



# Decision Trees with at Most 19 Vertices for Knowledge Representation

Mohammad Azad<sup>(✉)</sup>

College of Computer and Information Science, Jouf University,  
Sakaka 72441, Saudi Arabia  
mmazad@ju.edu.sa

**Abstract.** We study decision trees as a means of representation of knowledge. To this end, we design two techniques for the creation of CART (Classification and Regression Tree)-like decision trees that are based on bi-objective optimization algorithms. We investigate three parameters of the decision trees constructed by these techniques: number of vertices, global misclassification rate, and local misclassification rate.

**Keywords:** Knowledge representation · Decision trees · Dynamic programming · Bi-objective optimization

## 1 Introduction

Decision trees are used to a large degree as classifiers [5, 6, 10], as a means of representation of knowledge [4, 7], and as a kind of algorithms [20, 25]. We investigate here decision trees as a means of representation of knowledge.

Let us consider a decision tree  $\Gamma$  for a decision table  $D$ . We investigate three parameters of  $\Gamma$ :

- $N(\Gamma)$  – the number of vertices in  $\Gamma$ .
- $G(D, \Gamma)$  – the global misclassification rate [7], which is equal to the number of misclassifications of  $\Gamma$  divided by the number of rows in  $D$ .
- $L(D, \Gamma)$  – the local misclassification rate [7], which is the maximum fraction of misclassifications among all leaves of  $\Gamma$ . One can show that  $G(D, \Gamma)$  is at most  $L(D, \Gamma)$ .

The decision tree  $\Gamma$  should have a reasonable number of vertices to be understandable. To express properly knowledge from the decision table  $D$ , this tree should have an acceptable accuracy. In [7], we mentioned that the consideration of only the global misclassification rate may be insufficient: the misclassifications may be unevenly distributed and, for some leaves, the fraction of misclassifications can be high. To deal with this situation, we should consider also the local misclassification rate.

The optimization of the parameters of decision tree has been studied by many researchers [9, 11–13, 16–19, 24, 26]. One of the directions of the research is the bi-objective optimization [1–8]. In [7], we proposed three techniques for the building

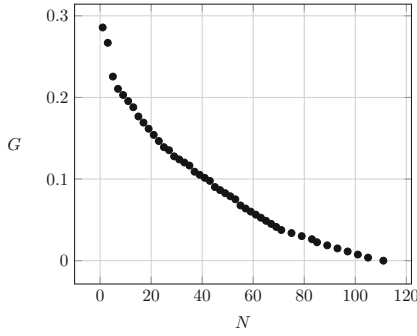
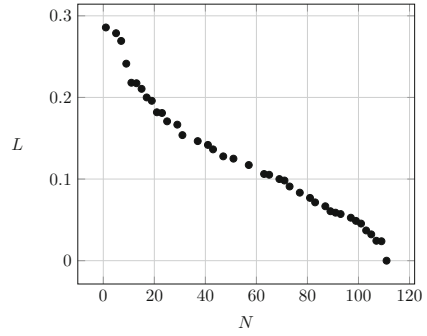
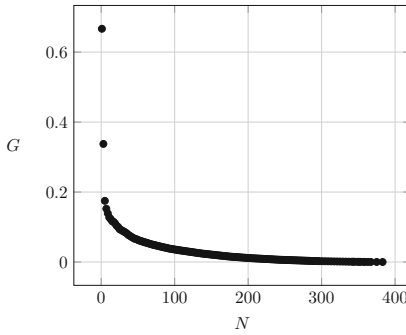
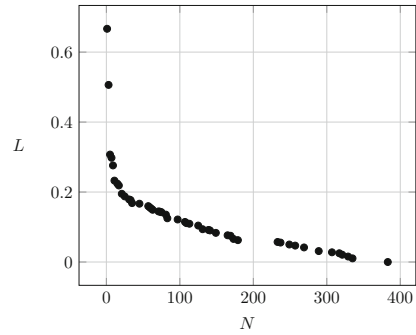
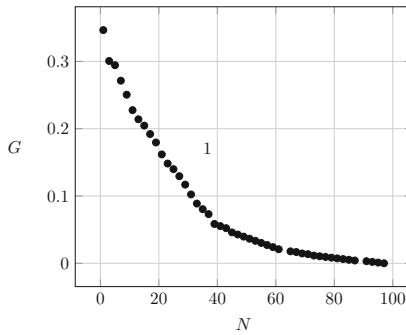
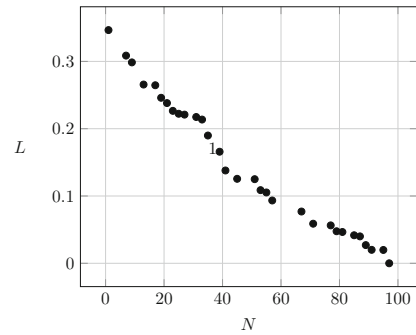
of decision trees based on the bi-objective optimization of trees and studied the parameters  $N$ ,  $G$ , and  $L$  of the constructed decision trees. Unfortunately, these techniques are applicable to medium-sized decision tables with categorical features only and, sometimes, the number of vertices in the trees is too high. In particular, the decision tree  $T_1$  with the minimum number of vertices constructed by these techniques for the decision table D NURSERY from the UCI Machine Learning Repository [15] has the following parameters:  $N(T_1) = 70$ ,  $G(D, T_1) = 0.10$ , and  $L(D, T_1) = 0.23$ .

