GOSDT

Mose Park

Department of Statistical Data Science
University of Seoul

Selective. Lab

March 12, 2024

Index

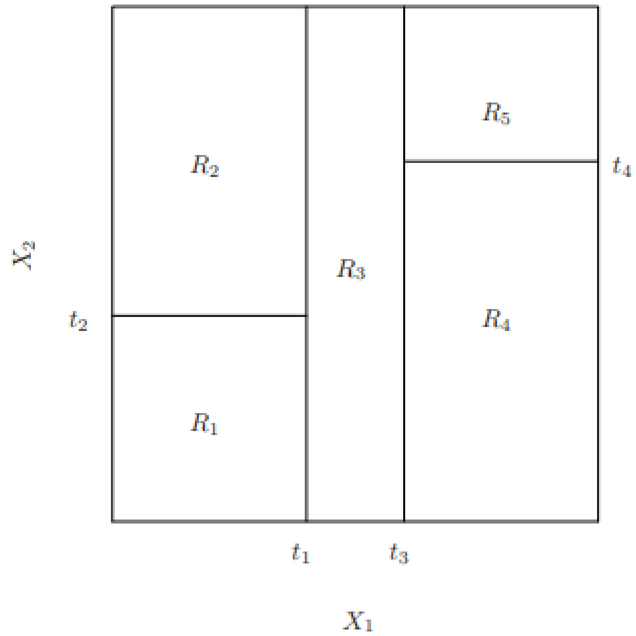
Overview

Optimal Decision Tree

Three Guesses

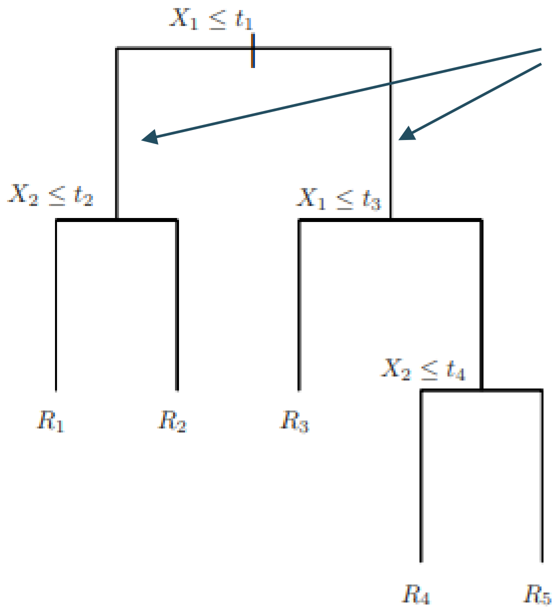
