Journal of the Brazilian Computer Society, 2025, 31:1, doi: 10.5753/jbcs.2025.5324

A Review of Interpretability Methods for Gradient Boosting Decision Trees

Victoria de Sousa Figueira Gonçalves [**University of Sao Paulo** | ***vsfgoncalvs@gmail.com***]
Vinicius Renan de Carvalho [**University of Sao Paulo** | ***vrcarvalho@usp.br***]

Escola Politécnica, University of Sao Paulo (USP), Av. Luciano Gualberto, 158, São Paulo, SP, 05508-010, Brazil

Received: 13 December 2024 • **Accepted:** 12 June 2025 • **Published:** 18 August 2025

Abstract This survey examines interpretability methods used or proposed for Gradient Boosting Decision Trees, which are advanced machine learning algorithms based on decision trees. The studies analyzed were gathered using synonyms for ”explainability” combined with synonyms for ”method,” as well as synonyms for ”Gradient Boosting Decision Trees.” The proposed or applied approaches are classified by their techniques and described in detail. Among these methods, we recommend using SHAP values to rank features based on their relevance, as this approach aligns well with the structure of Gradient Boosting Decision Trees. Additionally, we suggest considering inTrees, RULECOSI+, and Tree Space Prototypes when applicable.

Keywords: Explainability, Interpretability, Gradient Boosting Decision Trees, Machine Learning

1 Introduction

As machine learning (ML) techniques have evolved into complex algorithms, such as ensemble models that combine multiple simple machine learning models, understanding how these models work has become increasingly challenging. The ability to comprehend the decision-making process of an ML model is referred to as interpretability or explainability. This brings us to the performance-interpretability trade-off [Alex Goldstein and Pitkin, 2015] [Adadi and Berrada, 2018] [Das *et al.*, 2020] [Barredo Arrieta *et al.*, 2020] [Minh *et al.*, 2022] [Nadeem *et al.*, 2023] [Nagahisarchoghaei *et al.*, 2023], which suggests that, generally, the more complex the model, the better the expected performance and the higher difficulty of understanding its process.

To overcome this challenge, a range of methods that aim to delve into the decision-making process of ML models have been developed, reducing the performance-interpretability trade-off [Adadi and Berrada, 2018] [Barredo Arrieta *et al.*, 2020] [Minh *et al.*, 2022]. Despite the agreement on the need to understand the reasons for an algorithm's decision and the model's mechanism, there is not a consensus on the appropriate term or definition to describe this. Several studies use the terms “interpretability” and “explainability” interchangeably. Besides, the methods and processes used to comprehend ML models can be called Explainable Artificial Intelligence (XAI) or Interpretable Artificial Intelligence (IAI).

It is worth noting that the increasing attention to the Machine Learning Operations (MLOps) approach, which aims to automate ML processes, including training, monitoring, and deploying in production ML models, requires the use of interpretability methods to maintain the processes in a responsible line. Therefore, features related to interpretability may improve a company's competitiveness in many fields that use AI.

Additionally, regarding formal institutions, the European Parliament has adopted the General Data Protection Regulation (GDPR), which includes clauses on automated decision-making. This regulation introduces the right to receive “meaningful explanations of the logic involved” in such decisions [Guidotti *et al.*, 2018]. Moreover, the International Financial Reporting Standards (IFRS) require the use of interpretable models, ensuring that credit decisions can be adequately explained and justified to both customers and regulators [De Bock *et al.*, 2023].

According to Barredo Arrieta *et al.* [2020], decision tree ensembles are arguably among the most accurate ML models in use nowadays. In addition, Yasodhara *et al.* [2021] note that tree ensemble models are widely employed in research and industry due to their good performance. Specifically, Gradient Boosting Decision Trees (GBDTs), which are sequential ensembles of decision trees, generally have a high prediction accuracy [Ma *et al.*, 2023b]. However, these models are not inherently understandable since their mechanisms of decision-making are complex [Minh *et al.*, 2022] [Nagahisarchoghaei *et al.*, 2023] [Rawal *et al.*, 2022]. Interestingly, Minh *et al.* [2022] note the prevalence of studies on the interpretability of bagging techniques until 2018, when a shift occurred to other ensemble methods, such as stacking and boosting algorithms.

This review examines the available interpretability methods for Gradient Boosted Decision Trees (GBDTs) and organizes them into categories. The aim is to streamline the search for suitable techniques and summarize the current landscape of interpretability methods, while also recommending the adoption of certain approaches. To achieve this, we categorize the methods, analyze their underlying mechanisms, and investigate how they relate to the structure of GBDTs. A total of 70 studies were reviewed, including 23 articles focusing on the explainability of machine learning models, 39 studies that utilized interpretability methods with GBDTs, and 8 studies that introduced new explainability methods.

In Section 2, we examine the relevant concepts related to

Gradient Boosting Decision Trees and their interpretability. Section 3 analyzes surveys that focus on the interpretability of machine learning models. In Section 4, we outline the methodology used, which includes the selection of studies and the categorization of methods. Section 5 reviews the methods proposed or applied within the studies analyzed. Finally, in Section 6, we present our remarks, and in Section 7, we share our overall impressions.

2 Background

2.1 Gradient Boosting Decision Trees

A decision tree is a supervised machine learning model that maps input features to one or more outcome variables of interest. Its goal is to minimize a loss function, which describes how the model measures the difference between the predicted values and the actual values.

Decision trees consist of a series of "yes or no" questions based on specific cutoff thresholds. Each split in the tree aims to minimize a defined criterion for the algorithm. If the inequality condition is satisfied, the observation proceeds to the next step on the left; otherwise, it moves to the right.