In this paper, instead of conventional decision trees, we study CART-like (CART-L) decision trees introduced in the books [1, 2]. As the standard CART [10] trees, CART-L trees use binary splits instead of the initial features. The standard CART tree uses in each internal vertex the best split among all features. A CART-L tree can use in each internal vertex the best split for an arbitrary feature. It extends essentially the set of decision trees under consideration. In [1, 2], we applied Gini index to define the notion of the best split. In this paper, we use another parameter  $abs$  [2].

We design two techniques that build decision trees for medium-sized tables (at most 10,000 rows and at most 20 features) containing both categorical and numerical features. These techniques are based on bi-objective optimization of CART-L decision trees for parameters  $N$  and  $G$  [1], and for parameters  $N$  and  $L$ . Both techniques construct decision trees with at most 19 vertices (at most 10 leaves and at most nine internal vertices). The choice of 19 is not random. We consider enough understandable trees with small number of non-terminal vertices which can be useful from the point of view of knowledge representation. This choice is supported by some experimental results published in [1]. One technique ( $G$ -19 technique) was proposed in [1]. Another one ( $L$ -19 technique) is completely new. We apply the considered techniques to 14 data sets from the UCI Machine Learning Repository [15], and study three parameters  $N$ ,  $G$ , and  $L$  of the constructed trees. For example, for the decision table  $D$  NURSERY,  $L$ -19 technique constructs a decision tree  $T_2$  with  $N(T_2) = 17$ ,  $G(D, T_2) = 0.12$ , and  $L(D, T_2) = 0.22$ .

The obtained results show that at least one of the considered techniques ( $L$ -19 technique) can be useful for the extraction of knowledge from medium-sized decision tables and for its representation by decision trees. This technique can be used in different areas of data analysis including rough set theory [14, 21–23, 27]. In rough set, the decision rules are used extensively. We can easily derive decision rules from the constructed decision trees and use them in rough set applications.

We arrange the remaining of the manuscript as follows. Two techniques for decision tree building are explained in Sect. 2. The output of the experiments is in Sect. 3. Finally, Sect. 4 contains brief conclusion.

(a) BREAST-CANCER,  $N$  and  $G$ (b) BREAST-CANCER,  $N$  and  $L$ (c) NURSERY,  $N$  and  $G$ (d) NURSERY,  $N$  and  $L$ (e) TIC-TAC-TOE,  $N$  and  $G$ (f) TIC-TAC-TOE,  $N$  and  $L$ 

**Fig. 1.** Sets of Pareto optimal points for tables BREAST-CANCER, NURSERY, and TIC-TAC-TOE for pairs of parameters  $N$ ,  $G$  and  $N$ ,  $L$