Model Performance

CART vs Optimal DT



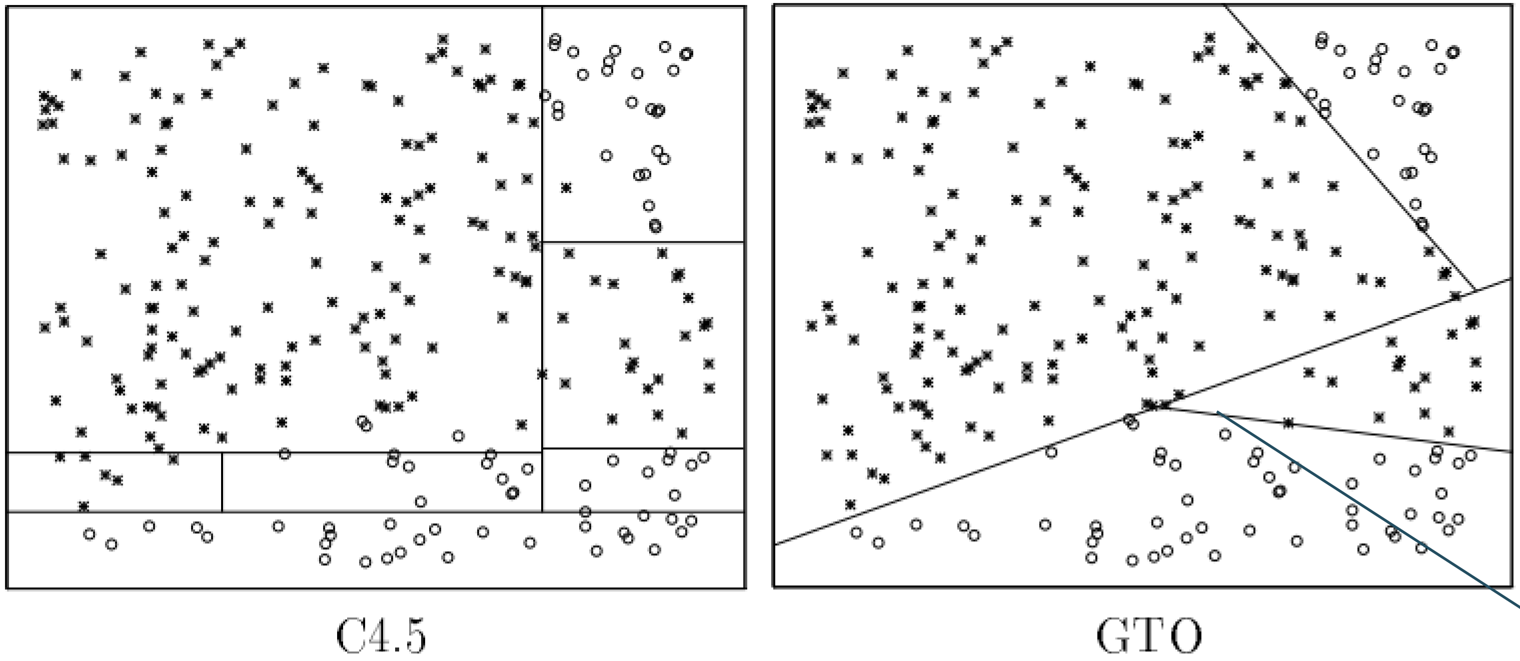
Locally Optimal

Greedy !



Gini or Entropy

CART vs Optimal DT



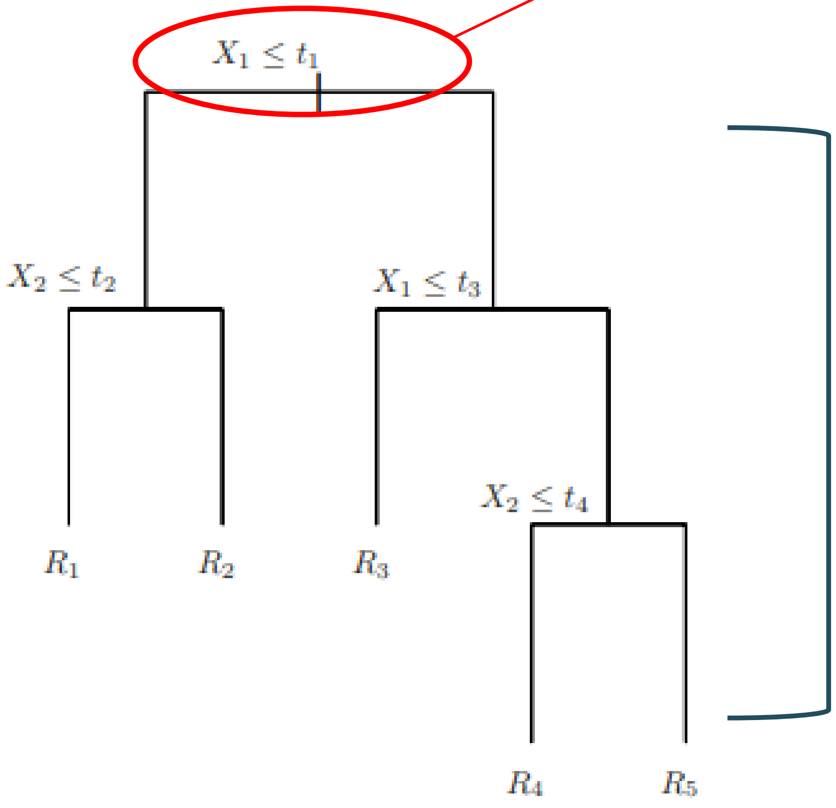
globally Optimal

* Optimal Decision Trees, Kristin P. Bennett et all.

