Delft University of Technology

Document Version
Final published version

Citation (APA)
Arslan, E., van der Linden, J. G. M., Hoogendoorn, S. P., Rinaldi, M., & Demirović, E. (2025). *SORTeD Rashomon Sets of Sparse Decision Trees: Anytime Enumeration*. Poster session presented at 39th Conference on Neural Information Processing Systems, NeurIPS 2025, San Diego, CA, USA / Mexico City, MX.
https://neurips.cc/media/PosterPDFs/NeurIPS%202025/118948.png?t=1762464122.0007844

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Copyright
In case the licence states “Dutch Copyright Act (Article 25fa)”, this publication was made available Green Open Access via the TU Delft Institutional Repository pursuant to Dutch Copyright Act (Article 25fa, the Taverne amendment). This provision does not affect copyright ownership.
Unless copyright is transferred by contract or statute, it remains with the copyright holder.

Sharing and reuse
Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy
Please contact us and provide details if you believe this document breaches copyrights.
We will remove access to the work immediately and investigate your claim.

This work is downloaded from Delft University of Technology.

SPOTLIGHT POSTER

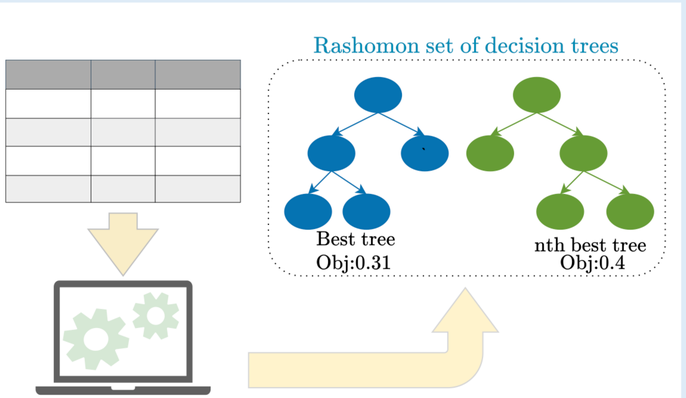
NeurIPS | 2025

SORTeD Rashomon Sets of Sparse Decision Trees: Anytime Enumeration

TU Delft | AI Initiative — XAIT AITL TLAB

Elif Arslan *van der Linden, Hoogendoorn, Rinaldi, Demirović*

1. Rashomon set of decision trees



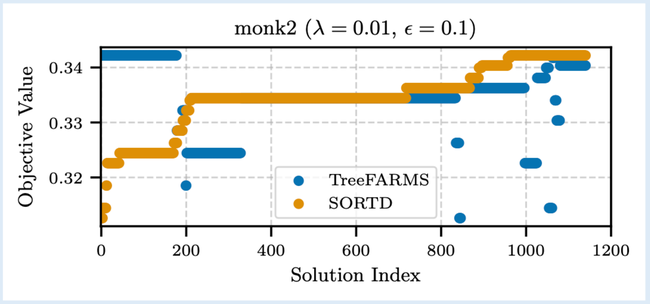
The set of all models within a tolerance of the globally optimal loss value:

$$\mathrm{R}(\mathrm{T}^*, \mathrm{D}, \varepsilon) = \{\mathrm{T} \in \mathcal{T}(\mathrm{d}) \mid \mathrm{C}(\mathrm{T}, \mathrm{D}) \leq (1 + \varepsilon)(\mathrm{C}(\mathrm{T}^*, \mathrm{D}))\}$$

Use Rashomon sets instead of single models to:

- ✓ Discover simpler models,
- ✓ Improve variable importance analysis,
- ✓ Enable stakeholders to select models aligning well with their constraints.

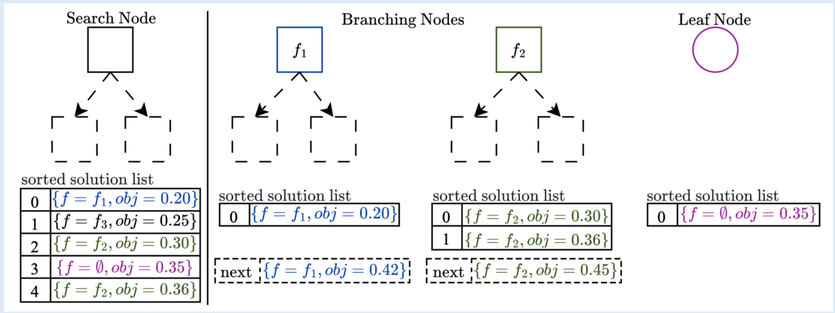
2. Why a sorted Rashomon set?



Find high-quality trees quickly.