## 2 Two Techniques for Decision Tree Construction

In the books [1,2], an algorithm  $\mathcal{A}_{\text{POPs}}$  is described. If we give this algorithm a decision table, then it builds the Pareto front – the set of all POPs (Pareto optimal points) for bi-objective optimization of CART-L trees relative to  $N$  and  $G$  (see, for example, Fig. 1(a), (c), (e)). We extend this algorithm to the building of the Pareto front for parameters  $N$  and  $L$  (see, for example, Fig. 1(b), (d), (f)). For each POP, we can get a decision tree with values of the considered parameters equal to the coordinates of this point. Both algorithm  $\mathcal{A}_{\text{POPs}}$  and its extension have exponential time complexity in the worst case. We now describe two techniques of decision tree building based on the operation of the algorithm  $\mathcal{A}_{\text{POPs}}$  and its extension. The time complexity of these two techniques is exponential in the worst case.

### 2.1 G-19 Technique

We apply the algorithm  $\mathcal{A}_{\text{POPs}}$  to a decision table  $D$ . The output of this algorithm is the Pareto front for the bi-objective optimization of CART-L trees for parameters  $N$  and  $G$ . We choose a POP with the maximum value of the parameter  $N$  which is at most 19. After that, we get a decision tree  $\Gamma$ , for which the parameters  $N$  and  $G$  are equal to the coordinates of this POP. The tree  $\Gamma$  is the output of  $G$ -19 technique. This technique was described in [1]. However, we did not study the parameter  $L$  for the constructed trees.

### 2.2 L-19 Technique

We apply the extension of the algorithm  $\mathcal{A}_{\text{POPs}}$  to a decision table  $D$  to create the Pareto front for the bi-objective optimization of CART-L trees for parameters  $N$  and  $L$ . We choose a POP with the maximum value of the parameter  $N$  which is at most 19. After that, we get a decision tree  $\Gamma$ , for which the parameters  $N$  and  $L$  are equal to the coordinates of this POP. The tree  $\Gamma$  is the output of  $L$ -19 technique. This is a new technique.

## 3 Results of Experiments

In Table 1, we describe 14 decision tables, each with its name, number of features as well as number of objects (rows). These tables are collected from the UCI Machine Learning Repository [15] for performing the experiments.

We applied  $G$ -19 and  $L$ -19 techniques to each of these tables and found values of the parameters  $N$ ,  $G$ , and  $L$  for the constructed decision trees. Table 2 describes the experimental results.

The obtained results show that the use of  $L$ -19 technique in comparison with  $G$ -19 technique allows us to decrease the parameter  $L$  on average from 0.16 to 0.11 at the cost of a slight increase in the parameter  $G$  on average from 0.06 to 0.07.

**Table 1.** Decision tables which are collected for performing the experiments

Decision table	#Features	#Objects (rows)
BALANCE-SCALE	5	625
BREAST-CANCER	10	266
CARS	7	1728
HAYES-ROTH-DATA	5	69
HOUSE-VOTES-84	17	279
IRIS	5	150
LENSES	5	10
LYMPHOGRAPHY	19	148
NURSERY	9	12960
SHUTTLE-LANDING	7	15
SOYBEAN-SMALL	36	47
SPECT-TEST	23	169
TIC-TAC-TOE	10	958
ZOO-DATA	17	59

**Table 2.** Results of experiments

Decision table	<i>G</i> -19 technique			<i>L</i> -19 technique		
	<i>N</i>	<i>G</i>	<i>L</i>	<i>N</i>	<i>G</i>	<i>L</i>
BALANCE-SCALE	19	0.19	0.38	11	0.20	0.32
BREAST-CANCER	19	0.16	0.26	19	0.18	0.20
CARS	19	0.10	0.39	19	0.12	0.23
HAYES-ROTH-DATA	17	0.06	0.17	17	0.06	0.17
HOUSE-VOTES-84	19	0.02	0.02	19	0.02	0.02
IRIS	17	0.00	0.00	17	0.00	0.00
LENSES	7	0.00	0.00	7	0.00	0.00
LYMPHOGRAPHY	19	0.06	0.20	19	0.10	0.11
NURSERY	19	0.11	0.34	17	0.12	0.22
SHUTTLE-LANDING	13	0.00	0.00	13	0.00	0.00
SOYBEAN-SMALL	7	0.00	0.00	7	0.00	0.00
SPECT-TEST	17	0.02	0.09	19	0.02	0.02
TIC-TAC-TOE	19	0.18	0.39	19	0.21	0.25
ZOO-DATA	17	0.00	0.00	17	0.00	0.00
Average	16.29	0.06	0.16	15.71	0.07	0.11