1

Problem

Problem



1. If wrong split, it cannot be undone.

2. Full DT optimizaiton → NP hard!

Notation

$\{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$: The Training dataset

- $\mathbf{x}$: N * M covariate matrix
- $\mathbf{x}_i$: M-vectors of features
- $\mathbf{x}_{ij}$: the j-th feature of $\mathbf{x}_i$
- $\widetilde{\mathbf{x}}$: binarized covariate matrix
- $y_i \in \{0, 1\}$: labels

$M = \sum_{j=1}^{M}(k_j - 1)$: The total # of features

k_j : The # of unique values realized by feature j

Objectives

$$\min_t \underbrace{L(t, \widetilde{\mathbf{x}}, \mathbf{y})}_{\text{Misclassification error}} \quad \text{s.t.} \quad \underbrace{\operatorname{depth}(t)}_{\text{depth bound}} \leq d. \tag{1}$$

$$\min_t \quad L(t, \widetilde{\mathbf{x}}, \mathbf{y}) + \underbrace{\lambda H(t)}_{\text{Sparsity = hyper parameter * \# of Leaves}} \tag{2}$$

tree t prediction

- $L(t, \widetilde{\mathbf{x}}, \mathbf{y}) = \frac{1}{N} \sum_{i=1}^{N} 1[y_i \neq \hat{y}_i^t]$
- In this paper, Combining (1) and (2) to produce a new objective.

Solution

1. **Guessing Thresholds**
   - Tree model need **binary** inputs.
   - We obtain thresholds from Reference model.

2. **Guessing Depth**
   - Guideliness for setting **depth** in the new ojb function.

3. **Guessing Tighter Lower Bounds**
   - Use lower bound to **prune** the search space.
   - Dynamic Programming

2

Guess

Guessing Thresholds

Column Elimination Algorithm

1. Starting with our reference model, extract all thresholds for all features used in all of the trees in the boosting model.
2. Order them by variable importance (we use Gini importance), and remove the least important threshold (among all thresholds and all features).
3. Re-fit the boosted tree with the remaining features.
4. Continue this procedure until the training performance of the remaining tree drops below a predefined threshold.

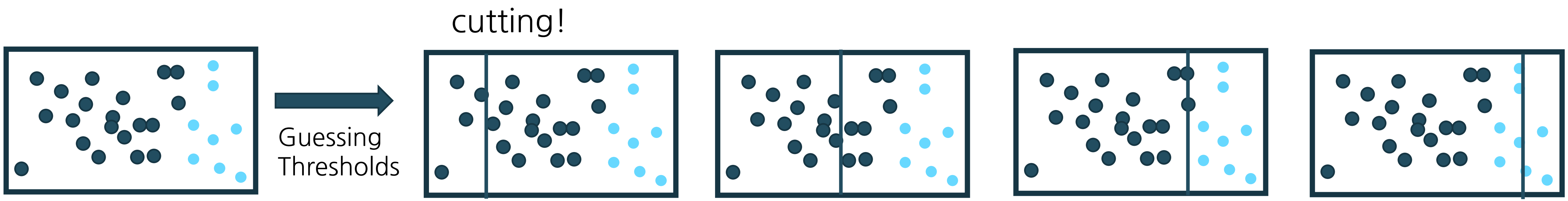
	Traveler Type	Loc_score	Clean_score	Serv_score	Fac_score	VfM_score	Duration
0	1.0	9.4	8.4	8.5	6.9	7.7	1.0
1	1.0	9.4	8.4	8.5	6.9	7.7	2.0
2	1.0	9.4	8.4	8.5	6.9	7.7	1.0
3	0.0	9.4	8.4	8.5	6.9	7.7	1.0
4	1.0	9.4	8.4	8.5	6.9	7.7	1.0
...	...	...	...	...	...	...	...
31771	1.0	9.6	9.8	9.5	9.2	9.1	1.0
31772	1.0	9.6	9.8	9.5	9.2	9.1	5.0
31773	1.0	9.6	9.8	9.5	9.2	9.1	3.0
31774	1.0	9.6	9.8	9.5	9.2	9.1	6.0
31775	1.0	9.6	9.8	9.5	9.2	9.1	2.0

