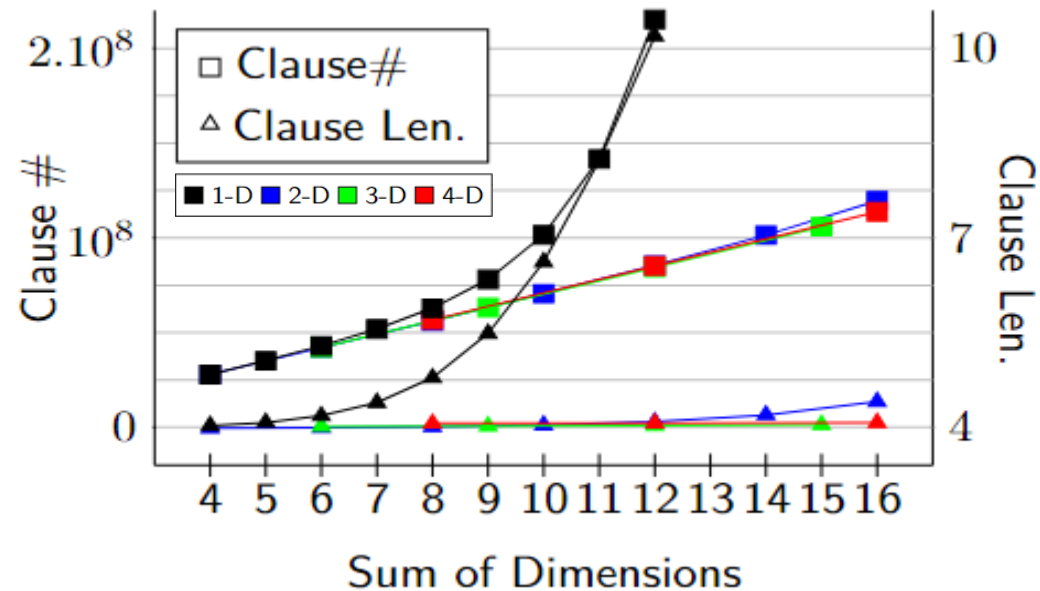
Multi-dimensional BDD

- **Clauses** are significantly **shorter**



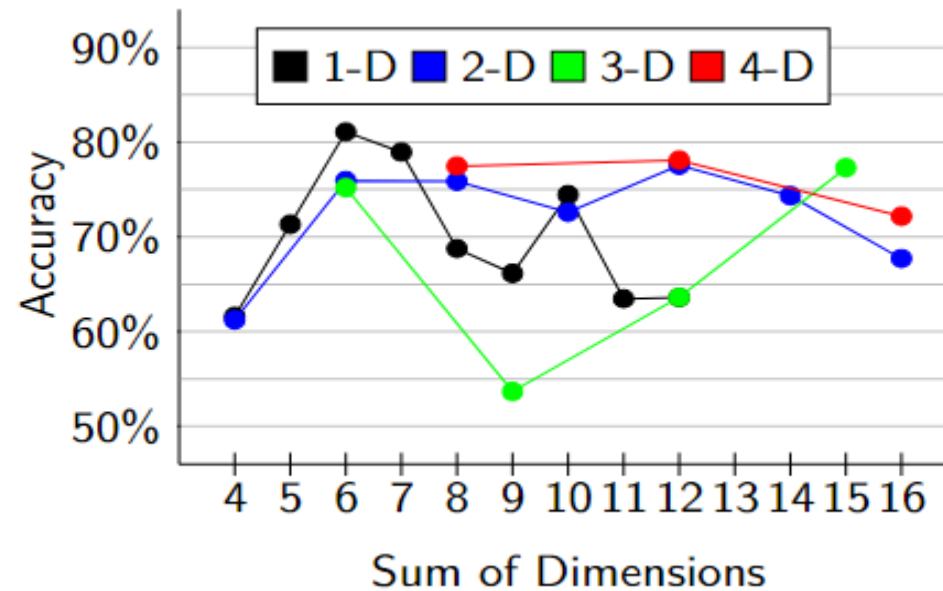
Multi-dimensional BDD

- Performance on **Adult dataset**
 - 32K points and 105 features
 - 60m timeout
- Number and length of clauses **scale significantly better**



Multi-dimensional BDD

- Performance on **Adult dataset**
 - 32K points and 105 features
 - 60m timeout
- Solutions are produced for **higher number of total dimensions**



Summary

- **SAT-based** encoding for learning **BDD classifiers**
 - Numerical feature encoding
 - Multi-label
- **BDD size optimization**
 - **1-stage** (as a secondary objective)
 - **2-stage** (as a post-processing step)
- **Multi-dimensional BDDs**
 - Directional inner BDDs and multi-dimensional splits
 - **Expressiveness** bounds
- Experiments:
 - **Better performance** than the state of the art
 - **Significant compactness** with little cost to accuracy
 - Better **generalization** through size reduction
 - **Multi-dimensional BDDs** to enable high number of total splits

Future Work

- Allow more control to **2-stage size optimization**
- Encode **nested** directional BDDs
- Investigate strategies to **balance the two objectives**
- Analyze the effects of the **dimension sequence parameter**

SAT-Based Learning of Compact Binary Decision Diagrams for Classification

Thank you for your time!

Q & A

POUYA SHATI, ELDAN COHEN, SHEILA MCILRAITH

UNIVERSITY OF TORONTO

VECTOR INSTITUTE FOR ARTIFICIAL INTELLIGENCE

CP-23

AUGUST 2023