Necessary and Sufficient Conditions for Optimal Decision Trees using Dynamic Programming

Jacobus G. M. van der Linden[1] [0009−0001−4015−0594], Mathijs M. de Weerdt[1] [0000−0002−0470−6241], and Emir Demirović[1] [0000−0003−1587−5582]

Delft University of Technology, Delft 2628 CD, The Netherlands

Abstract. Global optimization of decision trees has shown to be promising in terms of accuracy, size, and consequently human comprehensibility. However, many of the methods used rely on general-purpose solvers for which scalability remains an issue. Dynamic programming methods have been shown to scale much better because they exploit the tree structure by solving subtrees separately as independent subproblems. We explore this relationship in detail and show the necessary and sufficient conditions for such separability and generalize previous dynamic programming approaches into a framework that can optimize any combination of separable objectives and constraints. Experiments on five application domains show the general applicability of this framework, while outperforming the scalability of general-purpose solvers by a large margin.
This submission is based on our NeurIPS 2023 paper [9] and extended by our later work at AAI-24 [6] and ICML-24 [3].

Keywords: Optimal Decision Trees · Dynamic Programming · Interpretability.

1 Introduction

Many high-stake domains, such as medical diagnosis, housing appointments, and hiring procedures, require human-comprehensible machine learning (ML) models to ensure transparency, safety, and reliability [12]. Decision trees offer such human comprehensibility, provided the trees are small [11]. This motivates the search for *optimal* decision trees, i.e., trees that globally optimize an objective for a given maximum size. For broad-scale application of optimal decision trees, we need methods that can *generalize* to incorporate objectives and constraints from a variety of domains, while remaining *scalable* to real-world problem sizes.

However, many state-of-the-art methods for optimal decision trees lack the required scalability, for example, to optimize datasets with more than several thousands of instances or to find optimal trees beyond depth three. This includes approaches such as mixed-integer programming (MIP) [2,14], constraint programming (CP) [13], boolean satisfiability (SAT) [7] or maximum satisfiability (MaxSAT) [5].

Dynamic programming (DP) approaches show scalability that is orders of magnitude better [1,4,8] by directly exploiting the tree structure. Each problem is solved by a recursive step that involves two subproblems, each just half the size of the original problem. However, unlike general-purpose solvers, such as MIP, DP cannot trivially adapt to the variety of objectives and constraints mentioned before. For DP to work efficiently, it must be possible to solve the optimization task at hand *separately* for every subtree.

Therefore, the main research question considered here is to what extent can this separability property be generalized? In answering this question we provide a generic framework called *STreeD* (Separable Trees with Dynamic programming).[1] We push the limits of DP for optimal decision trees by providing conditions for separability that are both necessary and sufficient. These conditions are less strict and extend to a larger class of optimization tasks than those of state-of-the-art DP frameworks [8,10]. Examples of problems that could not be solved to global optimality by previous frameworks but can be solved with our framework, are group fairness constraints, nonlinear metrics, and revenue maximization under a capacity constraint. Moreover, STreeD implements several algorithmic techniques for increasing scalability, such as a specialized solver for trees up to depth two. We thus attain generalizability similar to general-purpose solvers, while preserving the scalability of DP methods.

In our experiments, we demonstrate the flexibility of STreeD on a variety of optimization tasks, including cost-sensitive classification, prescriptive policy generation, nonlinear classification metrics, and group fairness.

In later work, we used the same framework to optimize regression trees with constant, linear regression, and simple linear regression models in the leaf node [3], and also survival trees that predict the probability of surviving beyond a certain time, while being trained on partly censored data [6].

In summary, our main contributions are 1) a generalized DP framework (STreeD) for optimal decision trees that can optimize any *separable* objective or constraint; 2) a proof for necessary and sufficient conditions for separability; and 3) extensive experiments on five application domains that show the flexibility of STreeD, while performing on par or better than the state of the art. Two more application domains are added in follow-up work.

References

1. Aglin, G., Nijssen, S., Schaus, P.: Learning Optimal Decision Trees Using Caching Branch-and-Bound Search. In: Proceedings of AAAI-20. pp. 3146–3153 (2020)
2. Bertsimas, D., Dunn, J.: Optimal classification trees. Machine Learning **106**(7), 1039–1082 (2017)
3. van den Bos, M., van der Linden, J.G.M., Demirović, E.: Piecewise constant and linear regression trees: An optimal dynamic programming approach. In: Proceedings of ICML-24 (2024)
4. Demirović, E., Lukina, A., Hebrard, E., Chan, J., Bailey, J., Leckie, C., Ramamohanarao, K., Stuckey, P.J.: MurTree: Optimal Classification Trees via Dynamic Programming and Search. Journal of Machine Learning Research **23**(26), 1–47 (2022)

[1] Source repository: https://github.com/algtudelft/pystreed

5. Hu, H., Siala, M., Hebrard, E., Huguet, M.J.: Learning Optimal Decision Trees with MaxSAT and its Integration in AdaBoost. In: IJCAI-PRICAI 2020. pp. 1170–1176 (2020)
6. Huisman, T., van der Linden, J.G.M., Demirović, E.: Optimal survival trees: A dynamic programming approach. In: Proceedings of the AAAI-24. pp. 12680–12688 (2024)
7. Janota, M., Morgado, A.: SAT-Based Encodings for Optimal Decision Trees with Explicit Paths. In: Proceedings of the International Conference on Theory and Applications of Satisfiability Testing (SAT 2020). pp. 501–518 (2020)
8. Lin, J., Zhong, C., Hu, D., Rudin, C., Seltzer, M.: Generalized and Scalable Optimal Sparse Decision Trees. In: Proceedings of ICML-20. pp. 6150–6160 (2020)
9. van der Linden, J.G.M., de Weerdt, M.M., Demirović, E.: Necessary and Sufficient Conditions for Optimal Decision Trees using Dynamic Programming. In: Advances in NeurIPS-23. pp. 9173–9212 (2023)
10. Nijssen, S., Fromont, E.: Optimal constraint-based decision tree induction from itemset lattices. Data Mining and Knowledge Discovery **21**(1), 9–51 (2010)
11. Piltaver, R., Luštrek, M., Gams, M., Martinčić-Ipšić, S.: What makes classification trees comprehensible? Expert Systems with Applications **62**, 333–346 (2016)
12. Rudin, C.: Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. Nature Machine Intelligence **1**(5), 206–215 (2019)
13. Verhaeghe, H., Nijssen, S., Pesant, G., Quimper, C.G., Schaus, P.: Learning Optimal Decision Trees using Constraint Programming. Constraints **25**(3), 226–250 (2020)
14. Verwer, S., Zhang, Y.: Learning Optimal Classification Trees Using a Binary Linear Program Formulation. In: Proceedings of AAAI-19. pp. 1625–1632 (2019)