Improving Performance and Accuracy in Decision Trees: A Literature Survey on Impurity Functions

Abed Alsulami, Prof. Reda Khalifa, Dr. Wajdi Alghamdi
Information Technology Department, King Abdulaziz University, Jeddah, Saudi Arabia

***Abstract*—Decision tree algorithms have remained some of the most popular tools for supervised learning tasks. This has been because of their comprehensibility, malleability, and robustness to accommodate varying types of data. Nevertheless, their accuracy and performance are dependent on the use of impurity measures to guide the process of tree splits. The standard impurity measures, such as the Gini impurity, Entropy, and Classification Error, tend to face problems as the complexity, imbalance, and noise levels are raised. This often gives rise to overfitting. An examination of the disadvantages and present research efforts on impurity-based optimization of decision trees, as well as the study of new paradigms such as the use of Rényi Entropy, the utilization of the Tsallis Entropy, combinations of impurity functions, complexity-aware tree splits, and tailored impurity function augmentation, makes it clear that there are efforts underway to improve the accuracy, robustness, and comprehensibility, as well as the processing complexity, for the tasks of decision tree algorithms.**

***Keywords*—Decision trees; impurity functions; split criteria; decision tree optimization; literature review**

I. Introduction

Decision trees remain widely used in machine learning and data mining because they are interpretable, flexible, and effective for classification and prediction tasks [1][2][3]. Their practical value is especially evident in applications where transparent decision rules are needed to explain model behavior and support human decision-making.

However, the performance of a decision tree depends strongly on the impurity measure used to select splits [2][3]. When impurity is estimated poorly, the resulting tree may overfit the training data, become unnecessarily deep, and generalize poorly to unseen samples [4]. These limitations are more pronounced in complex, noisy, or high-dimensional datasets, where conventional split criteria may not capture the structure of the data effectively.

Decision trees, despite their seemingly simple model, possess the potential for representing and exploring the intricate interdependencies. In addition, decision trees constitute an understandable and easily comprehensible model, making them applicable for exploratory and descriptive data analysis [5]. Nevertheless, if the tree-growing algorithm ineffectively calculates the impurity of the nodes, it would most possibly lead to biased decision tree analysis. The effectiveness of tree growing as an algorithmic process would, therefore, depend on the choice of the impurity measure of the nodes that splits the potential splits. The values of the selected variables influence the point of splits and the corresponding split rule. Since the split rule becomes understandable, the decision tree provides an easily comprehensible representation of the data concerning the classes and the selected covariates. The model would express a tree structure, and the process of interpreting the tree would involve the nodes, branches, and terminal nodes of the tree [6].

II. Background and Significance

In the vast arena of data mining, decision tree algorithms are considered one of the basic tools, much admired for their high degree of interpretability and manipulability. Several decision tree algorithms have come into existence over the years, each having their respective algorithms belonging to CART, ID3, C4.5, CHAID, and then the Random Forest algorithm proposed by Breiman. This wide range of algorithms available, though, becomes one of the most significant challenges, as there always exists the challenge of choosing the best tree algorithm for optimal performance.

It needs to be accepted that the decision tree algorithms, especially for continuous variables, always face the potential challenge of an infinite number of splits and always tend to preserve the distribution of class probabilities, which often leads to the model overfitting. Moreover, the greediness factor always poses a significant challenge, as the basic motive for making the decisions always becomes the search for the highest degree of purity as the basic splitting criterion. These factors, specifically the challenges faced by the decision tree algorithms, like the potential challenge of an infinite number of splits and greediness, are already known and addressed by tree pruning and the addition of the factor of randomness during the development stage [7][8].

III. Research Motivation and Objectives

Even to date, the decision tree remains a basic classification technique in machine learning. Despite the many benefits of the existing classification algorithms based on the decision tree approach, ID3, C4.5, and CART classification algorithms have some limitations in predictive accuracy when applied to imbalanced, noisy, or high-dimensional datasets. One of the main demerits of the classification algorithms discussed above can be attributed to the need to rely on a fixed set of impurity criteria. These include Shannon entropy and the Gini index.

Current research developments in the context of impurity function design techniques—such as the application of general concepts of entropy, Rényi entropy, and the Tsallis entropy approach applied to RS-entropy—have shown promising results. There also exists a great scope of empirical research in the context of the application of improvements in the impurity function. At the same time, there also exists a research gap in terms of the complex relationships of impurity in trees. This research aims to improve the accuracy of the decision tree classification technique and prevent overfitting by designing new impurity measures. These research objectives are:

- To investigate the inadequacy of the classic measures of impurities (e.g., Gini index, Shannon entropy) in the context of decision trees for different datasets.
- To develop new impurity criteria involving general measures of entropy and the consideration of parameters for flexible evaluation of splits.
- To test the proposed impurity measures empirically on a series of benchmark databases in order to see how well the measures perform in terms of accuracy in comparison to traditional decision tree algorithms.
- To test the effect of the improved impurity functions when applied in combination with the concepts of pruning, ensemble learning techniques, and feature selection.
- To establish the statistical significance and generalizability of the devised methods by performing validation techniques such as cross-validation and non-parametric tests.

Achieving the above objectives would ensure the advancement of the existing decision tree approach to one that provides higher accuracy and robustness without compromising the need for interpretability in trustworthy AI.

IV. Structure of the Literature Review

The literature review in this study is organized into a logical and progressive structure designed to contextualize the role of impurity functions in decision tree construction, highlight current scientific gaps, and synthesize contemporary research addressing these gaps. The structure is intentionally crafted to guide the reader from foundational knowledge toward advanced theoretical and empirical developments. It is organized into the following major components:

A. Foundations of Decision Tree Methodology

This section discusses the theoretical background of decision tree algorithms, their history, and their universality within the machine learning and data mining communities. This section explains the important attributes such as explainability, hierarchical representation, and flexibility and how the use of impurity measures is fundamental to the design of high-performance tree-based algorithms. This section helps the reader understand the challenges of continuous features, greedy splitting, and overfitting.

B. Classical Impurity Functions and Their Limitations

This section of the study evaluates commonly employed impurity estimates, such as the Gini Index, Shannon Entropy, and Error, along with their definitions, mathematical formulation, and theoretical characteristics. This section lays considerable emphasis on the performance problems associated with conventional impurity estimation algorithms, which are prone to difficulties such as imbalance, noise robustness, scalability, and the generation of complex decision trees.

C. Emerging and Alternative Impurity Measures

This section integrates the latest research efforts for improvement and new forms of impurity values, such as the use of Rényi entropy, Tsallis entropy, MECC, and combinations. The latest research on comparisons between such impurity measures and classical impurity measures, as well as the advantages derived from tunable parameters and probabilistic formulations, are discussed.

D. Decision Tree Performance Challenges

After the discussions on impurity functions, the following section looks into the persisting issues involved in the creation of decision trees, such as overfitting, sparseness, missing values, instability, and lack of generalizability. This section combines the findings regarding the role of impurity functions in the above issues and the gaps for the creation of tailored impurity functions.

E. Complementary Optimization Techniques

Impurity measures alone cannot completely address the shortcomings of decision trees. In this section, we review methods that can improve performance when used in conjunction with improved impurity measures. The main strategies are:

- Pruning techniques, including cost-complexity pruning, error-based pruning, and bootstrap out-of-bag pruning.
- Feature selection methods, namely filter and wrapper approaches.
- Ensembling techniques, including Bagging, Boosting, and hybrid models.

Literature suggests that the incorporation of advanced impurity functions coupled with these complementary techniques results in decision trees that are more stable and generalizable.

F. Tree Optimization: State-of-the-Art Developments

This section covers modern developments, including gradient-based learning of decision trees, differentiable tree architectures, splitting criteria driven by changes in distributions, and parallelized frameworks for building trees. It also provides an overview of research into weakly supervised anomaly detection, financial modeling, educational analytics, and explainable AI to understand real-world usage and effectiveness of state-of-the-art tree optimization techniques.

G. Identified Gaps and Research Opportunities

The final section synthesizes the literature to identify specific enduring gaps:

- The need for impurity functions that are fit for complex, heterogeneous data.
- Lack of adaptive mechanisms for impurity selection.
- Limited integration with explainability and interpretability tools.
- A lack of theoretical guarantees for new impurity formulations.

These gaps motivate the current study, which seeks to systematically investigate and appraise customized impurity functions that enhance accuracy, reduce overfitting, and improve computational efficiency in decision tree learning.

V. Decision Tree: An Overview

A decision tree is indeed one of the most powerful algorithms in supervised learning for both classification and regression tasks. It is still the most widely used algorithm today and can be visually represented as a flowchart, where each leaf will correspond to the output that is expected from the decisions made from the internal branches. The tree needs to be structured from a root down to the leaves by balancing readability and performance (see Fig. 1) [9]. It should be compact and use easily interpretable, stable variables with solid classification power as measured by its impurity. This is significant in keeping the tree concise for making sure predictive strength is maintained, especially in high-stake decisions where early trees should still accommodate some level of uncertainty [2].

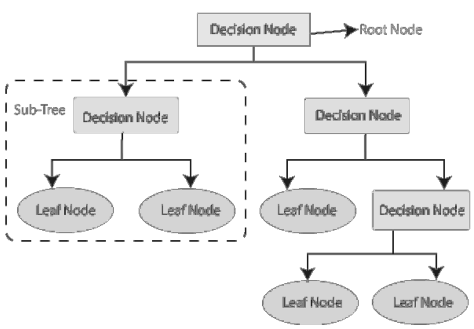

Fig. 1. Traditional decision tree [10].

A. Definition and Characteristics

Decision trees are generally known and accepted as basic techniques used in machine learning. The flexibility offered by decision trees is due to the ability to handle various attribute types, such as nominal, ordinal, interval, and continuously scaled attributes. In nominal attributes, the value is tested, and specific value information is used to transition to the next leaf [11]. In contrast, neighboring value ordering is used to establish a cut point, showing symmetry to the cut point, with the attribute value of the present case indicated by belonging to the region defined by the cut points. When the value of the region and corresponding cut points are divided into the region defined by the cut points, the attribute itself is said to be multidimensional. Varying the value of the region and the cut point could trigger a leaf transition, showing that the decision tree structure could vary by adjusting attribute values to the figure defined by quadruplets [12]. The ability to divide and conquer nominal and ordinal attributes enables the easy convergence of decision trees to attribute types. Further, using trees to continuously scaled attributes is enabled due to the structure of the leaf node, allowing the use of decision trees on problems involving interval, spatial, and continuously scaled attributes [13].