When a step in the tree is the final one, it is referred to as a leaf; if there is a subsequent split, it is called a node. Each leaf is associated with a value, which is the output for the observations that fall into that leaf. Therefore, the decisions made in the root node, which is the starting point, all the way to a leaf describe how a particular decision was reached. This sequence of decisions is known as a path. Additionally, the length of the longest path from the root node to a leaf node is known as the depth of the decision tree.

Mathematically, each node Q_i is achieved from a set of decisions s_j:

$$Q_i(\theta) = \bigcap_{j=0}^{i} s_j(\theta) \tag{1}$$

where θ is a set of parameters that describes a decision tree.

Consequently, the tree is a predictor function $h(x)$ from a family of functions $h(x;\theta)$, with the goal of minimizing a criterion at each node.

Decision trees are simple models that can be combined to form a unique model using bagging or boosting techniques, creating ensemble algorithms. In bagging algorithms, multiple models are trained simultaneously, while boosting involves a sequential training process.

Bagging algorithms [Breiman, 1996] operate by using voting or averaging to combine results from different models. Each model is trained on a sample of the training data, and the outcomes of these models are then aggregated, as illustrated in **Figure 1**.

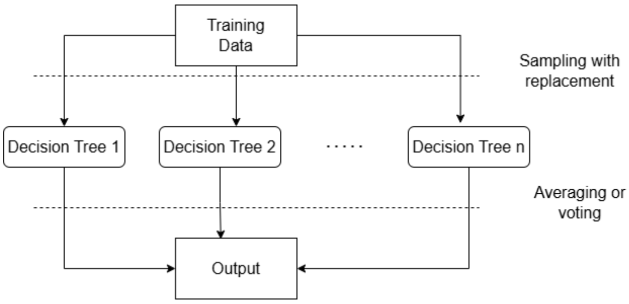

Figure 1. Bagging Decision Tree

In contrast, boosting algorithms [Schapire, 1990] focus on improving the performance of the preceding model in the sequence. Each subsequent model is designed to reduce the errors made by the previous one by using pseudo-residuals in the response variable (i.e., actual values to be predicted). Pseudo-residuals are calculated as the differences between the actual values to be predicted and the values predicted by the prior decision tree. Finally, the predicted values are aggregated, as illustrated in **Figure 2**.

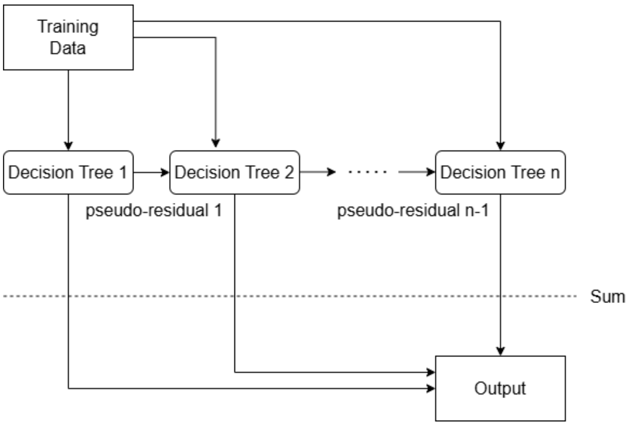

Figure 2. Boosting Decision Tree

Specifically, in Gradient Boosted Decision Trees (GBDTs), each subsequent decision tree is intended to improve upon the performance of the trained decision tree before it. Thus, the first step in the GBDT process can be described as follows:

$$F_0 = \arg\min_{\hat{y}} \sum_{i=1}^{n} \mathcal{L}(y_i, \hat{y}) \tag{2}$$

where $\mathcal{L}(y_i, \hat{y})$ is the loss function, $\hat{y}$ is the predicted value for the leaf and y_i is the true value.

Since the subsequent node aims to improve this result, the next step is to calculate the pseudo residuals r_{im}, where m denotes the order of the decision trees, being $h_m(x)$ the decision tree created based on these residuals:

$$r_{im} = -\mathcal{L}(y_i, F_{m-1}(x_i)) \tag{3}$$

Then, the output from the m decision tree is:

$$\hat{y} = \arg\min_{\hat{y}} \sum_{i=1}^{n} \mathcal{L}(y_i, \hat{y} + \gamma h_m(x_i)) \tag{4}$$

Where γ is the learning rate, which is a positive number that indicates how much the GBDT weighs the previous residuals.

Based on the analyzed studies, we notice that certain variations of GBDTs are frequently utilized. This subsection details the specifics of these models.

2.1.1 AdaBoost

The Adaptive Boosting Algorithm (Adaboost) [Freund and Schapire, 1997] consists of a boosting algorithm that adjusts adaptively to the errors of the weak learning algorithm trained. In these models, a parameter β is updated according to the error of the previous output, ε, so that the larger the pseudo-residual (ε), the larger the weight of the output for the next decision tree (β).

2.1.2 XGBoost

EXtreme Gradient Boosting (XGBoost) [Chen and Guestrin, 2016] introduces a regularization to the minimization problem. Instead of simply weighting the previous errors, it penalizes the complexity of the model (α) and smooths the final learned weights to avoid over-fitting (λ):

$$\mathcal{L}(\phi) = \sum_i l(\hat{y}_i, y_i) + \sum_k \Omega(f_k) \tag{5}$$

where $\Omega(f) = \alpha T + \frac{1}{2}\lambda \parallel w \parallel^2$, T is the number of leaves and w the leaf weights.

2.1.3 LightGBM

LightGBM [Ke *et al.*, 2017] introduces two techniques, Gradient-based One Side Sampling (GOSS) and Exclusive Feature Bundling (EFB), to deal with a large number of data instances and a large number of features, respectively.

