svmodt: An R Package for Linear SVM-Based Oblique Decision Trees

by Aneesh Agarwal, Jack Jewson, and Erik Sverdrup

Abstract An abstract of less than 150 words.

1 Introduction

2 Literature Review

Decision Trees

Decision Trees (DTs) are interpretable classification models that represent their decision-making process through a hierarchical, tree-like structure. This structure comprises internal nodes containing splitting criteria and terminal (leaf) nodes corresponding to class labels. The nodes are connected by directed edges, each representing a possible outcome of a splitting criterion. Formally, a DT can be expressed as a rooted, directed tree $T = (G(V, E), v_1)$, where V denotes the set of nodes, E represents the set of edges linking these nodes, and v_1 is the root node.

If the tree T has m nodes, then for any $j \in \{1, \ldots, m\}$, the set of child nodes of $v_j \in V$ can be defined as:

$$N^+(v_j) = \{v_k \in V \mid k \in \{1, \ldots, m\},\ k \neq j,\ (v_j, v_k) \in E\}.$$

Here, $N^+(v_j)$ denotes the set of nodes that are directly connected to v_j through outgoing edges, representing all possible child nodes that can be reached from v_j within the tree structure (Rivera-Lopez and Canul-Reich, 2018).

Decision tree algorithms can be categorized based on whether the same type of test is applied at all internal nodes. **Homogeneous trees** employ a single algorithm throughout (e.g., univariate or multivariate splits), whereas **hybrid trees** allow different algorithms such as linear discriminant functions, k-nearest neighbors, or univariate splits that can be used in different subtrees (Brodley and Utgoff, 1995). Hybrid trees exploit the principle of *selective superiority*, allowing subsets of the data to be modeled by the most appropriate classifier, thereby improving flexibility and accuracy.

Univariate Decision Trees

Univariate Decision Trees (UDTs) trees represent axis-parallel hyperplanes dividing the instance space into several disjoint regions. Axis-parallel decision trees, such as CART and C4.5, represent two of the most widely used algorithms for classification tasks. The **CART (Classification and Regression Trees)** algorithm employs a binary recursive partitioning procedure capable of handling both continuous and categorical variables as predictors or targets. It operates directly on raw data without requiring binning. The tree is expanded recursively until no further splits are possible, after which **cost-complexity pruning** is applied to remove branches that contribute least to predictive performance. This pruning process generates a sequence of nested subtrees, from which the optimal model is selected using independent test data or cross-validation, rather than internal training measures (Breiman et al., 1984).

In contrast, **C4.5**, an extension of the earlier **ID3** algorithm (Quinlan, 1986), utilizing information theory measures such as **information gain** and **gain ratio** to select the most informative attribute for each split (Quinlan, 2014). C4.5 also includes mechanisms to handle missing attribute values by weighting instances according to the proportion of known data and employs an **error-based pruning** method to reduce overfitting. Although these techniques are effective across diverse datasets, studies have shown that the choice of pruning strategy and stopping criteria can significantly affect model performance across different domains (Mingers, 1989; Schaffer, 1992).

While UDTs are highly interpretability, they are characterised by several representational limitations. Such trees often grow unnecessarily large, as they must approximate complex relationship between features through multiple axis-aligned partitions. This can result in the replication of subtrees and repeated testing of the same feature along different paths, both of which reduce efficiency and hinder generalization performance (Pagallo and Haussler, 1990).

Multivariate Decision Trees

Multivariate decision trees (MDTs) extends UDTs by allowing each internal node to perform splits based on linear or nonlinear combinations of multiple features. This flexibility enables the tree to form oblique decision boundaries that more accurately partition the instance space. For example, a single multivariate test such as $x + y < 8$ can replace multiple univariate splits needed to approximate the same boundary. The construction of MDTs introduces several design considerations, including how to represent multivariate tests, determine their coefficients, select features to include, handle symbolic and missing data, and prune to avoid overfitting (Brodley and Utgoff, 1995).

Various optimization algorithms—such as recursive least squares (Young, 1984), the pocket algorithm (Gallant, 1986), or thermal training (Frean, 1990)—may be used to estimate the weights. However, MDTs trade interpretability for representational power and often require additional mechanisms for **local feature selection**, such as *sequential forward selection* (SFS) or *sequential backward elimination* (SBE) (Kittler, 1986).