## 4 Conclusions

We proposed to evaluate the accuracy of decision trees not only by the global misclassification rate  $G$  but also by the local misclassification rate  $L$ , and designed new  $L$ -19 technique. This technique constructs decision trees having at most 19 vertices and acceptable values of the parameters  $G$  and  $L$ . Later we are planning to extend this technique to multi-label decision tables using bi-objective optimization algorithms described in [2, 3]. Also, our goal is to make more experiments with other numbers of vertices like 13, 15, 17, 21, 23, etc. Another direction of future research is to design some heuristics to overcome the problem of working with larger data set.

**Acknowledgments.** The author expresses his gratitude to Jouf University for supporting this research. The author would like to give thanks to Igor Chikalov and Mikhail Moshkov for valuable comments. The author also would like to give thanks to anonymous reviewers for their suggestions.

## References

1. AbouEisha, H., Amin, T., Chikalov, I., Hussain, S., Moshkov, M.: Extensions of Dynamic Programming for Combinatorial Optimization and Data Mining. Intelligent Systems Reference Library, vol. 146. Springer, Heidelberg (2019). <https://doi.org/10.1007/978-3-319-91839-6>
2. Alsolami, F., Azad, M., Chikalov, I., Moshkov, M.: Decision and Inhibitory Trees and Rules for Decision Tables with Many-Valued Decisions. Intelligent Systems Reference Library, vol. 156. Springer, Heidelberg (2020). <https://doi.org/10.1007/978-3-030-12854-8>
3. Azad, M.: Decision and inhibitory trees for decision tables with many-valued decisions. Ph.D. thesis, King Abdullah University of Science & Technology, Thuwal, Saudi Arabia (2018). <http://hdl.handle.net/10754/628023>
4. Azad, M.: Knowledge representation using decision trees constructed based on binary splits. KSII Trans. Internet Inf. Syst. **14**(10), 4007–4024 (2020). <https://doi.org/10.3837/tiis.2020.10.005>
5. Azad, M., Chikalov, I., Hussain, S., Moshkov, M.: Restricted multi-pruning of decision trees. In: 13th International FLINS Conference on Data Science and Knowledge Engineering for Sensing Decision Support, FLINS, pp. 371–378 (2018). [https://doi.org/10.1142/9789813273238\\_0049](https://doi.org/10.1142/9789813273238_0049)
6. Azad, M., Chikalov, I., Hussain, S., Moshkov, M.: Multi-pruning of decision trees for knowledge representation and classification. In: 3rd IAPR Asian Conference on Pattern Recognition, ACPR 2015, Kuala Lumpur, Malaysia, 3–6 November 2015, pp. 604–608. IEEE (2015). <https://doi.org/10.1109/ACPR.2015.7486574>
7. Azad, M., Chikalov, I., Moshkov, M.: Decision trees for knowledge representation. In: Ropiak, K., Polkowski, L., Artiemjew, P. (eds.) 28th International Workshop on Concurrency, Specification and Programming, CS&P 2019, Olsztyn, Poland, 24–26 September 2019. CEUR Workshop Proceedings, vol. 2571. CEUR-WS.org (2019). [http://ceur-ws.org/Vol-2571/CSP2019\\_paper\\_1.pdf](http://ceur-ws.org/Vol-2571/CSP2019_paper_1.pdf)
8. Azad, M., Chikalov, I., Moshkov, M.: Representation of knowledge by decision trees for decision tables with multiple decisions. Proc. Comput. Sci. **176**, 653–659 (2020). <https://doi.org/10.1016/j.procs.2020.09.037>