31776 rows × 34 columns

Guessing Thresholds

	VfM_score<=7.450000047683716	Serv_score<=7.25
0	0	0
1	0	0
2	0	0
3	0	0
4	0	0
...	...	...
31771	0	0
31772	0	0
31773	0	0
31774	0	0
31775	0	0

31776 rows × 2 columns



Guessing depth

$$\min_{t} \; L(t, \widetilde{\mathbf{x}}, \mathbf{y}) + \underbrace{\lambda H(t)}_{\text{a per-leaf penalty}} \quad \text{s.t.} \quad \underbrace{\text{depth}(t)}_{\text{depth bound}} \leq d. \tag{3}$$

$$\begin{tikzcd}
\text{Sparse} \arrow[dr] & \text{Fast} \arrow[dl] \\
\text{Slow} & \text{shallower than max depth}
\end{tikzcd}$$

- λ ← 1 / (# of dataset) = 1/n
- What value should depth d set? ⟶ Theorem 4.2

one level DT

Theorem 4.2. (*Min depth needed to match complexity of ensemble*). Let B be the base hypothesis class (e.g., decision stumps or shallow trees) that has VC dimension at least 3 and let $K \geq 3$ be the number of weak classifiers (members of B) combined in an ensemble model. Let F_{ensemble} be the set of weighted sums of weak classifiers, i.e., $T \in F_{\text{ensemble}}$ has $T(x) = \text{sign}\left(\sum_{k=1}^{K} w_k h_k(x)\right)$, where $w_k \in \mathbb{R}$ and $h_k \in B$. Let F_{dtree} be the class of single binary decision trees with depth at most:

$$d = \log_2((K \cdot VC(B) + K) \cdot (3\ln(K \cdot VC(B) + K) + 2))$$

It is then true that $VC(F_{\text{dtree}}) \geq VC(F_{\text{ensemble}})$.

Complexity

- VC dimension?
- If B : class of single tree depth <= 3, then To use depth 11 tree from B according to red line equation.

Guessing Tighter Lower Bounds

Step 1.

If $ub(s_a) \leq lb_{\text{guess}}(s_a) + \lambda$ or $d = 0$:

we are done with the subproblem.

else:

set the current lower bound $lb_{\text{curr}}(s_a, d) = lb_{\text{guess}}(s_a)$ and go to Step 2.

Step 2.

Search the space of possible trees for subproblem (s_a, d)