Forward pruning techniques, used to improve simplification models, are normally dependent on validation data, which is further divided into training and validation samples. These algorithms work to optimize decisions to obtain the lowest possible tree, which is guaranteed to keep validation samples having lower individual errors than training sets' probable errors. This is known as the cost-estimation approach, which expects methods used to estimate decision costs to avoid errors associated with performance, so as to derive an expansive and correct tree to predict and adapt to new and changing test/live situations.

B. Common Impurity Functions

In light of identifying Gini impurity and entropy as the most common impurity methods, the importance of the essential differences between their usages in decision tree building is discussed [14]. In comparison to more balanced datasets, Gini impurity is generally more biased toward the majority class, while entropy is generally more balanced and tends to generate smaller trees. In support of the comparative assessment, it has been found to work best on balanced datasets and class distributions [14]. This could explain why, under various datasets, Gini impurity has been found to outperform entropy on several instances [15][16]. When most instances are negligible, it has been observed to have less bias compared to entropy, which tends to reduce differences between impurities by biasing individuals to identify a more significant categorization ordering, albeit generating more suitable root node splittings [17]. In contexts when explainability is most paramount, the smaller-sized trees generated by Gini impurity could prove to be more beneficial, owing to their ease of understanding. Moreover, it is also more applicable to real-case problems due to high efficiency [18][19].

VI. Impurity Functions in Decision Trees

In decision tree models, nodes are represented by the attributes contained in the datasets, and the leaf node is used to define the class to which the datasets belong [20]. In discriminative learning, it is essential to identify the decision nodes by calculating the impurities, Eq. (1), present at the nodes used to divide the datasets. Given the fact that a node holds the impurities present in the datasets, the choice of input variables is important at the node during learning [21]. Various impurity measures have been developed to cope with input variable selection [21]. Impurities used to build decision trees include basic measures on errors, such as information entropy, the Gini index, and rule-based methods [22]. Using various impurities and a fixed probabilistic constraint along with learning on nodes, the idea of I-Optimal trees is developed based on the basic set of I-Optimal responses to probability, which aims to minimize the maximum non-negative relative entropy between the errors and the trees, and to develop insertable trees based on information functions and two-class tree optimization [23][24].

$$\text{impurity} = 1 - \text{purity} \tag{1}$$

Decision trees have found broad applications in data mining [25]. Decision trees have long been known to effectively and significantly provide easier methods to understand complicated procedures involving decision-making [26]. Moreover, the interpretable attribute of decision trees is unique to the algorithm, adding to their popularity [27]. Compared to other models used to perform classification, decision trees are surprisingly more interpretable [28]. Decision trees have found broad applications ever since their creation and belong to a broad array of fields, including but not restricted to, bioinformatics, simplification of constraint programming problems, engineering, machine learning, modeling survival probability of trees, natural resource management, learning parity problems, and virtual screening, apart from applications to ensembles and forests [29][30]. Various extensions to the basic concept of decision trees have been developed, including but not restricted to, conic trees, conditional inference trees, decision stumps, frequent decision trees, interactive fuzzy decision trees, multigrain trees, multitrigger model trees, quantile random forests, totally corrective linear decision trees, and weight boosting [11][31].

A. Gini Impurity

The Gini impurity has several advantages. First, it significantly handles the issue of class imbalance. While it is ideal to have a balanced learning set, the Gini impurity has a tendency to lean towards the smaller classes, thus increasing their chances of reaching the top of the decision tree [32]. Moreover, it places the most frequent class at the top of the tree, having a greater influence than any other class. As a result, it quickly divides the large set into the upper nodes of the tree. Thus, the Gini impurity is often considered one of the most suitable impurity measures to be used in decision trees and random forests. The integrity of the tree model would significantly be affected if inappropriate measures were used [33]. In addition, the Gini impurity is also effective in reducing the complexity of the tree, and the efficiency of the tree-building algorithm is increased due to the reduced number of operations [34]. Moreover, it has been evident that the computational complexity of the Gini impurity is lower than information gain, and it is computationally efficient [35].

The Gini impurity measures the probability of randomly choosing an instance that would be assigned to a different class than it should. For a node t, it is defined as Eq. (2):

$$\text{Gini}(t) = 1 - \sum_{j=1}^{k} p_j^2, \tag{2}$$

where, p_j is the proportion of instances of class j in node t and k is the number of classes [36]. The Gini impurity has value 0 when instances belonging to a node are maximally pure (all in a single class), and in the binary case it reaches its maximum of 0.5 when instances are equally distributed between the two classes, which corresponds to maximal impurity [37].

B. Entropy

Mathematically, the concept of entropy has been used to define the measure of Eq. (3) and Eq. (4) disorder or unusable energy present at the molecular level [38]. In contrast, the concept of entropy has been used to define the average amount of information, the measure of uncertainty associated with a source, and also the amount of randomness associated with a message. In essence, the more diversity, the more entropy [39]. Another concept, conditional entropy, is used to define the remaining uncertainty associated with the random variable, having knowledge of another variable, and is given by $H(Y|X) = S(I)$ [39]. There is a relationship between conditional entropy and entropy. In decision trees, the concept of information has been defined to mean the impurity associated with a dataset [40].

$$E(S) = -\sum_{x \in S} p(x) \log_2 p(x) \tag{3}$$

$$IG(S, A) = E(S) - \sum_{t \in T} p(t)E(t) = E(S) - E(S|A) \tag{4}$$

1) Rényi entropy: Current literature has further revived interest in using Rényi entropy, Eq. (5), to increase the correctness of decision trees [41]. This has been proven to outperform standard methods of measuring uncertainty and has prospects to tackle problems of unbalanced datasets. The impact of several Rényi measures on the efficiency of decision trees has also been analyzed, reflecting the ability to measure uncertainty associated with datasets to improve the accuracy of models [42]. Studies have also attempted to measure Rényi uncertainty along with other measures to increase decision tree efficiency [43].

$$R(D) = \frac{1}{1-\gamma} \log \left(\sum_{i=1}^{k} p_i^{\gamma} \right) \tag{5}$$

2) Tsallis entropy: When used as an impurity measure, Tsallis entropy, Eq. (6), has been proven to increase the efficacy of decision trees, as the inclusion of unconventional measures of impurity, including Tsallis entropy, has led to optimal results in decision tree models [44]. The importance of understanding the potential impact of Tsallis entropy on decision tree performance should not go unnoticed. Moreover, the inclusion of Tsallis entropy has led to significant improvements in the accuracy of decision trees. Cumulatively, the literature suggests that the inclusion of Tsallis entropy could prove to be a worthy tool to improve decision tree performance [41].

$$S_q = \frac{1 - \sum_{i=1}^{k} p_i^q}{q-1} \tag{6}$$

C. Classification Error

The impurity measures under consideration demonstrate the importance of weights used when determining impurities [45]. The classification error function, Eq. (7), is used as an impurity measure on the terminal set and is a measure used to evaluate the splitting attribute, which depends on the selection attribute used to decide on splitting [46]. The classification error function is universally suited to a number of classes and is known by the name Gini impurity coefficient when used on two classes, and it is one of the impurity measures used in the empirical field. When achieving accuracy is assessed, a high-value item could incur more costs than a lower-value item [47]. The relevance of understanding measures of cost or loss is critical to accurately compute splittings and variables [48].

The aim of the present review is to demonstrate the potential role of the use of ranging functions on the flexibility offered by decision trees to social scientific studies [49]. In some situations, it is also possible to utilize a boosting algorithm with closely similar statistical power to handle various MC matrices [50].

$$E(D) = 1 - \max_i(p_i) \tag{7}$$

VII. Decision Tree Performance Challenges

Decision tree induction involves mainly three broad phases: formation of the tree structure, pruning, and leaf node assignment. Formation of an efficient decision tree is very important, and it is always desirable to have a decision tree with similar class distributions of the child nodes [12]. Nevertheless, it is often difficult to satisfy such a restriction when working on practical problems. In addition, it is more complex to develop a homogeneously structured decision tree with morphological information than to generate an effectively shaped decision tree [29].

Decision trees are a widely used technique for classification and prediction. The strength behind decision trees lies in their simplicity, usefulness, and ability to capture relationships with a fewer number of restrictions on their structure. This technique is helpful when it is required to work on moderate to large-sized datasets and involves very little input information on specifying the problem. It is, however, important to keep in mind that decision trees are vulnerable to noise present within the input, which often causes the creation of a large number of rules [51].

A. Overfitting

As data scientists and machine learning experts, researchers have the power to set criteria to govern tree models to avoid overfitting. This is possible by incorporating the number of training samples (n) and the number of relevant variables to predict the target variable (m), such that n is at least equal to $2m$ [52]. The number representing relevant variables is used either in the leading section of the data divide or by adding up all the conditions used to classify each training sample to the target category. This enables each category to hold distinct activated conditions to facilitate the validity of inferential modeling. Although simplistic, it is very effective to control the structure and number of relevant input variables to grow the decision tree modeling approach. Other techniques involving the effective utilization of weighted and optimized systems to identify relevant attributes and unique amounts of data to correctly categorize are also capable of increasing the efficacy of the aforementioned rule to $n \geqslant 2m$ [53].

The cause of overfitting can therefore be said to result from the inclusion of noise and unnecessary attributes, which the tree is trying to capture and employ to classify or explain the training data. The overfitting of the data is not sudden during the attribute selection procedure but starts with the selection of the optimal attribute and the inclusion of less optimal attributes to the tree nodes. The tree continues to grow until a considerable amount of irrelevance and interference is injected into it. The problem occurs because the training data have been optimized solely to suit the specific data the tree is growing on and may result in incorrect classification during the presentation of new instances [54]. Beginning with decision trees, overfitting is a basic flaw associated with them owing to their ability to learn and generalize data but not accurately enough. Noisy data during training are also associated with this flaw. When the attributes of the tree grow to complexity with too many nodes compared to the number of training instances, it is said to experience overfitting, marked by too much unnecessary modeling [52].