3.a Methodology - Lazily generate the next best solution

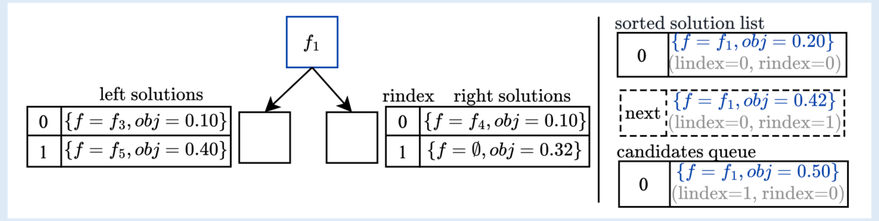
SORTD iteratively builds the Rashomon set by maintaining sorted lists of solutions for each node in the search tree.



Search tree structure with current node (left), branching nodes with features f_1 and f_2 (middle), and leaf node (right).

A search node:

- Returns its next best solution from its list **or, if none remain,**
- Selects the best candidate from its helper (branching or leaf) nodes, triggering that helper to generate its next solution.



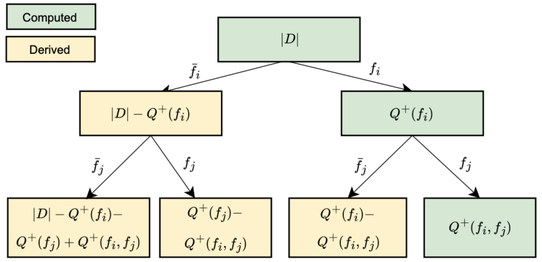
Next solution calculation in a branching node.

A branching node:

- Computes its next solution by combining the best solutions from its left and right child nodes, resulting in **an iteratively calculated sorted Cartesian sum** ($\{a + b \mid a \in A, b \in B\}$).

3.b Methodology - Depth-two subroutine

To improve runtime performance, SORTD employs the Depth-two subroutine that uses precomputed frequency counts to effectively compute trees of depth two.

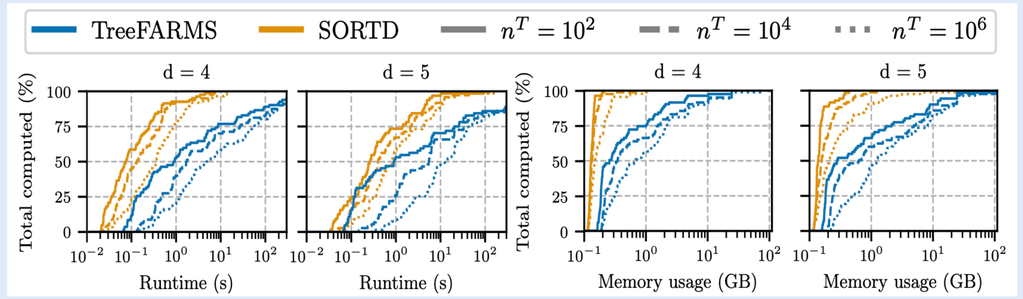


SORTD gradually creates the depth-two solutions to avoid creating all depth-two solutions unnecessarily when there is a size limit on Rashomon set.

Paper

Python Package

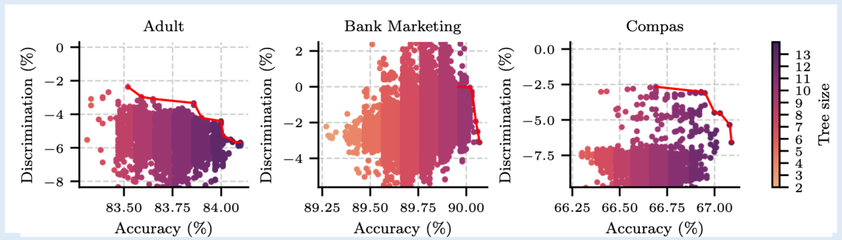
4. Results - Up to two orders of magnitude runtime improvement



Runtime and memory performance of methods to obtain 10^n trees.

Based on the evaluations using 30 datasets, SORTD achieves

- **up to two-orders of magnitude** runtime improvement,
- **an order of magnitude** decrease in memory usage.



The top 10^7 trees in the accuracy Rashomon set evaluated using *equality-of-opportunity* with max-depth four and $\lambda = 0.001$.

Additionally, SORTD

- **optimizes for** classification, regression and more,
- **evaluates** these **secondarily** along with others such as fairness.

TreeFARMS: Xin, R., Zhong, C., Chen, Z., Takagi, T., Seltzer, M., & Rudin, C. (2022). Exploring the whole Rashomon set of sparse decision trees. Advances in neural information processing systems, 35, 14071-14084.

Variable importance: Donnelly, J., Katta, S., Rudin, C., & Browne, E. (2023). The rashomon importance distribution: Getting rid of unstable, single model-based variable importance. Advances in Neural Information Processing Systems, 36, 6267-6279.

Depth-two subroutine: Demirović, E., Lukina, A., Hebrard, E., Chan, J., Bailey, J., Leckie, C., ... & Stuckey, P. J. (2022). Murtree: Optimal decision trees via dynamic programming and search. Journal of Machine Learning Research, 23(26), 1-47.