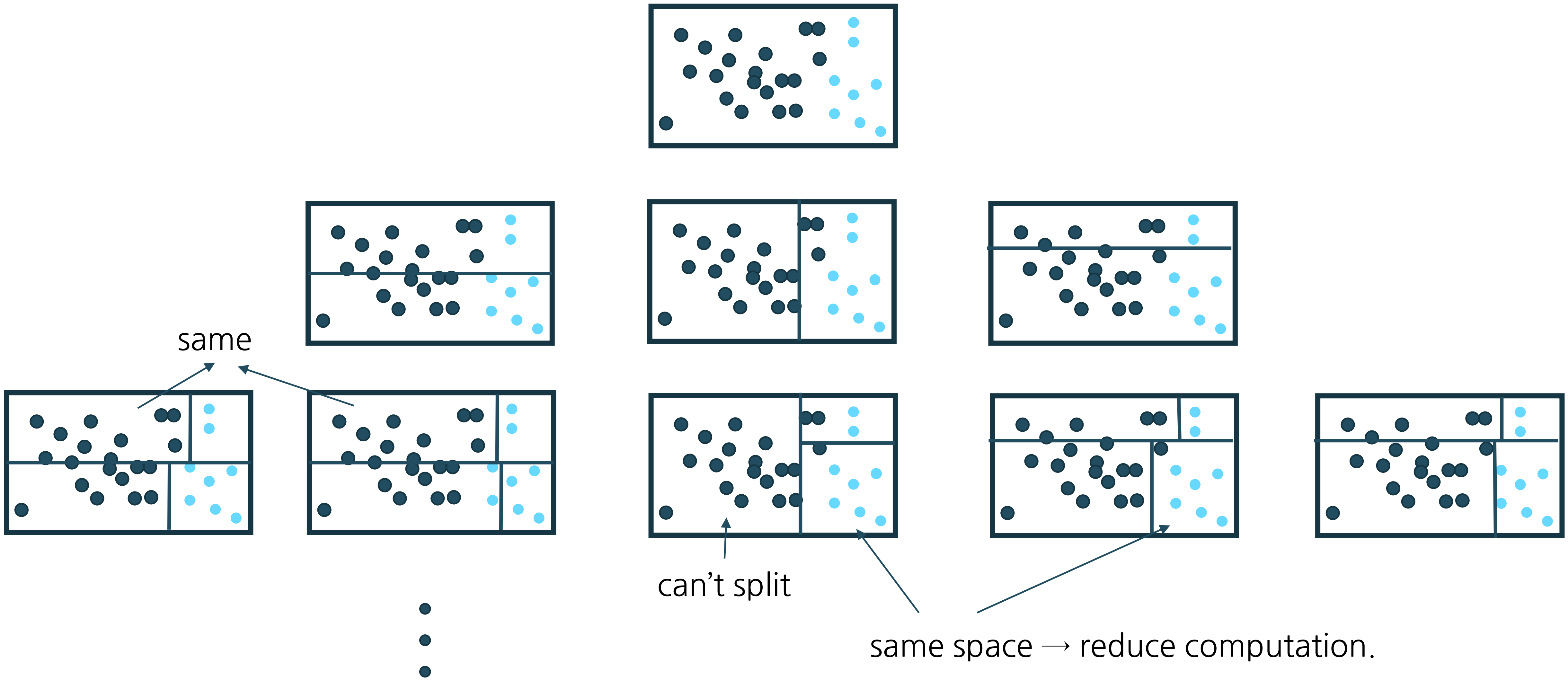
$\rightarrow$ two new subproblems with depth $d-1$ and solving recursively.

(a) If $\exists$ a subtree t for (s_a, d) such that $R(t, \tilde{x}(s_a), y(s_a)) \leq lb_{\text{current}}(s_a, d)$:

we are done with the subproblem.

(b) $lb_{\text{curr}}(s_a, d) \leftarrow \max(lb_{\text{curr}}(s_a, d), \min(ub(s_a), lb_{\text{spl}}))$

$$\min_{j \in \text{features}} (lb_{\text{curr}}(s_a \cap s_j, d') + lb_{\text{curr}}(s_a \cap s_j^c, d'))$$