B. Handling Missing Values

Decision trees are commonly used to classify instances within a specific domain defined by a problem. The instances are divided into several sub-domains, and a model is built based on each domain and a set of attributes. This is conducted recursively. Several other learning methods are also found, which contain nodes not connected to decision trees. These methods are mainly used to improve the induction of decision trees [55].

Decision trees with missing information can also be handled using measures of impurities, which determine how to treat the missing information. Missing information may require filling using the most frequent value, and another measure of impurities may also be used to measure the influence of the missing information on the decision tree. When attributes are continuous, the average is commonly used, and when discrete, the mode is used. After structuring information and classes associated with information, the algorithm determines the best splitting and modes.

On the other hand, Quinlan developed the C4.5 algorithm to build decision trees, which works effectively with attributes that require either continuous or discrete information. It divides instances into various areas defined by a test and assigns instances to paths showing maximum information gain. In the meanwhile, Fernando et al. have made improvements to develop more efficient training and increase the efficiency of neural learning by reducing the influence of missing information [56]. Some authors also go into detail on how missing data are handled during splitting when constructing decision trees, while others consider it another value to take into consideration along with other attribute values. As a result, the diversity discussed above encompasses a wide review of the subject to guarantee readers are knowledgeable on the various perspectives on the issue [57].

VIII. Enhancing Decision Tree Performance

Current algorithmic developments involving decision tree impurity are an interesting solution, although not without restrictions [58]. Moreover, the existence of impurities is not convex, thereby increasing the complexity of reaching an optimal solution, and it is not guaranteed to converge within a fixed time span dependent on the input attributes [59]. Thus, it has also been realized that, to a great extent, the rigid measure of impurity is often settled with the measure of minimum quality, with the hope of achieving optimal models with impurities below a certain threshold. Unfortunately, such relaxation of impurity measures has further triggered unwanted occurrences of overfitting decisions on the decision trees [60].

Decision trees, although having high capabilities on classification problems, are also faced with demerits, largely due to the presence of noisy information in the datasets. Most learning models are also known to be highly affected by impurities, and more so when it is regarding class imbalance [28]. Although it has been realized to a great extent that having robust positions against impurities and enough data regarding impurities to learn is important, impurity is not invariant. If it is considered to a large extent, it is known to negatively affect learning models, and if the impurities are considered to a lesser extent, it often has to consider a rule regarding the pure attributes of the minority class and is further embedded into the decision trees. Under such conditions, it has been realized to largely overfit the trees, correctly categorizing instances in the minority classes and largely misclassifying instances in the majority classes [61].

A. Pruning Techniques

There is a possibility of overfitting during the creation of a decision tree, which may result in the creation of a very large tree [62]. Although the aim is to develop an apt representation of data, it often tends to remember the data and perform very well on it. This problem can be mitigated through pruning. Pruning the tree can be achieved by cost-based pruning using the resubstitution error, the estimate of the error, or by methods such as the bootstrap out-of-bag error, which serves as a validation technique and is very useful, particularly when reducing the time complexity [63]. The bootstrap out-of-bag error can also act as a third option that combines these techniques and serves as a validation procedure, and is very helpful, particularly when reducing the time complexity.

The entire process of pruning allows the creation of an unbiased terminal tree, which provides efficient access to the basic structure of the data, namely the relation between impurities and the decision tree [64]. By using metrics such as the Gini index and entropy, along with validation and techniques such as the random forest, it is possible to increase robustness, which allows the creation of a better and more universally applicable model [65]. Decision tree methods are used owing to their immediacy and ability to divide observations into separate subsets, which results in them becoming an extremely efficient tool and very difficult to reproduce and verify by other techniques such as multivariate methods [27]. Decision trees along with impurities are flexible methods and can handle nominal and continuous independent variables [4]. The technique, which is similar to a set of binary tests, is able to identify optimal cutting points on the independent variables to increase the purity of the groups, which is very helpful to behavioral analysts and market experts, and is able to provide very rapid characterization and understanding of the data. Hence, it is able to act as an extremely efficient description tool, identifying the region of interest surrounding the most basic subsets of various variables [66].

B. Feature Selection Techniques

Input feature priority techniques imply ranking input features based on relevance to the target task. This is helpful to identify the optimal number and priority of input features to retain. Input feature selection is divided into two broad categories: filter and wrapper techniques. In filter techniques, importance is ranked, and measures such as the t-statistic and F-statistic are used. These techniques can select subsets of top-k features and are used during the first stages of data preprocessing. There are various input feature selection techniques, and some of the most important ones are discussed below.

In contrast, wrapper methods utilize a search procedure along with a model to assess the performance of various subsets of features [67]. Features are selected and dropped based on their ability to improve or hinder the performance of the model [68]. In contrast to filters, the wrapper technique can select a feature that is important to the model yet not directly associated with the target, and drop a feature that is less important but directly associated with the target [3]. As such, the wrapper technique often outperforms filters [69].

The objective of the feature selection procedure is to discover a set of attributes within the input features to enable the creation of a predictive model with enhanced accuracy, which can develop a more efficient predictive model by learning the most distinguishing features while avoiding unwanted characteristics [70]. Several advantages are associated with feature selection, including the creation of more interpretable models, the ability to develop models with faster speeds and smaller sizes, the elimination of variables highly associated with the target, and the attainment of negligible value. There are two types of feature selection methods [5, 71].

C. Ensemble Methods Techniques

In the decision trees literature, an ensemble technique combines N models (developed by using the same input learning algorithm) to attain a unique prediction for every instance within the test data [72]. There are two basic methods to design an ensembling technique of decision trees [73]. Bagging involves creating an ensemble by randomly sampling, with replacement, elements within the training example and using the sample to develop the respective model [74]. Boosting techniques are based on an iterative technique wherein models are developed on the entire training example, and weights are finally given to poorly predicted instances (see Fig. 2) [75]. These poorly predicted instances also display a greater tendency to be selected into the training example of the next models [76][77]. These methods are further coupled with decision trees to generate an ensembling technique of models [78][79].

1) Bagging: When it is used correctly, involving decision trees as base models, bagging can generate very good predictive models. However, it should never be forgotten that bagging works mainly to decrease variance, which is why it is most beneficial when used on high-variance models. Another important factor is to include homogeneous and robust base models. When base models are not satisfactory, it is possible to improve performance by differently initializing base models randomly and using differently defined discretization methods, such as divide techniques and Vapnik methods [80]. While it is true that, when used correctly, bagging and resampling methods can often involve more active and complex processes than other techniques used to generate ensembles, it is also very important to understand and appreciate the strength of bagging, which is to accurately and efficiently develop predictive models [81].

The basic bagging algorithms require a robust learning algorithm, which is true when bagging is used with decision trees [11]. When used on decision trees, bagging produces a mixture model where each individual model is obtained using a bootstrap sample of the training instances, and the training involves all instances and variables in the training set. The predictions of the individual models (decision trees) are averaged in regression and combined via voting in classification [76]. The best possible results are achieved when the training is carried out on noisy data and data with a high dimensionality. Bagging is most effective when it decreases the variance of the base learner when it is unstable.

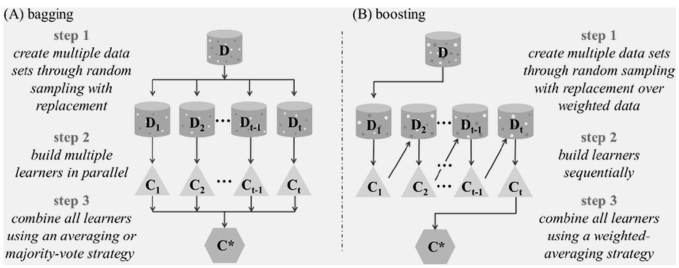

Fig. 2. Bagging and boosting tree-based ensemble [10].

2) Boosting: XGBoost, which is a very reliable and highly researched boosting algorithm, is very similar to residual-fitting boosting but is distinguished by adding a specific weight to each classifier based on the accuracy rate. The weights are incorporated into the overall classifier, which makes XGBoost a forward stage-wise fitting algorithm [82]. Unlike traditional methods, XGBoost gives more weight to instances that were classified inaccurately to influence the learning process of new classifiers. The concept of boosting has the ability to improve a weak solution to a strong solution [6].

Boosting techniques present the problem of optimization by working on iterative improvements of specific loss functions by training on instances with more importance. The next classifier is dependent on the most important instances, which are determined by the distribution defined by the earlier classifier. These techniques, such as AdaBoost and RankBoost, increase the importance of instances by modifying the distribution after each classifier. Other boosting techniques, such as BrownBoost, work differently by approximating the loss function without changing the distribution after each classifier [73][79].

IX. Related Work

Mao et al. [83] present a new decision tree classification algorithm based on a two-term RS-entropy concept, aiming to improve the flexibility and accuracy of models. The authors highlight the insufficiencies of existing decision tree approaches such as ID3, C4.5, and CART, which rely on rigid splitting rules defined by Shannon entropy and the Gini index. RS-entropy, a generalized concept encompassing Shannon, Rényi, Tsallis, and r-type entropies, allows dynamic splitting rules defined by adjustable parameters. Furthermore, the authors enhance the two-term information formula by incorporating penalty factors to account for attribute complexity, resulting in the RSEIM algorithm. Empirical experiments on eight UCI datasets show that RSE and RSEIM greatly outperform traditional algorithms in classification accuracy without increasing model complexity. Genetic algorithm–based parameter optimization further strengthens RSEIM's robustness and resistance to overfitting. Pairwise comparisons and Nemenyi tests confirm the statistical superiority of RSEIM.

He et al. [84] propose a dynamic Bayesian node-splitting algorithm to improve the performance of the random forest algorithm by overcoming inefficiencies in traditional splitting methods. The algorithm dynamically calculates posterior probability distributions during the splitting process, enabling more accurate identification of optimal split points. The authors also introduce a dynamic thresholding mechanism weighted by misclassification errors to prevent overfitting. Experiments conducted across multiple UCI datasets show improvements in accuracy, robustness, and computational efficiency compared to classical random forest models.