Empirical comparisons across multiple datasets demonstrate that multivariate trees generally achieve higher accuracy and smaller tree sizes than their univariate counterparts, though this comes at the cost of reduced interpretability. Moreover, MDTs retain key advantages of standard decision trees—such as sequential split criteria evaluation and transparent decision procedures—while offering improved modeling flexibility for complex datasets (Koziol and Wozniak, 2009; Friedl and Brodley, 1997; Liu and Setiono, 1998; Cañete et al., 2021).

Bibliography

L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone. *Classification and Regression Trees.* Wadsworth, 1984. URL http://lyle.smu.edu/~mhd/8331f06/cart.pdf. [p1]

C. E. Brodley and P. E. Utgoff. Multivariate decision trees. *Machine Learning*, 19:45–77, 1995. URL https://api.semanticscholar.org/CorpusID:16836572. [p1, 2]

L. Cañete, R. Monroy, and M. Medina-Pérez. A review and experimental comparison of multivariate decision trees. *IEEE Access*, PP:1–1, 08 2021. doi: 10.1109/ACCESS.2021.3102239. [p2]

M. R. Frean. *Small nets and short paths: Optimising neural computation*. PhD thesis, 1990. [p2]

M. Friedl and C. Brodley. Decision tree classification of land cover from remotely sensed data. *Remote Sensing of Environment*, 61(3):399–409, 1997. ISSN 0034-4257. doi: https://doi.org/10.1016/S0034-4257(97)00049-7. URL https://www.sciencedirect.com/science/article/pii/S0034425797000497. [p2]

S. I. Gallant. Optimal linear discriminants. *Eighth International Conference on Pattern Recognition*, pages 849–852, 1986. URL https://cir.nii.ac.jp/crid/1573668924476144256. [p2]

J. Kittler. Feature selection and extraction. *Handbook of Pattern Recognition and Image Processing*, pages 59–83, 1986. URL https://cir.nii.ac.jp/crid/1573950400671608448. [p2]

M. Koziol and M. Wozniak. *Multivariate Decision Trees vs. Univariate Ones*, pages 275–284. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009. ISBN 978-3-540-93905-4. doi: 10.1007/978-3-540-93905-4_33. URL https://doi.org/10.1007/978-3-540-93905-4_33. [p2]

H. Liu and R. Setiono. Feature transformation and multivariate decision tree induction. In S. Arikawa and H. Motoda, editors, *Discovey Science*, pages 279–291, Berlin, Heidelberg, 1998. Springer Berlin Heidelberg. ISBN 978-3-540-49292-4. [p2]

J. Mingers. An empirical comparison of pruning methods for decision tree induction. *Machine Learning*, 4:227–243, 01 1989. doi: 10.1023/A:1022604100933. [p1]

G. Pagallo and D. Haussler. Boolean feature discovery in empirical learning. *Machine Learning*, 5:71–99, 1990. URL https://api.semanticscholar.org/CorpusID:5661437. [p1]

J. R. Quinlan. Induction of decision trees. *Machine Learning*, 1:81–106, 1986. URL https://api.semanticscholar.org/CorpusID:189902138. [p1]

J. R. Quinlan. *C4. 5: programs for machine learning*. Elsevier, 2014. [p1]

R. Rivera-Lopez and J. Canul-Reich. Construction of near-optimal axis-parallel decision trees using a differential-evolution-based approach. *IEEE Access*, PP:1–1, 01 2018. doi: 10.1109/ACCESS.2017.2788700. [p1]

C. Schaffer. Deconstructing the digit recognition problem. In D. Sleeman and P. Edwards, editors, *Machine Learning Proceedings 1992*, pages 394–399. Morgan Kaufmann, San Francisco (CA), 1992. ISBN 978-1-55860-247-2. doi: https://doi.org/10.1016/B978-1-55860-247-2.50056-5. URL https://www.sciencedirect.com/science/article/pii/B9781558602472500565. [p1]

P. C. Young. Recursive estimation and time-series analysis: An introduction. 1984. URL https://api.semanticscholar.org/CorpusID:60181335. [p2]

Aneesh Agarwal
Monash University

aaga0022@student.monash.edu

Jack Jewson
Monash University
Department of Econometrics and Business Statistics, Monash University, Australia
Jack.Jewson@monash.edu

Erik Sverdrup
Monash University
Department of Econometrics and Business Statistics, Monash University, Australia
Erik.Sverdrup@monash.edu