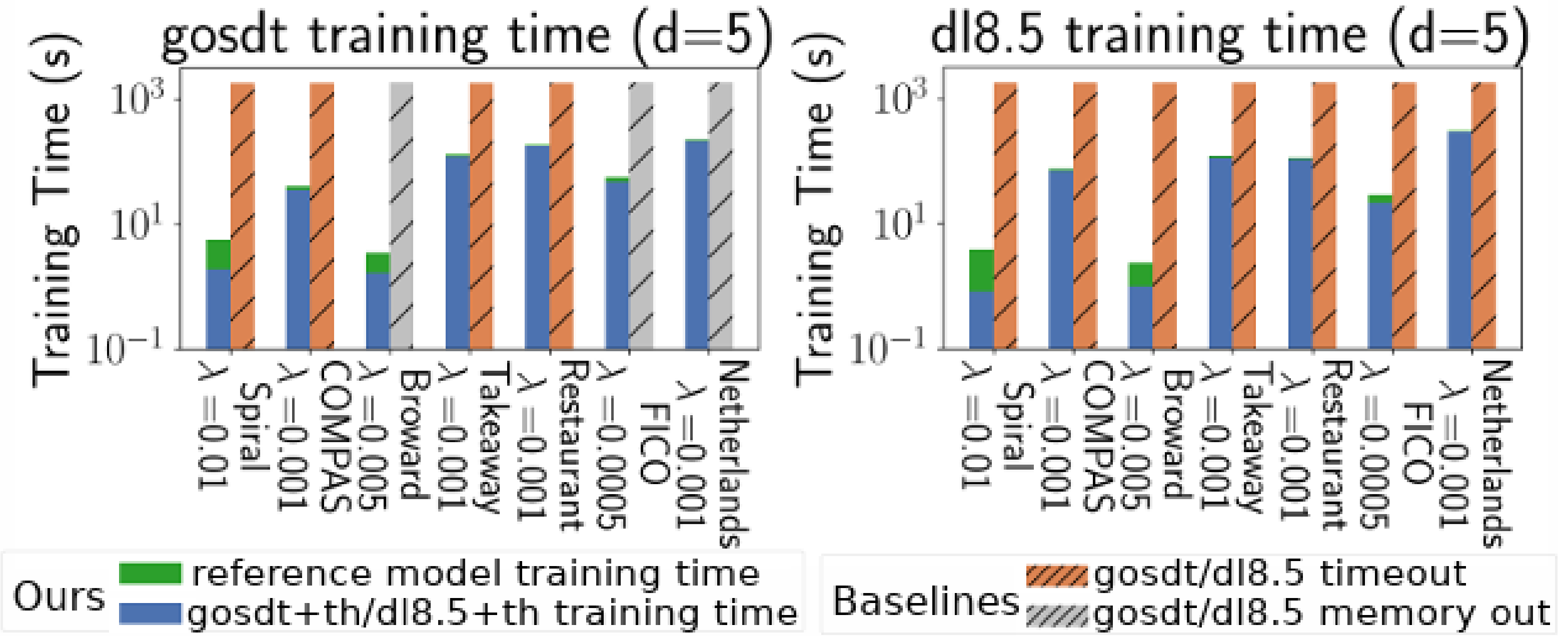

same
can't split
same space → reduce computation.