In the work of Ferreira et al. [85], decision tree models are used to support clinical decision-making regarding impaired physical mobility in trauma patients. Using the C4.5 algorithm, the authors identify eight essential nursing diagnoses and construct classification trees to determine key predictors such as age, length of hospital stay, and injury severity score. The results demonstrate that decision trees can effectively guide clinicians in prioritizing patient care and integrating data-driven assessments into trauma management.

Ciftcioglu et al. [86] introduce a predictive model for estimating shear strength in reinforced concrete beams, optimizing decision trees using the Lévy Flight algorithm to overcome the limitations of traditional machine learning approaches. Using an extensive dataset featuring numerous structural variables, the authors compare classical decision tree models with Lévy Flight–enhanced variants. Results show that the optimized models produce superior accuracy, robustness, and adaptability while reducing overfitting. Metrics such as RMSE, MAE, and R^2 confirm that the new method significantly outperforms traditional decision trees.

Li et al. [87] present an extensive analysis of a decision support system (DSS) for vending machines using machine learning techniques, including an improved C4.5 decision tree algorithm and neural networks. The modified C4.5 algorithm reduces runtime and slightly improves accuracy relative to the original. Using a backpropagation neural network (BPNN), the authors show strong alignment between predicted and real sales trends. Reinforcement learning is also incorporated to further enhance the DSS's efficiency and forecasting accuracy.

New splitting rules for Gradient Boosting Decision Trees (GBDT) are proposed by DeLise [88] to address out-of-distribution (OOD) generalization—a major challenge due to era-dependent distribution shifts. Unlike Empirical Risk Minimization (ERM), which assumes i.i.d. data, the new rules leverage cross-era information to build more robust models. Theoretical analysis and experiments using simulated datasets and real-world financial market data (e.g., Numerai) show improvements over classical GBDT.

Marton et al. [89] introduce GradTree, a technique for learning hard, axis-aligned decision trees using gradient descent rather than traditional greedy methods. By employing backpropagation and the straight-through estimator, GradTree jointly learns all tree weights, producing enhanced accuracy and generalization performance in binary and multiclass tasks. Experimental results demonstrate that GradTree surpasses existing algorithms while supporting external loss functions and dynamic splitting.

Zeng [20] examine fundamental assumptions about impurity functions in decision trees, focusing on non-negativity and concavity. The authors show that impurity functions are not inherently non-negative and analyze conditions where impurity decreases may be harmful. They further demonstrate that concavity guarantees non-negative impurity decreases in multi-way splits. Experiments on the German credit dataset show that a linear hybrid of the Gini index and entropy outperforms each impurity measure individually.

In their study, Arbiv et al. [90] explore human-centered optimizations for decision trees to improve interpretability. The authors evaluate design modifications and algorithmic enhancements aimed at reducing the gap between complex data structures and user understanding. Their findings show that optimally designed trees significantly improve human decision-making by presenting information in a more comprehensible format.

Loreti and Visani [91] assess parallelization strategies for building decision trees in the GLEAMS explainability framework, which struggles with scalability. By applying distributed computing through the Ray platform, they compare multiple parallel methods and identify performance strengths and weaknesses. The results highlight improvements in efficiency and scalability in alignment with modern explainability demands.

Finke et al. [92] investigate tree-based methods for weakly supervised anomaly detection at the Large Hadron Collider (LHC). Their work demonstrates that Boosted Decision Trees (BDTs) outperform deep neural networks in training efficiency and robustness to noise. Using gradient boosting and ensemble techniques, the authors show that BDTs can effectively operate in high-dimensional spaces with many uninformative variables, improving model-independent searches for new physics phenomena.

Baena-Rojas et al. [93] present a hybrid approach to Multi-Criteria Decision-Making (MCDM) to identify international markets for small and medium-scale coffee roasting businesses. The hybrid approach combines and balances the benefits of quantitative and qualitative techniques to improve the efficiency and viability of International Market Selection (IMS). A case study involving 18 Colombian SMEs producing coffee shows that cultural variables and techniques improve the efficiency of the MCDM solution. The most promising countries are the US, the Netherlands, and South Korea, demonstrating the feasibility of the proposed solution.

Ujwal and Malik [94] suggest a system called PerformanceX, which utilizes Educational Data Mining (EDM) to enhance performance and minimize dropout ratios. The authors present a selective algorithm involving filter and wrapper methods to select important attributes based on student datasets. Using machine learning classifiers, including a fusion voting classifier, the system achieves an accuracy of 99.41%, with the best result obtained by the random forest classifier. The work clearly emphasizes the relevance of structuring student performance on effective attributes to improve learning and career advancement.

Bouke et al. [18] introduce the BukaGini algorithm, a new feature selection algorithm suitable for machine learning. The algorithm leverages the concept of the Gini impurity index to improve the evaluation of interactions between features, including linear and nonlinear associations. Comparisons conducted on various datasets, such as High School Students' Performance, Cancer Data, Spambase, and UNSW-NB15, indicate that BukaGini generally attains greater accuracy improvements than conventional Gini index methods, ranging between 0.32% and 2.50%.

Pathan and Sharma [95] analyze improvements to the decision tree algorithm to increase classification accuracy and minimize tree size. Their discussion of preprocessing techniques focuses on handling missing information and uncertainty. The proposed decision tree algorithm, optimized using the Gini impurity, is more efficient than regular decision trees based on criteria such as precision, recall, F-score, and accuracy.

Subramani and Muruganantharaj [96] propose the EnhancedTree+ algorithm, an advanced decision tree classifier designed to enhance accuracy, interpretability, and robustness when dealing with complex datasets. Although decision tree algorithms are easy to implement, they suffer from overfitting and loss of interpretability when handling complex data. EnhancedTree+ addresses these problems by incorporating smarter splitting rules and pruning. Extensive experimentation on the Titanic dataset shows that EnhancedTree+ outperforms standard decision tree classifiers and competes favorably with models such as random forests and gradient boosting. The experiments clearly indicate improvements in accuracy, precision, recall, and F1-score, implying strong practical viability.

Asif et al. [97] address global health issues related to cardiovascular diseases by presenting a machine learning solution involving preprocessing, hyperparameter tuning, and ensemble learning techniques. The authors combine three Kaggle datasets with similar attributes to construct an extensive dataset. Using an Extra Trees classifier, normalization, and grid search cross-validation, the proposed solution achieves an effectiveness of 98.15%. The results suggest strong potential to correctly identify cardiovascular disease, which may greatly aid in prevention, diagnosis, and treatment, thereby reducing morbidity and mortality.

The work by Cui et al. [98] targets the robustness of Gradient Boosted Decision Trees (GBDT) and offers a technique to improve it by employing one-hot encoding and regularization. Although efficient, GBDT models are vulnerable to slight changes in input data. By converting the models into linear form using one-hot encoding and applying L_1 and L_2 regularization, substantial improvements in robustness are achieved. The study shows that regularization methods help alleviate vulnerabilities to data perturbations, ultimately improving efficiency.

Custode and Iacca [99] explore the integration of evolutionary algorithms and Q-learning to provide interpretable decision trees for reinforcement learning (RL). Their work tackles problems such as premature convergence to local optima by adopting a two-level optimizer. Experimental analysis conducted on classical RL tasks (CartPole-v1, MountainCar-v0, and LunarLander-v2) and on a pandemic control domain clearly shows the competitive efficiency of the proposed solution for both interpretable and non-interpretable models. The results demonstrate that interpretable models can match or even improve upon the performance of non-interpretable ones, even in high-stakes domains such as pandemic control.

Riansyah et al. [100] publish a study on improving the accuracy of the C5.0 decision tree algorithm for data mining classification using adaptive boosting (AdaBoost). They highlight several flaws of the C5.0 algorithm, including susceptibility to overfitting and accuracy degradation when dealing with unbalanced data. These flaws are mitigated by AdaBoost, which adjusts the weights of incorrectly classified samples. In a case study involving hepatitis data, accuracy is increased from 80.58% to 82.98%, while the classification error rate is significantly reduced. This shows AdaBoost's ability to transform a weak classifier into a strong one through reweighting of training samples.

In their work, B et al. [101] present Permutation Decision Trees (PDTs), an extension of classical decision trees that uses the Effort To Compress (ETC) complexity measure—interpreted as an impurity measure—to enable permutations of instances. Compared to Shannon entropy and Gini impurity, ETC captures the importance of instance ordering, making PDTs sensitive to permutations. The authors also propose Permutation Bagging, an implementation of PDTs that does not rely on stochastic sampling but instead uses random feature selection. Although the approach has promising applications, comparisons with standard random forests clearly show the superiority of Permutation Bagging and Permutation Decision Forests, particularly when handling critical data instances and dependencies.

Awad and Fraihat [102] study the efficacy of recursive feature elimination with cross-validation using decision tree models (DT-RFECV) for feature selection in intrusion detection systems (NIDSs). Employing the UNSW-NB15 dataset with 42 features, they improve accuracy while reducing the number of features. The findings reveal that using the optimal 15 features obtained by DT-RFECV yields a binary classification accuracy of 95.30%, comparable to that achieved using the full feature set. This work maintains high accuracy while speeding up detection and reducing storage requirements by 65% through feature selection.

Zhang and Gionis [16] propose a new algorithm for decision tree induction that incorporates complexity guarantees during learning. Although classic algorithms such as C4.5 and CART are effective in terms of accuracy, they offer no guarantees regarding the complexity of the constructed trees and may thus produce poorly interpretable models. The new algorithm improves on these approaches by adding a greedy heuristic that balances tree size, cost, and impurity, yielding logarithmic guarantees on complexity. As a result, it produces trees that are both interpretable and accurate. The paper represents an important contribution by explicitly addressing the trade-off between accuracy and interpretability.