GOSS retains all instances with a significant absolute gradient (i.e., residual) value, while performing random sampling on those with smaller gradients, which are assigned a constant weight. This approach prioritizes the instances that are under-trained.

EFB involves combining exclusive features—features that never take nonzero values simultaneously—into a single feature. This bundling can significantly reduce processing time in a sparse feature space.

2.1.4 CatBoost

CatBoost, which stands for Categorical Boosting [Prokhorenkova *et al.*, 2019], implements ordered boosting and is capable of handling categorical data through an ordering principle. To illustrate how the ordered boosting is implemented, assume that a random permutation σ of the training examples is sampled and n supporting models are maintained ($M_1, ..., M_n$), such that the model M_i is learned using the first i examples in the permutation. At each step, in order to obtain the residual for the j-th sample, we use the M_{j-1} model. We then have n models.

To handle categorical data, the algorithm uses Target Statistics (TS), which involves replacing a category x^i_k, which is the category k of the feature i by a numeric feature equal to a TS $\hat{x}^i_k$, based on organizing the data sequentially in time. If time is not available, a random permutation is used.

2.2 Interpretability

Understanding the decision-making process of complex models like GBDT is challenging, prompting the development of various methods to gain insights into this area [Adadi and Berrada, 2018] [Barredo Arrieta *et al.*, 2020] [Minh *et al.*, 2022]. Despite the agreement on the need to understand the reasons for an algorithm's decision and the model's mechanism; there is not a consensus on the appropriate term or definition to describe this. Several studies use the terms "interpretability" and "explainability" interchangeably. Besides, the methods and processes used to comprehend ML models can be called Explainable Artificial Intelligence (XAI) or Interpretable Artificial Intelligence (IAI).

Das *et al.* [2020] define IML and interpretability, respectively, as a way of exploring an ML model aiming to provide a more reasoned explanation of the predictions and the degree to which a human can understand the reason behind decisions made by an ML model.

Barredo Arrieta *et al.* [2020] criticize the interchangeable use of interpretability and explainability. According to them, interpretability expresses a passive characteristic of a model, referring to the level at which a model makes sense for a human, while explainability is an interface between humans and a decision maker, being, ergo, an accurate proxy of the decision maker and comprehensible to humans. Similarly, Rawal *et al.* [2022] state that the explanation of the system must go beyond the interpretation since interpretability is the ability of the AI system to be explained in human terms, while explainability is a set of processes or methods that ensures that the system allows humans to comprehend its overall decision and reasoning. Additionally, Burkart and Huber [2021] states that interpretability usually refers to the comprehension of how the model works as a whole, whereas explainability often refers to explanations surrounding prediction models that are incomprehensible themselves.

On the contrary, Nagahisarchoghaei *et al.* [2023] understand explainability as more concerned with explaining the inner mechanism of the model and interpretability as focused on understanding the cause of a decision.

In the present study, interpretability and explainability are used interchangeably, consisting of techniques that aim to enhance the comprehension surrounding the decision-making process of an ML model. Consequently, as exposed by Nagahisarchoghaei *et al.* [2023], XAI's aims to make a black-box model more transparent or use a transparent model to help end-users better understand model behaviors.

Moreover, Rawal *et al.* [2022] conclude that explainability can aid in avoiding biases that cause unethical and harmful consequences, and Das *et al.* [2020] remark that it has multiple benefits, helping users to extract interpretable patterns from trained ML models, identify the reasons behind poor predictions, increase trust in model predictions, detect bias in ML models, and protect against overfitting. They also note that legal requirements and ethics might compel the use of interpretability.

It is worth noting that, as observed by Adadi and Berrada [2018], the need for explainability depends on the degree of functional opacity caused by the complexity of the ML model and the degree of resistance of the domain to errors. Therefore, when the functional opacity is high and/or the cost of wrong decisions is high, the need for interpretability is high. We could compare the need to understand the logic behind a decision-making process in medical diagnosis and a social media recommendation; clearly, the cost of wrong decisions in the medical field (e.g., a false negative for the prediction of a disease) is higher than in social media (e.g., showing a post that does not fit the user's preferences). Nonetheless, explainability tools are desired in both cases to increase trust in the decisions.

To better understand machine learning (ML) models, various methods can be applied to analyze their decision-making process. These methods typically rank the features based on specific metrics. Explanations for ML models can be broadly categorized into two types: global explanations and local explanations. Global explanations provide insights into the model as a whole, while local explanations focus on specific predictions made by the model.

Furthermore, explanation tools can be classified as either model-agnostic or model-specific. Model-agnostic methods are versatile and can be used to explain a wide range of models, whereas model-specific methods are designed for particular types of models [Alex Goldstein and Pitkin, 2015] [Rawal *et al.*, 2022] [Islam *et al.*, 2021] [Das *et al.*, 2020] [Carvalho *et al.*, 2019] [Nagahisarchoghaei *et al.*, 2023]. As a result, model-specific methods are intrinsically linked to the underlying mechanisms of the model itself.

As the field of interpretability in machine learning has evolved, several reviews have been conducted to explore its concepts and available methods. These surveys typically focus on specific concepts, fields, or methods. Since the present study concentrates on interpretability for GBDT, the related surveys help us understand the existing concepts and the methods that have already been explored.

3 Surveys available in the literature

Various surveys focus on machine learning explainability, highlighting its importance and examining various fields and methods. Most surveys emphasize concepts by analyzing terms such as interpretability and explainability and categorizing interpretability methods.