3

Experiment

Guessing - 1

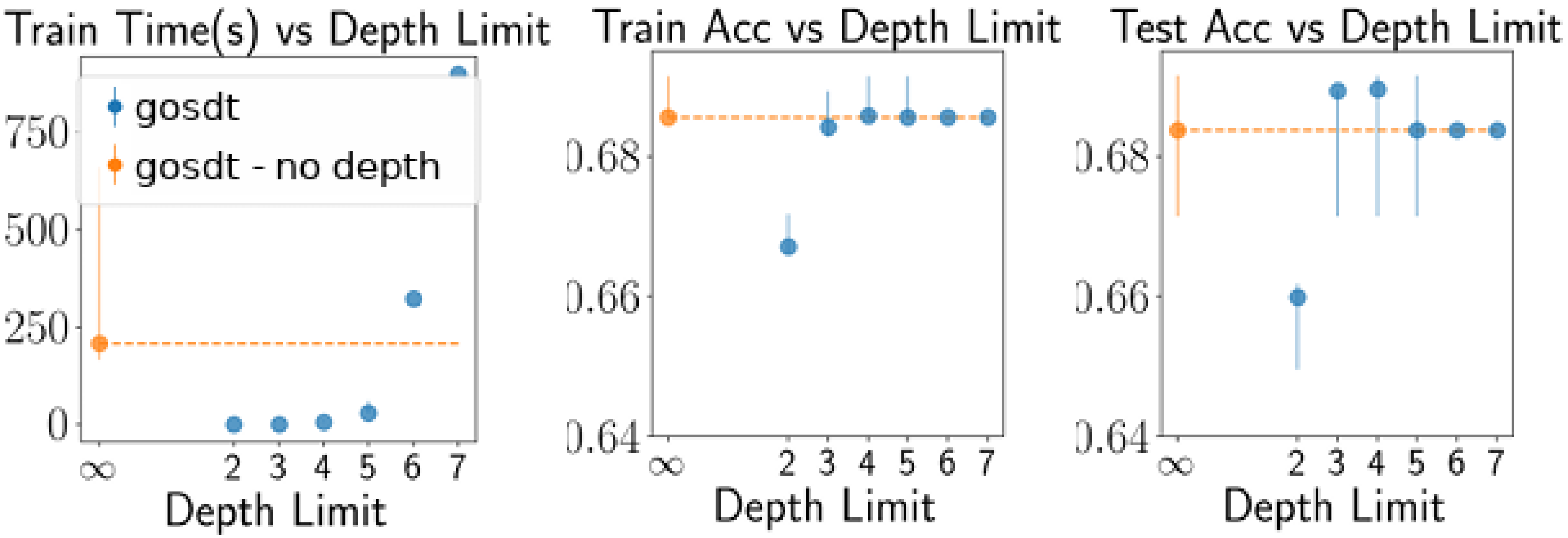
30 minute
125 GB memory



- threshold guessing time
- time out
- over memory

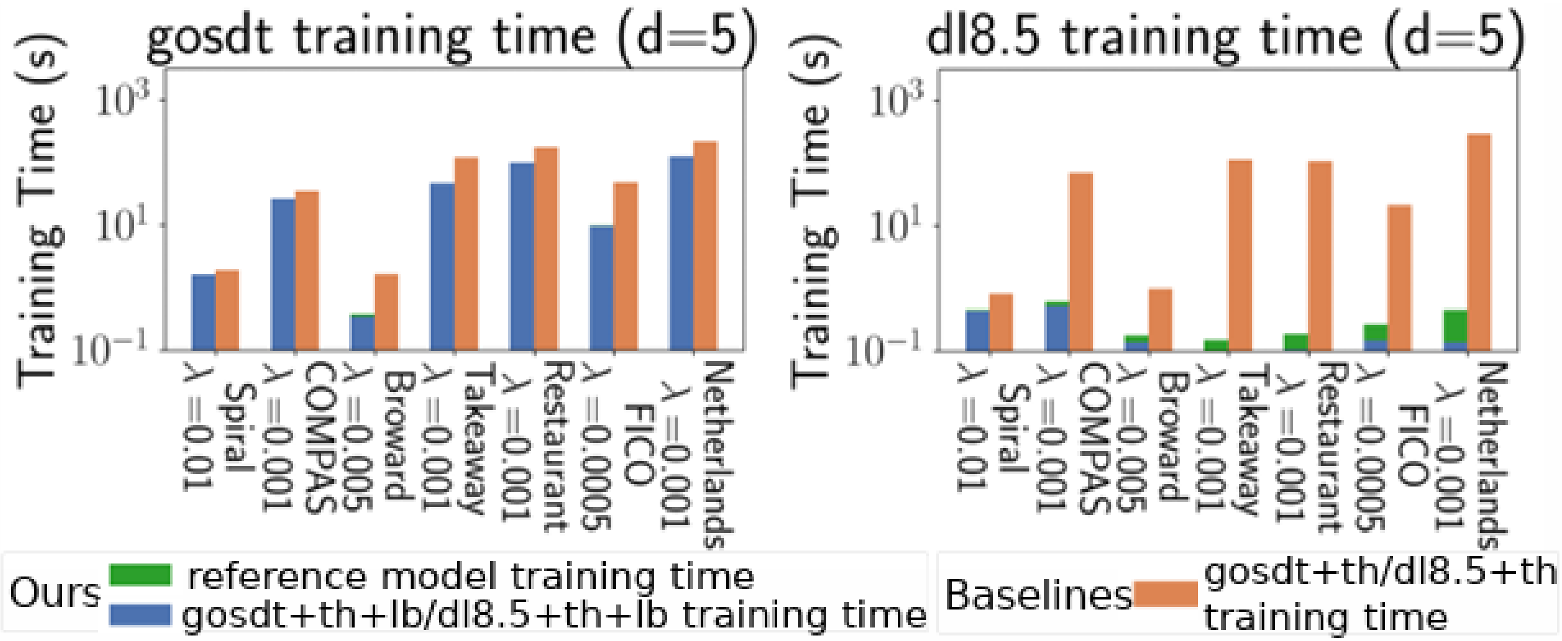
Guessing method faster than baseline.