Roshanski et al. [103] propose a new algorithm, FACET, designed to ensure that decision trees remain compact while preserving modeling power. Although decision trees often provide a simple and robust modeling framework, their growth can lead to overly large and complex models. FACET seeks to maintain compactness by using automated modeling principles and feature extraction to construct concise tree representations. Experimental analysis on 82 different datasets is conducted to verify the efficiency and effectiveness of the proposed technique.

Bodine and Hochbaum [104] present the Max-Cut Decision Tree with modified PCA, published in SN Computer Science, with the aim of optimizing the decision tree algorithm. Their work remedies problems associated with standard CART Gini trees by incorporating two major innovations. First, Max-Cut is designed to maximize separation between class instances, making it less prone to overfitting. Second, the integration of Node Means PCA dynamically adapts dimensions at each node, reducing running time and improving efficiency. Experimental validation across multiple datasets, including CIFAR-100, shows considerable performance improvements.

Horváth et al. [105] propose a new method for adding robustness to the certification of decision-stump ensembles via deterministic smoothing. Although tree-based models are widely used in high-risk domains such as healthcare and finance, their robustness certification lags behind that of neural networks. The authors introduce deterministic robustness certificates for ensembles with numerical and categorical attributes. Their method uses dynamic programming to compute exact output distributions under randomization and provides an MLE-optimal learner for robust stump-based boosters. Experiments demonstrate improved efficiency and accuracy compared to existing robustness-certification approaches for tree models.

Li and Chen [106] address the shortcomings of conventional univariate decision trees (UDTs) by introducing a new splitting rule based on Geometric Mean Metric Learning (GMML). They propose the diagonalized GMMLDT, which accelerates tree construction by quickly identifying the most discriminative splitting feature while preserving explainability. Experiments on standard benchmark datasets show that both dGMMLDT and its multivariable extension, dGMMLMDT, achieve comparable or superior accuracy with up to a ten-fold speedup relative to full GMML.

Lee et al. [33] propose a decision tree algorithm that improves classification accuracy by incorporating class distances. The method modifies the traditional Gini index using a combination of Rao and Stirling distances, producing more clearly separated class boundaries during splitting. Testing on multiple datasets—including contraceptive method choice, car evaluation, and yeast—demonstrates that the new algorithm consistently outperforms the traditional Gini index.

Manzali et al. [107] propose Enhancing Weak Nodes in Decision Trees (EWNDT), an approach designed to improve decision tree efficiency by addressing weak nodes using data augmentation. The algorithm augments data associated with weak nodes by artificially grouping neighboring nodes and re-evaluating candidate splits, thereby increasing model robustness. Experiments on UCI repository datasets show that EWNDT outperforms conventional pruning-based techniques.

Agarwal et al. [108] propose Hierarchical Shrinkage (HS), a post-processing regularization technique for decision trees and random forests. HS shrinks predictions toward the mean of ancestor nodes through a single regularization parameter while preserving the tree structure. Experiments across varied datasets demonstrate that HS significantly improves predictive performance and interpretability. The authors also link their method to ridge regression applied to decision stumps, providing theoretical grounding.

Arabameri et al. [81] assess decision tree-based ensemble machine learning algorithms for landslide susceptibility mapping in the Taleghan–Alamut Basin, Iran. The study employs Credal Decision Tree (CDT) models integrated with Bagging, Multiboost, and Subspace algorithms using eighteen conditioning factors. Model performance is evaluated using ROC curves and k-fold cross-validation. Results show that CDT-Multiboost achieves the highest performance, with an average AUC of 0.993, indicating strong feasibility for generating reliable landslide-susceptibility maps to mitigate future hazards.

X. Methodology

This study follows a structured literature-review methodology rather than an original empirical design. The aim is to synthesize recent research on impurity measures used in decision-tree learning and to compare how the reviewed studies report performance, robustness, and interpretability.

The review methodology is organized into three phases:

- **Literature identification:** Relevant studies published between 2020 and 2025 were collected from the literature cited in this study, with emphasis on work addressing classical impurity measures, generalized entropy-based criteria, hybrid split functions, pruning strategies, feature selection, and ensemble-based decision-tree optimization.
- **Screening and categorization:** The selected studies were grouped according to their primary contribution, including impurity-function design, performance optimization, overfitting control, interpretability, and application domain. This step enabled consistent comparison across methods and research themes.
- **Comparative synthesis:** Findings were synthesized qualitatively by comparing the reported outcomes of the reviewed papers, including accuracy trends, robustness to class imbalance and noise, model complexity, and computational considerations. No new experiments were conducted in this study.

This methodology aligns the study with its actual scope as a survey and provides a transparent basis for the discussion and comparison presented in later sections.

XI. Discussion

The reviewed literature shows a clear evolution in the role of impurity functions in decision-tree construction. While classical approaches such as the Gini index and Shannon entropy remain dominant for their simplicity and computational efficiency, they exhibit significant limitations when applied to modern datasets characterized by imbalance, high dimensionality, noise, and non-linear interactions. As documented throughout the study, these drawbacks manifest through biased splits, deeper and less interpretable trees, and an increased tendency toward overfitting—an issue exacerbated by the greedy, locally optimal nature of standard tree-building algorithms.

Recent studies explore alternative impurity measures—such as Rényi entropy, Tsallis entropy, cost-sensitive error functions, MECC, and ETC-based measures—to more accurately quantify uncertainty in multi-class, imbalanced, or temporally sensitive datasets. These functions provide improved modeling flexibility by introducing tunable parameters, non-linear weighting schemes, or domain-specific structural priors. Importantly, emerging work demonstrates that combining impurity measures (for example, linear mixtures of Gini and entropy) can outperform individual functions for specific classes of problems.

Beyond impurity functions themselves, the literature underscores the importance of complementary techniques such as pruning, ensemble learning, feature selection, and error-based optimization to mitigate overfitting and excessive tree growth. Studies show that when decision trees employ improved impurity functions in conjunction with these techniques, the resulting models consistently outperform baseline CART and C4.5 variants in terms of accuracy, robustness, computational cost, and interpretability.

XII. Results and Comparison

This section summarizes the findings reported in the reviewed literature rather than presenting original experimental results. The comparison focuses on four recurring dimensions discussed across the surveyed studies:

- **Predictive performance:** Reported changes in accuracy, precision, recall, and F1-score.
- **Model complexity:** Tree depth, number of nodes, and resulting interpretability.
- **Robustness and overfitting:** Sensitivity to noise, class imbalance, and training-data variation.
- **Computational efficiency:** Reported training cost and scalability with larger datasets.

As summarized in Table I, the reviewed studies generally indicate that adaptive impurity measures and impurity-aware optimization strategies can improve decision tree behavior relative to classical split criteria, especially when combined with pruning, feature selection, or ensemble learning. These findings should be interpreted as a synthesis of published evidence, not as new benchmark results produced by the present study.

XIII. Conclusion

This literature survey demonstrates that impurity functions play a foundational role in shaping the performance, interpretability, and computational properties of decision-tree models. Traditional impurity measures, while effective for structured and balanced datasets, often fall short in modern applications characterized by complexity, heterogeneity, and noise. A new generation of impurity functions—incorporating tunable parameters, non-linear weighting, temporal sensitivity, or structural complexity guarantees—has shown substantial empirical and theoretical promise.

TABLE I. CONDENSED SUMMARY OF REVIEWED DECISION TREE RESEARCH (2020–2025)

ID	Authors / Year	Method / Algorithm	Key Contributions	Datasets	Results
1	Mao (2025)	RSEIM / RS-Entropy	Generalized entropy-based DT with tunable splitting	8 UCI	Higher accuracy vs CART/C4.5
2	He (2025)	Dynamic Bayesian Splits	Posterior-probability split optimization	UCI	Faster + more robust RF
3	Ferreira (2025)	C4.5 Clinical DT	Mobility prediction in trauma patients	Clinical	Better diagnostic insights
4	Ciftcioglu (2025)	DT + Lévy Flight	Optimization for nonlinear engineering data	Concrete beams	Improved RMSE, MAE, R^2
5	Li (2024)	Modified C4.5 DSS	Faster DT for vending machine DSS	DSS data	Higher accuracy, reduced runtime
6	Delise (2024)	Era-aware GBDT	Robust OOD generalization under distribution shifts	Numerai	Strong OOD performance
7	Marton (2024)	GradTree	Full-tree gradient-descent optimization	Benchmarks	Higher accuracy vs greedy DT
8	Zeng (2024)	Impurity Theory	Clarifies impurity properties; beneficial mixtures	German Credit	Mixed measures outperform baseline
9	Loreti (2024)	Parallel DTs (Ray)	Scalable parallelization for explainability	GLEAMS	Faster DT building
10	Finke (2024)	BDTs for Weak Supervision	Noise-robust anomaly detection	LHC	Better than deep nets
11	Baena-Rojas (2023)	Hybrid MCDM Tree	IMS ranking for SMEs	18 SMEs	Identified US, NL, KR as best markets
12	KVG (2023)	PerformanceX EDM	Attribute selection + DT	Student data	99.41% accuracy
13	Bouke-Bukagini (2023)	BukaGini	Interaction-aware impurity	UNSW, UCI	+0.32–2.50% accuracy
14	Pathan (2023)	Improved Gini DT	Missing-data robust preprocessing	UCI	Higher precision/recall
15	Subramani (2023)	EnhancedTree+	Improved splitting + pruning	Titanic	Beats DT/RF/GBDT
16	Asif (2023)	Ensemble Health Trees	Extra Trees + preprocessing	Kaggle heart	98.15% accuracy
17	Cui (2023)	Robust GBDT	L1/L2 regularized GBDT	Real + synthetic	More robust predictions
18	Custode (2023)	EVO + Q-Learning Trees	Interpretable RL trees	RL benchmarks	Competitive with deep RL
19	Riansyah (2023)	C5.0 + AdaBoost	Boosting for imbalance	Hepatitis	Accuracy ↑ from 80.6→83%
20	Bodin (2022)	Max-Cut DT + PCA	Class-separating splits + node PCA	CIFAR-100	49% better accuracy, 94% faster
21	Horváth (2022)	Robust Stumps	DP-based exact robustness certification	Mixed datasets	Better certified robustness
22	Li (2022)	GMMLDT	Metric-learning-based DT splitting	Benchmarks	Up to 10× speedup
23	Lee (2022)	Distance-based Gini	Rao–Stirling distances in impurity	UCI	Higher accuracy
24	Manzali (2022)	EWNDT	Weak-node data augmentation	UCI	Superior to pruning
25	Arabameri (2022)	CDT + Multiboost	Uncertainty-aware hazard mapping	Landslides	AUC = 0.993

Moreover, the synergy between enhanced impurity functions and complementary techniques such as pruning, ensemble learning, feature selection, and hybrid modeling emerges as a critical factor in achieving high-performing and robust tree-based models. These developments indicate a decisive shift toward impurity-adaptive and context-aware decision-tree learning frameworks.