Interestingly, Rawal *et al.* [2022] mentioned that decision trees and ensemble methods are not transparent models. Therefore, it is necessary to use interpretability methods when developing them. However, some techniques successfully explain certain models and perform poorly when explaining other models, which is referred to as the challenge of transferability. Minh *et al.* [2022] also states that tree ensembles require model simplification or feature relevance techniques and mentions the prevalence of studies on bagging techniques until 2018, when a shift occurred to other ensemble methods such as stacking and boosting algorithms.

In the study by Guidotti *et al.* [2018], an approach is considered generalizable when a purely reverse engineering procedure is followed; thus, the black box is only queried with different input records, while a method is called random if a random perturbation or permutation of the original dataset is used. Moreover, Islam *et al.* [2021] classifies whether the method approximates the model's behavior or not and whether the method alone is inherently interpretable or not.

In the study by Stepin *et al.* [2021], the focus is on counterfactual and contrastive methods. It is stated that contrastive explanations highlight the difference between the actual and a hypothetical decision, while counterfactual explanations specify the minimal changes needed in the input to obtain a contrastive output. Moreover, they note that there are explanations with both characteristics, known as contrastive-counterfactual explanations, that it is common to use the terms "contrastive" and "counterfactual" interchangeably. de Oliveira and Martens [2021] also examine counterfactual explanations, with a focus on tabular data, and discuss how these methods concentrate on determining how features can be modified to change the output classification.

Covert *et al.* [2022] analyze methods that simulate feature removal to quantify the influence of each feature. These methods, known as removal-based explanations, could be categorized as feature relevance based on the major categories used by general reviews.

Burkart and Huber [2021] focus on explainability for supervised machine learning models. The definition from Doshi-Velez and Kim [2017] is adopted, which states that explainability is required when incompleteness exists, meaning that there is something about the problem that cannot be sufficiently encoded into the model.

Yasodhara *et al.* [2021], global feature importance methods (Gain and SHAP) are analyzed, with a focus on explanations for tree ensemble models, as these methods are widely used in industry. Perturbations in both the model and dataset are created to evaluate how the methods perform in terms of accuracy and stability using Random Forest (RF), Gradient Boosting Machines, and XGBoost.

Li *et al.* [2023b], the concept of eXplainable Anomaly Detection (XAD) is explored in the context of anomaly detection. XAD is defined as the extraction of relevant knowledge from an anomaly detection model regarding the relationships within the data or learned by the model. The study highlights the usefulness of XGBoost as a surrogate model for anomaly detection, emphasizing the need for an explainability method as well.

Başağaoğlu *et al.* [2022] focus on IAI and XAI models for tree-based ensemble hydroclimatic factor predictors. The models considered consisted of XGBoost, LightGBM, CatBoot, Extremely Randomized Trees, and RFs.

Bharati *et al.* [2023] concentrate on explainability in healthcare, emphasizing its criticality due to the significance of diagnostics in saving human lives. On the other hand, Rjoub *et al.* [2023] and Nadeem *et al.* [2023] analyze XAI for cybersecurity. While the former focuses on use cases and categorization in the cybersecurity field, the latter examines different stakeholders and their objectives. Similarly, Moustafa *et al.* [2023] analyzes explainability in the context of intrusion detection, noting it is limited to analyzing feature relevance and the correlation of security events.

3.1 Surveys Analysis

A total of 9 surveys analyzed cover the broader concept of explainability in machine learning ([Das *et al.*, 2020] [Rawal *et al.*, 2022] [Barredo Arrieta *et al.*, 2020] [Minh *et al.*, 2022] [Adadi and Berrada, 2018] [Islam *et al.*, 2021] [Carvalho *et al.*, 2019] [Guidotti *et al.*, 2018] [Nagahisarchoghaei *et al.*, 2023]). Additionally, 8 surveys focus on interpretability within specific fields ([Li *et al.*, 2023b], [Başağaoğlu *et al.*, 2022], [Bharati *et al.*, 2023], [Kök *et al.*, 2023], [De Bock *et al.*, 2023], [Rjoub *et al.*, 2023], [Nadeem *et al.*, 2023],[Moustafa *et al.*, 2023]). Furthermore, 6 surveys examine interpretability concerning specific machine learning methods or explainability techniques ([Stepin *et al.*, 2021] [de Oliveira and Martens, 2021] [Covert *et al.*, 2022] [Sahakyan *et al.*, 2021] [Burkart and Huber, 2021] [Yasodhara *et al.*, 2021]) examine interpretability in the context of specific ML methods or specific explainability methods. The surveys that focus on specific fields can be found in **Table 1**, while those that concentrate on specific methods are presented in **Table 2**.

Table 1. Specific Fields Reviews

Ref.	Field
Li *et al.* [2023b]	Anomaly Detection
Başağaoğlu *et al.* [2022]	Ecology
Bharati *et al.* [2023]	Healthcare
Kök *et al.* [2023]	IOT
De Bock *et al.* [2023]	Operational Research
Rjoub *et al.* [2023]	Cybersecurity
Nadeem *et al.* [2023]	Cybersecurity
Moustafa *et al.* [2023]	Cybersecurity

It is noteworthy that while there is a prevalence of surveys focused on cybersecurity, the specific methods analyzed are quite diverse, despite the limited number of specific reviews.

This survey concentrates on the interpretability of Gradient Boosted Decision Trees (GBDT), examining explainability techniques applied or proposed in studies and how these methods relate to the mechanisms of GBDT. We have not identified a review with this specific objective. Therefore, our aim is to address the challenge of transferability mentioned by Rawal *et al.* [2022] by conceptually connecting the ML model and the concept of explainability in the method. Likewise, we have not found a survey on explainability for GBDT; it is worth citing Yasodhara *et al.* [2021], which analyzes Gain and SHAP methods for Tree Ensembles.