Guessing - 2



- constraint depth → ACC : A little performance improvement

Guessing - 3



- similar result

Model Output

```
X: (442, 7)
y: (442,)
gosdt reported successful execution
training completed. 0.000/0.000/0.000 (user, system, wall), mem=0 MB
bounds: [0.020362..0.020362] (0.000000) loss=0.011312, iterations=17
evaluate the model, extracting tree and scores
Model training time: 0.0
Training accuracy: 0.9886877828054299
# of leaves: 4
if Score<=5.849999904632568 = 1 then:
    predicted class: 1
    misclassification penalty: 0.0
    complexity penalty: 0.002

else if Loc_score<=9.349999904632568 = 1 and Score<=5.849999904632568 != 1 and Score<=6.549999952316284 = 1 then:
    predicted class: 1
    misclassification penalty: 0.011
    complexity penalty: 0.002

else if Loc_score<=9.349999904632568 != 1 and Score<=5.849999904632568 != 1 and Score<=6.549999952316284 = 1 then:
    predicted class: 0
    misclassification penalty: 0.0
    complexity penalty: 0.002

else if Score<=5.849999904632568 != 1 and Score<=6.549999952316284 != 1 then:
    predicted class: 0
    misclassification penalty: 0.0
    complexity penalty: 0.002
```

4

Appendix

Appendix 1 - VC dimension

$$VC(h) = max(d|\text{there exists a set of d points that can be shattered by h})$$

VC dimension = 5

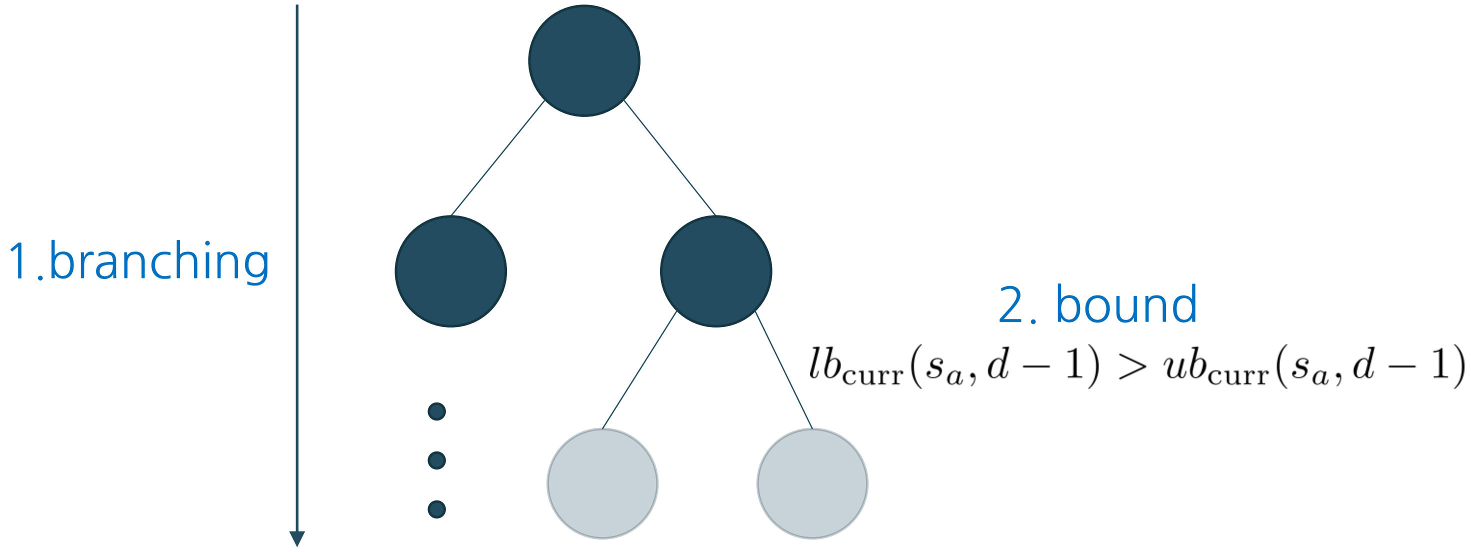
- To measure complexity of Machine Learning Model
- the largest number of points that can be shattered by a binary classifier without misclassification.

Appendix 2 - Branch and Bound

(s_a, d) : subproblem

$lb_{\text{curr}}(s_a, d)$: current lower bound

$ub_{\text{curr}}(s_a, d)$: current upper bound



Recursively : problem → subprob → subsub → . . .