This study thus underscores the importance of rethinking impurity measurement in decision-tree design to meet the demands of increasingly challenging data environments and motivates future work toward systematically designing, analyzing, and benchmarking customized impurity functions.

ACKNOWLEDGMENT

The project was funded by the KAU Endowment (WAQF) at the King Abdulaziz University, Jeddah, Saudi Arabia. The authors, therefore, acknowledge with thanks WAQF and the Deanship of Scientific Research (DSR) for their technical and financial support.

REFERENCES

[1] M. J. Kochenderfer, T. A. Wheeler, and K. H. Wray, *Algorithms for decision making*. The MIT Press.

[2] G. Pappalardo, S. Cafiso, A. Di Graziano, and A. Severino, "Decision tree method to analyze the performance of lane support systems," vol. 13, no. 2, p. 846, number: 2 Publisher: Multidisciplinary Digital Publishing Institute. [Online]. Available: https://www.mdpi.com/2071-1050/13/2/846

[3] T. Zhang, H. Wang, J. Chen, and E. He, "Detecting unfavorable driving states in electroencephalography based on a PCA sample entropy feature and multiple classification algorithms," vol. 22, no. 11, p. 1248. [Online]. Available: https://www.mdpi.com/1099-4300/22/11/1248

[4] E. Scornet, "Trees, forests, and impurity-based variable importance." [Online]. Available: http://arxiv.org/abs/2001.04295

[5] W. Liu, H. Fan, and M. Xia, "Credit scoring based on tree-enhanced gradient boosting decision trees," vol. 189, p. 116034. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0957417421013798

[6] M. A. Carreira-Perpinan and A. Zharmagambetov, "Ensembles of bagged TAO trees consistently improve over random forests, AdaBoost and gradient boosting," in *Proceedings of the 2020 ACM-IMS on Foundations of Data Science Conference*. ACM, pp. 35–46. [Online]. Available: https://dl.acm.org/doi/10.1145/3412815.3416882

[7] A. Zharmagambetov, S. S. Hada, M. Gabidolla, and M. A. Carreira-Perpinan, "Non-greedy algorithms for decision tree optimization: An experimental comparison," in *2021 International Joint Conference on Neural Networks (IJCNN)*. IEEE, pp. 1–8. [Online]. Available: https://ieeexplore.ieee.org/document/9533597/

[8] T. Thomas, A. P. Vijayaraghavan, and S. Emmanuel, *Applications of Decision Trees*. Springer Singapore, pp. 157–184. [Online]. Available: http://link.springer.com/10.1007/978-981-15-1706-8_9

[9] B. Charbuty and A. Abdulazeez, "Classification based on decision tree algorithm for machine learning," vol. 2, no. 1, pp. 20–28. [Online]. Available: https://www.jastt.org/index.php/jasttpath/article/view/65

[10] X. Yang, Y. Wang, R. Byrne, G. Schneider, and S. Yang, "Concepts of artificial intelligence for computer-assisted drug discovery," vol. 119, no. 18, pp. 10 520–10 594. [Online]. Available: https://pubs.acs.org/doi/10.1021/acs.chemrev.8b00728

[11] S. N. H. Bukhari, J. Webber, and A. Mehbodniya, "Decision tree based ensemble machine learning model for the prediction of zika virus t-cell epitopes as potential vaccine candidates," vol. 12, no. 1, p. 7810. [Online]. Available: https://www.nature.com/articles/s41598-022-11731-6

[12] R. Rivera-Lopez, J. Canul-Reich, E. Mezura-Montes, and M. A. Cruz-Chávez, "Induction of decision trees as classification models through metaheuristics," vol. 69, p. 101006. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S2210650221001681

[13] X. Han, X. Zhu, W. Pedrycz, and Z. Li, "A three-way classification with fuzzy decision trees," vol. 132, p. 109788. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S1568494622008377

[14] W. Wegier, M. Koziarski, and M. Wozniak, "Multicriteria classifier ensemble learning for imbalanced data," vol. 10, pp. 16 807–16 818. [Online]. Available: https://ieeexplore.ieee.org/document/9706443/

[15] S. Tangirala, "Evaluating the impact of GINI index and information gain on classification using decision tree classifier algorithm*," vol. 11, no. 2. [Online]. Available: http://thesai.org/Publications/ViewPaper?Volume=11&Issue=2&Code=IJACSA&SerialNo=77

[16] G. Zhang and A. Gionis, "Regularized impurity reduction: accurate decision trees with complexity guarantees," vol. 37, no. 1, pp. 434–475. [Online]. Available: https://link.springer.com/10.1007/s10618-022-00884-7

[17] R. A. Disha and S. Waheed, "Performance analysis of machine learning models for intrusion detection system using gini impurity-based weighted random forest (GIWRF) feature selection technique," vol. 5, no. 1, p. 1. [Online]. Available: https://doi.org/10.1186/s42400-021-00103-8

[18] M. A. Bouke, A. Abdullah, J. Frnda, K. Cengiz, and B. Salah, "BukaGini: A stability-aware gini index feature selection algorithm for robust model performance," vol. 11, pp. 59 386–59 396. [Online]. Available: https://ieeexplore.ieee.org/document/10147829/

[19] Y. Zhang, B. Nie, J. Du, J. Chen, Y. Du, H. Jin, X. Zheng, X. Chen, and Z. Miao, "Feature selection based on neighborhood rough sets and gini index," vol. 9, p. e1711. [Online]. Available: https://peerj.com/articles/cs-1711

[20] G. Zeng, "On impurity functions in decision trees," pp. 1–19. [Online]. Available: https://www.tandfonline.com/doi/full/10.1080/03610926.2024.2317359

[21] E. J. Sturm, M. R. Carbone, D. Lu, A. Weichselbaum, and R. M. Konik, "Predicting impurity spectral functions using machine learning," vol. 103, no. 24, p. 245118, publisher: American Physical Society. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevB.103.245118

[22] X. Zhao and X. Nie, "Splitting choice and computational complexity analysis of decision trees," vol. 23, no. 10, p. 1241. [Online]. Available: https://www.mdpi.com/1099-4300/23/10/1241

[23] M. Przybyła-Kasperek and S. Aning, "Stop criterion in building decision trees with bagging method for dispersed data," vol. 192, pp. 3560–3569. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S1877050921018688

[24] T.-T. Huynh-Cam, L.-S. Chen, and H. Le, "Using decision trees and random forest algorithms to predict and determine factors contributing to first-year university students' learning performance," vol. 14, no. 11, p. 318. [Online]. Available: https://www.mdpi.com/1999-4893/14/11/318

[25] I. H. Sarker, "Data science and analytics: An overview from data-driven smart computing, decision-making and applications perspective," vol. 2, no. 5, p. 377. [Online]. Available: https://doi.org/10.1007/s42979-021-00765-8

[26] X. Jiang, "Online english teaching course score analysis based on decision tree mining algorithm," vol. 2021, pp. 1–10. [Online]. Available: https://www.hindawi.com/journals/complexity/2021/5577167/

[27] O. Sagi and L. Rokach, "Explainable decision forest: Transforming a decision forest into an interpretable tree," vol. 61, pp. 124–138. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S1566253519307869

[28] M. Bansal, A. Goyal, and A. Choudhary, "A comparative analysis of k-nearest neighbor, genetic, support vector machine, decision tree, and long short term memory algorithms in machine learning," vol. 3, p. 100071. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S2772662222000261

[29] M. M. Ghiasi, S. Zendehboudi, and A. A. Mohsenipour, "Decision tree-based diagnosis of coronary artery disease: CART model," vol. 192, p. 105400. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0169260719308971

[30] S. M. D. A. C. Jayatilake and G. U. Ganegoda, "Involvement of machine learning tools in healthcare decision making," vol. 2021, no. 1, p. 6679512. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1155/2021/6679512

[31] A. N. Elmachtoub, J. C. N. Liang, and R. McNellis,

"Decision trees for decision-making under the predict-then-optimize framework."