4 Systematical Analysis

The goal of this analysis is to examine studies that either propose or apply methods of explainability for GBDTs, relating the techniques to the GBDT's structure. To identify relevant techniques from the articles and reviews, we conducted a search using the terms "interpretability" or "explainability" in conjunction with the terms "method", "tool", "algorithm", "technique" or "approach" as well as at least one of the terms "gradient boosting trees", "gradient boosted trees", "gradient boosting decision trees" and "gradient boosted decision trees". This search was conducted in IEEE Xplore, ACM, and Scopus, taking into account the available studies as of May 2024.

Table 3 presents the total amount of research gathered for each tool, including any duplicates that may exist between them.

The string used for each research tool is indicated below.

IEEE Xplore

((“Full Text & Metadata”:“interpretability method”) OR (“Full Text & Metadata”:“explainability method”) OR (“Full Text & Metadata”:“interpretability tool”) OR (“Full Text & Metadata”:“explainability tool”) OR (“Full Text & Metadata”:“interpretability algorithm”) OR (“Full Text & Metadata”:“explainability algorithm”) OR (“Full Text & Metadata”:“interpretability technique”) OR (“Full Text & Metadata”:“explainability technique”) OR (“Full Text & Metadata”:“interpretability approach”) OR (“Full Text & Metadata”:“explainability approach”)) AND ((“Full Text & Metadata”:“gradient boosting trees”) OR (“Full Text & Metadata”:“gradient boosted trees”) OR (“Full Text & Metadata”:“gradient boosting decision trees”) OR (“Full Text & Metadata”:“gradient boosted decision trees”))

ACM

[[All: “interpretability method”] OR [All: “explainability method”] OR [All: “interpretability tool”] OR [All: “explainability tool”] OR [All: “interpretability algorithm”] OR [All: “explainability algorithm”] OR [All: “interpretability technique”] OR [All: “explainability technique”] OR [All: “interpretability approach”] OR [All: “explainability approach”]] AND [[All: “gradient boosting trees”] OR [All: “gradient boosted trees”] OR [All: “gradient boosting decision trees”] OR [All: “gradient boosted decision trees”]]

Scopus

(ALL (“ interpretability method”) OR ALL (“explainability method”) OR ALL (“interpretability tool”) OR ALL (“explainability tool”) OR ALL (“interpretability algorithm”) OR ALL (“explainability algorithm”) OR ALL (“interpretability technique”) OR ALL (“explainability technique”) OR ALL (“interpretability approach”) OR ALL (“explainability approach”)) AND (ALL (“gradient boosting trees”) OR ALL (“gradient boosted trees”) OR ALL (“gradient boosting decision trees”) OR ALL (“gradient boosted decision trees”))

Synonyms of "method" and "gradient boosting decision trees" were used. The words "interpretability" and "explainability" were not used in isolation because they would refer to studies surrounding the difficulties and definitions rather than the methods available.

After performing a search, we sifted through reviews that discuss interpretability techniques and articles that mention the use of an explainability method and GBDT or propose a method applicable to GBDTs by reading their abstracts. We specifically excluded studies that use GBDT in a more complex architecture without focusing on GBDT explainability. Additionally, we assessed studies referenced in the reviews and applied the filter based on abstract reading to them as well. In total, we analyzed 70 accessible studies from this approach. The process is described in **Figure 3**, according to systematic reviews and meta-analyses (PRISMA) standards.

Table 2. Specific Methods Reviews

Ref.	Field
Stepin *et al.* [2021]	Contractual and Contrastive Methods
de Oliveira and Martens [2021]	Counterfactual Methods for Tabular Data
Covert *et al.* [2022]	Removal-based explanations
Sahakyan *et al.* [2021]	Tabular Data
Burkart and Huber [2021]	Supervised Models
Yasodhara *et al.* [2021]	Global Explanations for Tree Ensembles

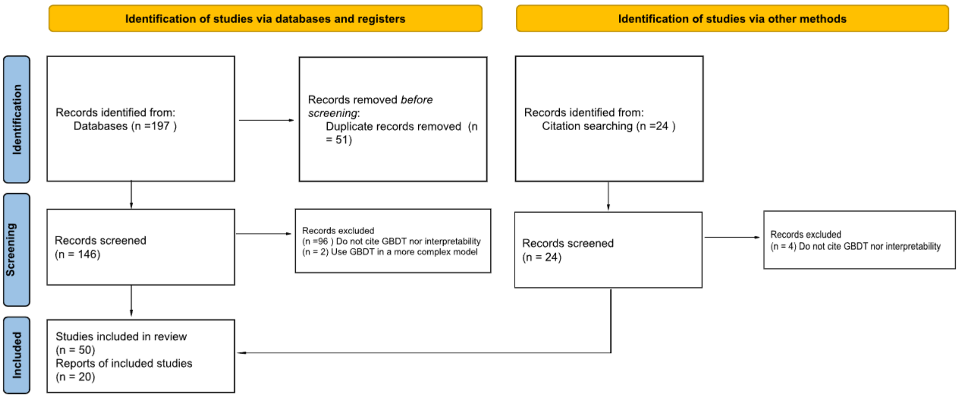
Identification of studies via databases and registers
Identification of studies via other methods
Identification
Screening
Included
Records identified from:
Databases (n =197)
Records removed *before screening*:
Duplicate records removed (n = 51)
Records identified from:
Citation searching (n =24)
Records screened
(n = 146)
Records excluded
(n =96) Do not cite GBDT nor interpretability
(n = 2) Use GBDT in a more complex model
Records screened
(n = 24)
Records excluded
(n = 4) Do not cite GBDT nor interpretability
Studies included in review
(n = 50)
Reports of included studies
(n = 20)

Figure 3. PRISMA

Table 3. Quantity of studies per tool

IEE Xplorer	ACM	Scopus
24	23	150

Out of the total studies, 23 are surveys, as described in Section 3. Additionally, 39 studies use various methods, while 8 studies propose new methods. In Section 5, we analyze the methods presented or applied.