9. Boryczka, U., Kozak, J.: New algorithms for generation decision trees - ant-miner and its modifications. In: Abraham, A., Hassanien, A.E., de Leon Ferreira de Carvalho, A.C.P., Snásel, V. (eds.) *Foundations of Computational Intelligence - Volume 6: Data Mining*, vol. 206, pp. 229–262. Springer, Heidelberg (2009). [https://doi.org/10.1007/978-3-642-01091-0\\_11](https://doi.org/10.1007/978-3-642-01091-0_11)
10. Breiman, L., Friedman, J.H., Olshen, R.A., Stone, C.J.: *Classification and Regression Trees*. Wadsworth and Brooks, Monterey (1984)
11. Breitbart, Y., Reiter, A.: A branch-and-bound algorithm to obtain an optimal evaluation tree for monotonic boolean functions. *Acta Inf.* **4**, 311–319 (1975)
12. Chai, B., Zhuang, X., Zhao, Y., Sklansky, J.: Binary linear decision tree with genetic algorithm. In: *13th International Conference on Pattern Recognition, ICPR 1996*, Vienna, Austria, 25–19 August 1996, vol. 4, pp. 530–534. IEEE (1996)
13. Chikalov, I., Hussain, S., Moshkov, M.: Totally optimal decision trees for Boolean functions. *Discrete Appl. Math.* **215**, 1–13 (2016)
14. Delimata, P., Moshkov, M., Skowron, A., Suraj, Z.: *Inhibitory Rules in Data Analysis: A Rough Set Approach*, *Studies in Computational Intelligence*, vol. 163. Springer, Heidelberg (2009). <https://doi.org/10.1007/978-3-540-85638-2>
15. Dua, D., Graff, C.: *UCI machine learning repository*. University of California, Irvine, School of Information and Computer Sciences (2017). <http://archive.ics.uci.edu/ml>
16. Garey, M.R.: Optimal binary identification procedures. *SIAM J. Appl. Math.* **23**, 173–186 (1972)
17. Heath, D.G., Kasif, S., Salzberg, S.: Induction of oblique decision trees. In: Bajcsy, R. (ed.) *13th International Joint Conference on Artificial Intelligence, IJCAI 1993*, Chambéry, France, 28 August–3 September 1993, pp. 1002–1007. Morgan Kaufmann (1993)
18. Hyafil, L., Rivest, R.L.: Constructing optimal binary decision trees is NP-complete. *Inf. Process. Lett.* **5**(1), 15–17 (1976)
19. Martelli, A., Montanari, U.: Optimizing decision trees through heuristically guided search. *Commun. ACM* **21**(12), 1025–1039 (1978)
20. Moshkov, M.J.: Time complexity of decision trees. In: Peters, J.F., Skowron, A. (eds.) *Transactions on Rough Sets III*. LNCS, vol. 3400, pp. 244–459. Springer, Heidelberg (2005). [https://doi.org/10.1007/11427834\\_12](https://doi.org/10.1007/11427834_12)
21. Moshkov, M., Zielosko, B.: *Combinatorial Machine Learning - A Rough Set Approach*, *Studies in Computational Intelligence*, vol. 360. Springer, Heidelberg (2011). <https://doi.org/10.1007/978-3-642-20995-6>
22. Pawlak, Z.: *Rough Sets - Theoretical Aspect of Reasoning About Data*. Kluwer Academic Publishers, Dordrecht (1991)
23. Pawlak, Z., Skowron, A.: Rudiments of rough sets. *Inf. Sci.* **177**(1), 3–27 (2007)
24. Riddle, P., Segal, R., Etzioni, O.: Representation design and brute-force induction in a Boeing manufacturing domain. *Appl. Artif. Intell.* **8**, 125–147 (1994)
25. Rokach, L., Maimon, O.: *Data Mining with Decision Trees: Theory and Applications*. World Scientific Publishing, River Edge (2008)
26. Schumacher, H., Sevcik, K.C.: The synthetic approach to decision table conversion. *Commun. ACM* **19**(6), 343–351 (1976)
27. Skowron, A., Rauszer, C.: The discernibility matrices and functions in information systems. In: Słowiński, R. (ed.) *Intelligent Decision Support. Handbook of Applications and Advances of the Rough Set Theory*, pp. 331–362. Kluwer Academic Publishers, Dordrecht (1992)