[32] A. Kim and H. Kim, "A new classification tree method with interaction detection capability," vol. 165, p. 107324. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167947321001584

[33] S. Lee, C. Lee, K. G. Mun, and D. Kim, "Decision tree algorithm considering distances between classes," vol. 10, pp. 69 750–69 756. [Online]. Available: https://ieeexplore.ieee.org/document/9810212/

[34] X. Xie, M.-J. Yuan, X. Bai, W. Gao, and Z.-H. Zhou, "On the gini-impurity preservation for privacy random forests." [Online]. Available: https://openreview.net/forum?id=d47iuwOt3j&referrer=%5Bthe%20profile%20of%20Wei%20Gao%5D(%2Fprofile%3Fid%3D~Wei_Gao7)

[35] M. Aria, C. Cuccurullo, and A. Gnasso, "A comparison among interpretative proposals for random forests," vol. 6, p. 100094. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2666827021000475

[36] J. Velasco-Mata, V. Gonzalez-Castro, E. F. Fernandez, and E. Alegre, "Efficient detection of botnet traffic by features selection and decision trees," vol. 9, pp. 120 567–120 579. [Online]. Available: https://ieeexplore.ieee.org/document/9523853/

[37] R. Dunne, R. Reguant, P. Ramarao-Milne, P. Szul, L. M. Sng, M. Lundberg, N. A. Twine, and D. C. Bauer, "Thresholding gini variable importance with a single-trained random forest: An empirical bayes approach," vol. 21, pp. 4354–4360. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S2001037023003082

[38] S. Yu, F. Alesiani, X. Yu, R. Jenssen, and J. C. Principe, "Measuring dependence with matrix-based entropy functional." [Online]. Available: http://arxiv.org/abs/2101.10160

[39] K. Gajowniczek and T. Ząbkowski, "ImbTreeEntropy: An r package for building entropy-based classification trees on imbalanced datasets," vol. 16, p. 100841. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S2352711021001242

[40] K. Uma and S. A. Alias Balamurugan, "C5.0 decision tree model using tsallis entropy and association function for general and medical dataset," pp. –1–1. [Online]. Available: http://autosoftjournal.net/paperShow.php?paper=100000153

[41] V. Ignatenko, A. Surkov, and S. Koltcov, "Random forests with parametric entropy-based information gains for classification and regression problems," vol. 10, p. e1775. [Online]. Available: https://peerj.com/articles/cs-1775

[42] P. Yu and Y. Deng, "On the scaled rényi entropy and application," vol. 0, no. 0, pp. 1–0, publisher: Taylor & Francis _eprint: https://doi.org/10.1080/03610926.2024.2301986. [Online]. Available: https://doi.org/10.1080/03610926.2024.2301986

[43] K. Gajowniczek and T. Ząbkowski, "Interactive decision tree learning and decision rule extraction based on the ImbTreeEntropy and ImbTreeAUC packages," vol. 9, no. 7, p. 1107. [Online]. Available: https://www.mdpi.com/2227-9717/9/7/1107

[44] R. P. Bonidia, A. P. Avila Santos, B. L. S. De Almeida, P. F. Stadler, U. Nunes Da Rocha, D. S. Sanches, and A. C. P. L. F. De Carvalho, "Information theory for biological sequence classification: A novel feature extraction technique based on tsallis entropy," vol. 24, no. 10, p. 1398. [Online]. Available: https://www.mdpi.com/1099-4300/24/10/1398

[45] C. Kumar, G. Podestá, K. Kilpatrick, and P. Minnett, "A machine learning approach to estimating the error in satellite sea surface temperature retrievals," vol. 255, p. 112227. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0034425720306003

[46] A. Hamadani, N. A. Ganai, S. Mudasir, S. Shanaz, S. Alam, and I. Hussain, "Comparison of artificial intelligence algorithms and their ranking for the prediction of genetic merit in sheep," vol. 12, no. 1, p. 18726. [Online]. Available: https://www.nature.com/articles/s41598-022-23499-w

[47] C. J. Nzekwe, S. Kim, and S. A. Mostafa, "Interaction selection and prediction performance in high-dimensional data: A comparative study of statistical and tree-based methods," pp. 259–279. [Online]. Available: https://jds-online.org/doi/10.6339/24-JDS1127

[48] Y. Wu, Y. Ke, Z. Chen, S. Liang, H. Zhao, and H. Hong, "Application of alternating decision tree with AdaBoost and bagging ensembles for landslide susceptibility mapping," vol. 187, p. 104396. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0341816219305387

[49] S. Mishra, P. K. Mallick, H. K. Tripathy, A. K. Bhoi, and A. González-Briones, "Performance evaluation of a proposed machine learning model for chronic disease datasets using an integrated attribute evaluator and an improved decision tree classifier," vol. 10, no. 22, p. 8137. [Online]. Available: https://www.mdpi.com/2076-3417/10/22/8137

[50] P. Yariyan, S. Janizadeh, T. Van Phong, H. D. Nguyen, R. Costache, H. Van Le, B. T. Pham, B. Pradhan, and J. P. Tiefenbacher, "Improvement of best first decision trees using bagging and dagging ensembles for flood probability mapping," vol. 34, no. 9, pp. 3037–3053. [Online]. Available: https://doi.org/10.1007/s11269-020-02603-7

[51] B. Mahbooba, M. Timilsina, R. Sahal, and M. Serrano, "Explainable artificial intelligence (XAI) to enhance trust management in intrusion detection systems using decision tree model," vol. 2021, pp. 1–11. [Online]. Available: https://www.hindawi.com/journals/complexity/2021/6634811/

[52] V. Aceña, I. Martín De Diego, R. R. Fernández, and J. M. Moguerza, "Minimally overfitted learners: A general framework for ensemble learning," vol. 254, p. 109669. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0950705122008450

[53] Z. Azam, M. M. Islam, and M. N. Huda, "Comparative analysis of intrusion detection systems and machine learning-based model analysis through decision tree," vol. 11, pp. 80 348–80 391. [Online]. Available: https://ieeexplore.ieee.org/document/10185955/

[54] W. Lian, G. Nie, B. Jia, D. Shi, Q. Fan, and Y. Liang, "An intrusion detection method based on

decision tree-recursive feature elimination in ensemble learning," vol. 2020, pp. 1–15. [Online]. Available: https://www.hindawi.com/journals/mpe/2020/2835023/

[55] X. Luo, X. Wen, M. Zhou, A. Abusorrah, and L. Huang, "Decision-tree-initialized dendritic neuron model for fast and accurate data classification," vol. 33, no. 9, pp. 4173–4183. [Online]. Available: https://ieeexplore.ieee.org/document/9380661/

[56] M. Fernando, F. Cèsar, N. David, and H. José, "Missing the missing values: The ugly duckling of fairness in machine learning," vol. 36, no. 7, pp. 3217–3258. [Online]. Available: https://onlinelibrary.wiley.com/doi/10.1002/int.22415

[57] S. Nijman, A. Leeuwenberg, I. Beekers, I. Verkouter, J. Jacobs, M. Bots, F. Asselbergs, K. Moons, and T. Debray, "Missing data is poorly handled and reported in prediction model studies using machine learning: a literature review," vol. 142, pp. 218–229. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0895435621003759

[58] Z. He, Z. Wu, G. Xu, Y. Liu, and Q. Zou, "Decision tree for sequences," pp. 1–1. [Online]. Available: https://ieeexplore.ieee.org/document/9411677/

[59] A. Simonetto, E. Dall'Anese, S. Paternain, G. Leus, and G. B. Giannakis, "Time-varying convex optimization: Time-structured algorithms and applications," publisher: arXivVersion Number: 1. [Online]. Available: https://arxiv.org/abs/2006.08500

[60] J. Hatwell, M. M. Gaber, and R. M. A. Azad, "CHIRPS: Explaining random forest classification," vol. 53, no. 8, pp. 5747–5788. [Online]. Available: https://link.springer.com/10.1007/s10462-020-09833-6

[61] V. Blanco, A. Japón, and J. Puerto, "Robust optimal classification trees under noisy labels," vol. 16, no. 1, pp. 155–179. [Online]. Available: https://link.springer.com/10.1007/s11634-021-00467-2

[62] S. B. Jabeur, A. Sadaaoui, A. Sghaier, and R. Aloui, "Machine learning models and cost-sensitive decision trees for bond rating prediction," vol. 71, no. 8, pp. 1161–1179. [Online]. Available: https://www.tandfonline.com/doi/full/10.1080/01605682.2019.1581405

[63] E. García-Martín, N. Lavesson, H. Grahn, E. Casalicchio, and V. Boeva, "Energy-aware very fast decision tree," vol. 11, no. 2, pp. 105–126. [Online]. Available: https://link.springer.com/10.1007/s41060-021-00246-4

[64] R. Lasfar and G. Tóth, "The difference of model robustness assessment using cross-validation and bootstrap methods," vol. 38, no. 6, p. e3530. [Online]. Available: https://analyticalsciencejournals.onlinelibrary.wiley.com/doi/10.1002/cem.3530

[65] N. Syam and R. Kaul, *Random Forest, Bagging, and Boosting of Decision Trees*. Emerald Publishing Limited, pp. 139–182. [Online]. Available: https://www.emerald.com/insight/content/doi/10.1108/978-1-80043-880-420211006/full/html

[66] V. A. S. Hernández, R. Monroy, M. A. Medina-Pérez, O. Loyola-González, and F. Herrera, "A practical tutorial for decision tree induction: Evaluation measures for candidate splits and opportunities," vol. 54, no. 1, pp. 1–38. [Online]. Available: https://dl.acm.org/doi/10.1145/3429739

[67] S. Bashir, I. U. Khattak, A. Khan, F. H. Khan, A. Gani, and M. Shiraz, "A novel feature selection method for classification of medical data using filters, wrappers, and embedded approaches," vol. 2022, pp. 1–12. [Online]. Available: https://www.hindawi.com/journals/complexity/2022/8190814/

[68] N. Singh and P. Singh, "A hybrid ensemble-filter wrapper feature selection approach for medical data classification," vol. 217, p. 104396. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0169743921001647

[69] A. Nugroho, A. Z. Fanani, and G. F. Shidik, "Evaluation of feature selection using wrapper for numeric dataset with random forest algorithm," in *2021 International Seminar on Application for Technology of Information and Communication (iSemantic)*. IEEE, pp. 179–183. [Online]. Available: https://ieeexplore.ieee.org/document/9573249/

[70] C. Wu and W. Li, "Enhancing intrusion detection with feature selection and neural network," vol. 36, no. 7, pp. 3087–3105. [Online]. Available: https://onlinelibrary.wiley.com/doi/10.1002/int.22397

[71] R.-C. Chen, C. Dewi, S.-W. Huang, and R. E. Caraka, "Selecting critical features for data classification based on machine learning methods," vol. 7, no. 1, p. 52. [Online]. Available: https://journalofbigdata.springeropen.com/articles/10.1186/s40537-020-00327-4

[72] Z. Liang, C. Wang, and K. U. J. Khan, "Application and comparison of different ensemble learning machines combining with a novel sampling strategy for shallow landslide susceptibility mapping," vol. 35, no. 6, pp. 1243–1256. [Online]. Available: https://link.springer.com/10.1007/s00477-020-01893-y

[73] D. Arpit, H. Wang, Y. Zhou, and C. Xiong, "Ensemble of averages: Improving model selection and boosting performance in domain generalization," version Number: 4. [Online]. Available: https://arxiv.org/abs/2110.10832