The interpretability methods analyzed are classified based on their techniques and can be categorized as either model-agnostic or model-specific and as global or local, as shown in **Table 4**. Furthermore, we indicate whether each method has been adopted in studies other than the one in which it was originally proposed. Below, we provide a brief description of each category of technique.

Feature Importance It defines the importance of the feature as the use of a mathematical concept to rank the features in the model, defining the relevance of the feature. This can be achieved by either directly extracting or computing internal metrics.

These metrics can be aggregated or calculated for different values of the features. On the one hand, aggregated feature importance metrics express the general relevance of the feature. On the other hand, the metrics that can be analyzed for different values of the feature or other features allow relationship analysis.

Simplification These techniques involve simplifying the original model by either approximating the original model with an explainable surrogate model or extracting the nodes decisions.

Contrastive Explanations As explained by Stepin *et al.* [2021], contrastive explanations highlight the difference between the actual and a hypothetical decision, while counterfactual explanations specify the minimal changes needed in the input to obtain a contrastive output. However, the terms are often used interchangeably; therefore, this category includes both.

Prototypes Using the definition of prototypes as representative points of data, we can explain a prediction by presenting similar, representative points.

5 Studies

The methods of explainability applied in the studies analyzed are briefly described in this section and classified in **Table 4**. The studies employing a method and their respective domains and techniques used are detailed in **Table 5**.

5.1 Feature Importance

Gain Proposed by [Quinlan, 1993], this method refers to the difference in a specific metric resulting from a particular split, usually calculated based on accuracy. It is typically averaged across all splits where the feature is utilized, but it can also be considered as the total gain from all splits involving that feature.

For instance, Ho *et al.* [2022] analyze feature importance in two XGBoost models by using two interpretations of gain, one based on the reduction in the loss function and the other derived from the Gini coefficient. Nevertheless, the authors use the terms gain and ELI5 when referencing gain according to the loss function and the Gini coefficient, respectively.

Table 4. Methods of Explainability

Method	Global or Local	Agnostic or Specific	Classification	Applied
Gain	Global	Specific	Feature Importance	Yes
Split	Global	Specific	Feature Importance	Yes
Coverage	Global	Specific	Feature Importance	Yes
Prediction Value Change	Global	Specific	Feature Importance	Yes
SHAP	Both	Both	Feature Importance	Yes
PDP	Global	Agnostic	Feature Importance	Yes
ICE	Both	Agnostic	Feature Importance	Yes
ALE	Global	Agnostic	Feature Importance	Yes
Permutation Feature Importance	Global	Agnostic	Feature Importance	Yes
Surrogate Model	Both	Agnostic	Simplification	Yes
LIME	Local	Agnostic	Simplification	Yes
Contrastive Explanations	Local	Agnostic	Contrastive Explanations	Yes
Decision Contribution	Global	Agnostic	Simplification	No
Tree Space Prototype	Local	Specific	Prototypes	No
inTrees	Global	Specific	Simplification	No
RULECOSI+	Global	Specific	Simplification	No
Bayesian Model Selection	Global	Specific	Simplification	No

Using Python functions' names instead of concepts suggests a focus on tools rather than definitions, leading to confusing conclusions.

Split It quantifies how frequently a feature is used to split (i.e., the number of nodes involving the variable) the data in the model, making it a measure of its involvement in the decision-making process of the ML model. However, this metric does not account for the magnitude of a feature's contribution; for instance, a feature may appear in numerous nodes that contribute only marginally to the differentiation of outcomes. Conversely, a feature used in fewer nodes may exert a more substantial influence if those splits lead to significant changes in output.

In their study, Calderón-Díaz *et al*. [2024] conducted a split-based feature importance analysis on an XGBoost model aimed at predicting muscle injuries in professional soccer players. While the authors cite the use of a Python library for this purpose, they do not describe the methodological details, effectively defaulting to the library's standard behavior, which relies on split count.

Coverage It represents the average percentage of instances that meet the condition across all splits where the feature is utilized. Therefore, such as for Split, these nodes might contribute only marginally to the differentiation of outcomes.

Prediction Values Change It expresses how much, on average, the prediction changes if the feature value changes.

Tran *et al*. [2021] apply Prediction Values Change for both feature importance per se and feature selection. As a result, only the most relevant features are retained in the final model, helping to reduce the risk of overfitting. This approach illustrates the versatility of interpretability methods.

SHAP SHapley Additive exPlanations (SHAP)[Lundberg and Lee, 2017] method is based on cooperative game theory, where each feature acts as an agent that collaborates with others to achieve a better outcome. This approach examines the interactions between features and their combined impact on the final result, considering all possible feature combinations. In game theory, participants consider the potential actions of others when deciding how to act, allowing us to analyze hypothetical yet plausible scenarios. The outcome for each participant depends on the actions of everyone involved.

This concept aligns well with Gradient Boosting Decision Trees (GBDT), as well as other models based on decision trees, as sequential games can be represented through decision trees. Each tree aims to improve upon the performance of its predecessor. Additionally, features located in earlier nodes generally have higher SHAP values, reinforcing the idea that decisions made earlier significantly influence the final decision.

Since SHAP is an additive feature attribution method, each observation can be represented as a linear combination of feature importance. A feature's importance value is calculated as the weighted average of the differences between the outcomes from a trained model that includes it and the outcomes from a trained model that excludes it, considering all possible subsets of features.