[74] Z. Sun, C. Wang, Y. Zhao, and C. Yan, "Multi-label ECG signal classification based on ensemble classifier," vol. 8, pp. 117 986–117 996. [Online]. Available: https://ieeexplore.ieee.org/document/9125941/

[75] J. Abdollahi and B. Nouri-Moghaddam, "A hybrid method for heart disease diagnosis utilizing feature selection based ensemble classifier model generation," vol. 5, no. 3, pp. 229–246. [Online]. Available: https://link.springer.com/10.1007/s42044-022-00104-x

[76] T. Zeng, L. Wu, D. Peduto, T. Glade, Y. S. Hayakawa, and K. Yin, "Ensemble learning framework for landslide susceptibility mapping: Different basic classifier and ensemble strategy," vol. 14, no. 6, p. 101645. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S1674987123001123

[77] P. Pintelas and I. E. Livieris, "Special issue on ensemble learning and applications," vol. 13, no. 6, p. 140. [Online]. Available: https://www.mdpi.com/1999-4893/13/6/140

[78] I. D. Mienye and Y. Sun, "A survey of ensemble learning: Concepts, algorithms, applications, and prospects," vol. 10, pp. 99 129–99 149. [Online]. Available: https://ieeexplore.ieee.org/document/9893798/

[79] C. Chen, Y. Tsai, F. Chang, and W. Lin, "Ensemble

feature selection in medical datasets: Combining filter, wrapper, and embedded feature selection results," vol. 37, no. 5, p. e12553. [Online]. Available: https://onlinelibrary.wiley.com/doi/10.1111/exsy.12553

[80] G. Tuysuzoglu and D. Birant, "Enhanced bagging (eBagging): A novel approach for ensemble learning," vol. 17, no. 4, pp. 515–528. [Online]. Available: https://iajit.org/PDF/July%202020,%20No.%204/17698.pdf

[81] A. Arabameri, S. Chandra Pal, F. Rezaie, R. Chakrabortty, A. Saha, T. Blaschke, M. Di Napoli, O. Ghorbanzadeh, and P. T. Thi Ngo, "Decision tree based ensemble machine learning approaches for landslide susceptibility mapping," vol. 37, no. 16, pp. 4594–4627. [Online]. Available: https://www.tandfonline.com/doi/full/10.1080/10106049.2021.1892210

[82] R. Sibindi, R. W. Mwangi, and A. G. Waititu, "A boosting ensemble learning based hybrid light gradient boosting machine and extreme gradient boosting model for predicting house prices," *Engineering Reports*, vol. 5, no. 4, p. e12599, Apr. 2023. [Online]. Available: https://onlinelibrary.wiley.com/doi/10.1002/eng2.12599

[83] R. Mao, X. Shi, and Z. Shi, "A Decision Tree Classification Algorithm Based on Two-Term RS-Entropy," *Entropy*, vol. 27, no. 10, p. 1069, Oct. 2025. [Online]. Available: https://www.mdpi.com/1099-4300/27/10/1069

[84] J. He, Z. Li, and L. Yin, "An Efficient and Accurate Random Forest Node-Splitting Algorithm Based on Dynamic Bayesian Methods," *Machine Learning and Knowledge Extraction*, vol. 7, no. 3, p. 70, Jul. 2025. [Online]. Available: https://www.mdpi.com/2504-4990/7/3/70

[85] R. C. Ferreira, K. Dunn-Lopez, S. Moorhead, A. Krupp, B. V. Zuchatti, L. A. C. Carvalho, M. T. D. S. L. Botelho, and E. C. M. Duran, "Decision Trees for Managing Impaired Physical Mobility in Multiple Trauma Patients," *Journal of Advanced Nursing*, p. jan.17010, May 2025. [Online]. Available: https://onlinelibrary.wiley.com/doi/10.1111/jan.17010

[86] A. O. Ciftcioglu, A. Delikanlı, T. Shafighfard, and F. Bagherzadeh, "Machine learning based shear strength prediction in reinforced concrete beams using Levy flight enhanced decision trees," *Scientific Reports*, vol. 15, no. 1, p. 27488, Jul. 2025. [Online]. Available: https://www.nature.com/articles/s41598-025-12359-y

[87] P. Li, F. Xiong, X. Huang, and X. Wen, "Construction and optimization of vending machine decision support system based on improved c4.5 decision tree," vol. 10, no. 3, p. e25024. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S2405844024010557

[88] T. DeLise, "Era splitting – invariant learning for decision trees." [Online]. Available: http://arxiv.org/abs/2309.14496

[89] S. Marton, S. Lüdtke, C. Bartelt, and H. Stuckenschmidt, "GradTree: Learning axis-aligned decision trees with gradient descent." [Online]. Available: http://arxiv.org/abs/2305.03515

[90] R. C. Arbiv, L. Lovat, A. Rosenfeld, and D. Sarne, "Optimizing decision trees for enhanced human comprehension," in *Artificial Intelligence. ECAI 2023 International Workshops*, S. Nowaczyk, P. Biecek, N. C. Chung, M. Vallati, P. Skruch, J. Jaworek-Korjakowska, S. Parkinson, A. Nikitas, M. Atzmüller, T. Kliegr, U. Schmid, S. Bobek, N. Lavrac, M. Peeters, R. Van Dierendonck, S. Robben, E. Mercier-Laurent, G. Kayakutlu, M. L. Owoc, K. Mason, A. Wahid, P. Bruno, F. Calimeri, F. Cauteruccio, G. Terracina, D. Wolter, J. L. Leidner, M. Kohlhase, and V. Dimitrova, Eds. Springer Nature Switzerland, vol. 1947, pp. 366–381, series Title: Communications in Computer and Information Science. [Online]. Available: https://link.springer.com/10.1007/978-3-031-50396-2_21

[91] D. Loreti and G. Visani, "Parallel approaches for a decision tree-based explainability algorithm," vol. 158, pp. 308–322. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0167739X24001857

[92] T. Finke, M. Hein, G. Kasieczka, M. Krämer, A. Mück, P. Prangchaikul, T. Quadfasel, D. Shih, and M. Sommerhalder, "Tree-based algorithms for weakly supervised anomaly detection," vol. 109, no. 3, p. 034033. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevD.109.034033

[93] J. J. Baena-Rojas, T. M. Mackenzie-Torres, G. Cuesta-Giraldo, and A. Tabares, "A HYBRID MULTI-CRITERIA DECISION-MAKING TECHNIQUE FOR INTERNATIONAL MARKET SELECTION IN SMES," vol. 27, no. 1, pp. 26–45. [Online]. Available: https://pjms.zim.pcz.pl/resources/html/article/details?id=611202

[94] U. J. Ujwal and S. Malik, "A hybrid weight based feature selection algorithm for predicting students' academic advancement by employing data science approaches," vol. 13, no. 5, pp. 1–22. [Online]. Available: https://www.mecs-press.org/ijeme/ijeme-v13-n5/v13n5-1.html

[95] S. Pathan and S. K. Sharma, "Design an optimal decision tree based algorithm to improve model prediction performance," vol. 11, no. 6, pp. 127–133. [Online]. Available: https://ijritcc.org/index.php/ijritcc/article/view/7295

[96] M. K. Subramani and M. G. Muruganantharaj, "EnhancedTree+: A novel approach for improving decision tree classifiers," vol. 11, no. 7, pp. 208–216.

[97] D. Asif, M. Bibi, M. S. Arif, and A. Mukheimer, "Enhancing heart disease prediction through ensemble learning techniques with hyperparameter optimization," vol. 16, no. 6, p. 308. [Online]. Available: https://www.mdpi.com/1999-4893/16/6/308

[98] S. Cui, A. Sudjianto, A. Zhang, and R. Li, "Enhancing robustness of gradient-boosted decision trees through one-hot encoding and regularization." [Online]. Available: http://arxiv.org/abs/2304.13761

[99] L. L. Custode and G. Iacca, "Evolutionary learning of interpretable decision trees," vol. 11, pp. 6169–6184. [Online]. Available: https://ieeexplore.ieee.org/document/10015004/

[100] M. Riansyah, S. Suwilo, and M. Zarlis, "Improved accuracy in data mining decision tree classification using adaptive boosting (adaboost)," vol. 8, no. 2, pp. 617–622. [Online]. Available: https://jurnal.polgan.ac.id/index.php/sinkron/article/view/12055

[101] H. N. B, A. Jain, and N. Nagaraj. Permutation decision trees. [Online]. Available: https://arxiv.org/abs/2306.02617v3

[102] M. Awad and S. Fraihat, "Recursive feature elimination with cross-validation with decision tree: Feature selection method for machine learning-based intrusion detection systems," vol. 12, no. 5, p. 67. [Online]. Available: https://www.mdpi.com/2224-2708/12/5/67

[103] I. Roshanski, M. Kalech, and L. Rokach, "Automatic feature engineering for learning compact decision trees," *Expert Syst. Appl.*, vol. 229, no. PA, Nov. 2023. [Online]. Available: https://doi.org/10.1016/j.eswa.2023.120470

[104] J. Bodine and D. S. Hochbaum, "A better decision tree: The max-cut decision tree with modified PCA improves accuracy and running time," vol. 3, no. 4, p. 313. [Online]. Available: https://link.springer.com/10.1007/s42979-022-01147-4

[105] M. Z. Horváth, M. N. Müller, M. Fischer, and M. Vechev, "(de-)randomized smoothing for decision stump ensembles," 2022. [Online]. Available: https://arxiv.org/abs/2205.13909

[106] D. Li and S. Chen, "A novel splitting criterion inspired by geometric mean metric learning for decision tree," 2022. [Online]. Available: https://arxiv.org/abs/2204.11011

[107] Y. Manzali, M. El Far, M. Chahhou, and M. Elmohajir, "Enhancing weak nodes in decision tree algorithm using data augmentation," *Cybernetics and Information Technologies*, vol. 22, pp. 50–65, 06 2022.

[108] A. Agarwal, Y. S. Tan, O. Ronen, C. Singh, and B. Yu, "Hierarchical shrinkage: improving the accuracy and interpretability of tree-based methods," *CoRR*, vol. abs/2202.00858, 2022. [Online]. Available: https://arxiv.org/abs/2202.00858