SHAP value has several variations that are designed to better fit different machine learning models and optimize the SHAP algorithm for easier computation. For example, TreeSHAP was specifically developed for tree-based machine learning models [Lundberg *et al*., 2020]. As a result, we can view SHAP as both model-agnostic and model-specific. Additionally, the Shapley–Lorenz method combines the SHAP value with the Lorenz curve to generate a SHAP value based on the differential contribution to global predictive accuracy, rather than the differential contribution to each individual predicted value [Giudici and Raffinetti, 2021].

Finally, by using SHAP values, we can analyze multiple outputs. In local analysis, we can visualize the output as a result of the importance of each feature for the instance. From this local analysis, we can create global analyses by aggregating the SHAP values for each observation, typically calculating the average or the absolute values of the averages. Additionally, we can examine the interactions between features by plotting one feature's value alongside its SHAP value and the value of another feature Lundberg *et al.* [2019].

For example, Herrera *et al.* [2023] analyzed SHAP values at the level of individual observations and features, while also exploring the relationships among features, including their connections to the target variable. Their study explored outputs from an XGBoost model, a neural network, and two linear regression models. Notably, the SHAP value distributions differed substantially between the neural network and XGBoost models, highlighting how model architecture can significantly influence the interpretation of feature importance.

Similarly, Stojić *et al.* [2019] examine the importance of features of an XGBoost based on the mean absolute SHAP Values, Gain, Coverage, and Split. As expected, the outputs from these techniques are significantly different as a result of the concept's differences.

It is worth noting that most studies (32 out of 39) utilize SHAP, which aligns effectively with the mechanisms of Gradient Boosting Decision Trees (GBDTs). It is important to note that none of the studies explicitly mention that SHAP's rationale aligns with the structure of GBDTs.

Permutation Feature Importance In Permutation Feature Importance Fisher *et al.* [2019], the feature j is randomly shuffled for each repetition k in $1, ..., K$ to generate a corrupted version of the data. The importance of the feature is determined by calculating the difference between the original score s (usually the accuracy) and the averaged score for the corrupted data:

$$i_j = s - \frac{1}{K}\sum_{k=1}^{K} s_{k,j} \tag{6}$$

PDP Partial dependence plots (PDP) [Friedman, 2001] illustrate the relationship between predicted values and selected features of interest while averaging over the values of all other features. Mathematically, consider a set of features X_S from the vector of features X_T and let X_C be the complement. When $f(x_s, x_C)$ is not differentiable, as for GBDTs, it is calculated as:

$$pd_{X_S}(x_S) = \frac{1}{N}\sum_{i=1}^{N} f(x_S, x_C^{(i)}) \tag{7}$$

Therefore, it describes how one feature influences the output while considering the impact of other features. It overlooks the possibility that some hypothetical values of those other features might not be feasible for a specific instance.

Rhee *et al.* [2020] examine four types of models: AdaBoost, Decision Trees, RFs, Extremely Randomized Trees, and multiple linear regression. These models are designed to detect hydrological droughts in ungauged areas across different time scales. The Permutation Feature Importances are normalized so that the total importance scores of all variables sum to one hundred, and PDP is applied.

ICE Individual Conditional Expectation (ICE) [Alex Goldstein and Pitkin, 2015] involves plotting the relationship between the predicted response and a specific feature for individual observations. This approach helps identify interactions and extrapolations in predictor space. Visually, ICE plots disaggregate the output of Partial Dependence Plots (PDPs). For each of the N observed and fixed values of x_C, a curve is plotted against the observed values of x_S. Each curve defines the conditional relationship between x_S and the predicted value at fixed values of x_C.

Using the centered version of ICE is recommended when we want to observe heterogeneity in the model, especially when the curves exhibit a wide range of intercepts. In this case, the predictions are centered around a selected location x^* within the range of x_S.

ALE Accumulated Local Effects (ALE) [Apley and Zhu, 2020] also illustrates the relationship between the predicted value and the features of interest. However, instead of averaging the predictions over the marginal distribution, ALE averages the predictions over the conditional distribution. This approach is taken due to the limitations of PDP for a feature that is strongly correlated with other features since, when $f(x)$ is not differentiable, the partial dependence plot consists of averaging predictions of artificial data instances.

The uncentered ALE for the feature j:

$$A\tilde{L}E_j(x) = \sum_{k=1}^{k_j(x)} \frac{1}{n_j(k)} \sum_{i:x_j^{(i)} \in N_j(k)} [f(x_{k,j}, x_C^{(i)}) - f(x_{k-1,j} x_C^{(i)})] \tag{8}$$

where $N_j(k)$ is a partition of the sample range of $\{x_{i,j} : i = 1, 2, ..., n\}$ into K intervals. We get the centered ALE by centering this value using its mean value.

Thus, it averages the changes in the predictions and accumulates them over a small interval. As for PDP, we can use heatmaps to analyze two features simultaneously.

Interestingly, Wu *et al.* [2023] apply PDP, ICE, and ALE to analyze feature importance plots for a GBDT that aims to predict the severity of COVID-19. Firstly, the most important features are recognized from the Permutation Feature Importance, and then PDP, ICE, and ALE are plotted for some features. It is worth noting that ALE is more suitable in healthcare since the features are generally highly correlated, whereas ICE allows the analysis of heterogeneity.

5.2 Simplification

Surrogate Model This process involves approximating the original model with an explainable surrogate model. In this approach, the outputs from the original model serve as the dependent variable for the explainable model, which uses the original model's features as inputs.

LIME Consists of the local approximation using an interpretable model Ribeiro *et al.* [2016]. Therefore, it does not relate directly to the GBDT's structure.

For each instance selected for analysis, LIME generates perturbed versions by uniformly sampling random data points in the vicinity of the original instance. The number of perturbed instances is also sampled uniformly. The predictions from the machine learning model are obtained for these perturbed instances, capturing the model's output within the local neighborhood of the original instance.

Finally, LIME assigns weights to the perturbed instances based on their proximity to the original instance and uses these weights to develop an interpretable model. The importance of each feature is then derived from this interpretable model.

Interestingly, Aljadani *et al.* [2023] propose using ML models alongside LIME for credit scoring. They emphasize the relevance of plotting the feature importance of each variable and analyzing the most impactful features in decision-making.

Decision Contribution In GBDTs, we calculate the absolute value of the difference between the current node's decision and the previous decision to determine the contribution of that decision [Delgado-Panadero *et al.*, 2022]. This output can be aggregated for either a decision space or a specific feature.

For a decision space analysis, we compute the intersection spaces defined by the splits from the nodes along the decision path. This process removes redundancy and consolidates the decision contributions within the defined interval.

For feature analysis, we assess how much the prediction changes after each decision related to the feature of interest and then sum all these decision contributions.

Decision Rules Since GBDTs' outputs come from intersections of conditions, we can extract decision rules consisting of conditions based on distinct features to better understand the model's mechanism.

inTrees A rule can be derived from the root node of a decision tree to a leaf node, describing the decision made at the leaf. This is denoted as $C \Rightarrow T$, where C is the condition (i.e., a conjunction of variable-value pairs defined by a threshold) and T is the output of the rule (i.e., the value in the leaf node). To extract the rules, we identify the conditions in each node from the root node to the node of interest. This leads to the following metrics:

- Frequency: The proportion of observations that satisfy the rule, serving as a measure of the rule's popularity.
- Error: The proportion of incorrectly classified instances that satisfy the rule or the mean squared error in regression problems.
- Length: The number of variable-value pairs in the condition, reflecting the complexity of the rule.

From these metrics, several analyses can be performed, such as reducing rules based on the decrease in error associated with each variable-value pair, selecting rules through feature selection, extracting variable interactions using association rule analysis, and aggregating the extracted rules into a rule-based learner known as a Simplified Tree Ensemble Learner (STEL). This framework is referred to as inTrees [Deng, 2014].

RULECOSI+ RULECOSI+, which stands for "Rule Combination and Simplification" [Obregon and Jung, 2023], is a method that extracts and selects decision rules based on the concepts of coverage (the fraction of observations that satisfy a rule) and accuracy. Initially, paths are defined in terms of conditions specified by nodes in a decision tree, forming rule sets. These pairs of rule sets are then combined and simplified, with overlapping intervals being streamlined.

Pairs that have an accuracy exceeding a specified threshold, denoted as α, are retained. A logit score is then generated from the decision rules, similar to GBDTs. The rules within each ruleset are organized in descending order according to their accuracy and coverage. Rules with accuracy greater than α are added to a pruned ruleset, and the data covered by those rules is excluded from a copy of the original dataset. This process is repeated with the remaining rules and data until no additional rules or data are left to test.

Ultimately, only the rules in a final ruleset that meet the following criteria are retained: coverage exceeding a specified threshold β, accuracy greater than α, and a significant impact on the error rate. To evaluate the last condition, the confidence interval of the error is considered, using a binomial distribution typically at a confidence level of 25%.

It is important to note that RULECOSI [Obregon *et al.*, 2019] is not directly explained, despite being proposed in an analyzed study, because it is the basis of RULECOSI+ [Obregon and Jung, 2023].

Bayesian Model Selection Approach We can also extract rules from ensembles of decision trees by using the concept of regions from geometrics and apply this to a Bayesian Model Selection Approach [Hara and Hayashi, 2017] to find an approximation of the model from the extracted rules. To achieve this, we simplify a rule, which is essentially the path from the root node to a specific node in a decision tree. We do this by analyzing the intervals for each feature and describing this simplification as a region in the space defined by the thresholds. In the case of tree ensembles, each observation falls within the intersection of multiple regions. Therefore, to enhance the interpretability of the tree ensemble, we can use a smaller number of regions to approximate the original model.

Since a condition imposed by a node corresponds to a binary outcome based on a defined region, we can model this as a Bernoulli distribution that reflects the probability of the features belonging to each interval. From this, we can derive the posterior distribution of y given the binary feature using Bayes' rule and estimate the model parameters and, if not pre-defined, the future number of regions.

Given that we may obtain multiple candidates, we can select the model that minimizes the error. For this selection process, we employ a Bayesian model selection algorithm known as factorized asymptotic Bayesian (FAB) inference.

5.3 Contrastive Explanations

It explicitly highlight the distinctions between an actual decision and a hypothetical one. This approach allows for a direct comparison between the actual output derived from specific inputs and an alternative scenario that uses different inputs, leading to a distinct outcome.

The only study analyzed that applies Contrastive Explanations, as well as a Surrogate Model, is from Aguilar-Palacios *et al*. [2020]. In this work, the authors generate contrastive explanations based on the outputs of a linear regression surrogate model, specifically in the context of sales forecasting.

5.4 Prototypes

Tree Space Prototype For Gradient Boosted Decision Trees (GBDTs), the distance functions consider both the predictions made by the model and how the model uses features to produce those outputs. Consequently, the prototypes derived from this process are referred to as “tree space prototypes” [Tan *et al*., 2020]. In GBDTs, the proximity between a pair of points is calculated as a weighted average based on the number of trees in the model where the points land in the same leaf, with the learning rate serving as the weight for each decision tree. To finalize this, a k-medoids problem is solved to minimize the sum of distances from each object to its nearest medoid (or prototype), with the distance defined as $1 - proximity$.

6 Discussion

The SHAP method is well-suited for analyzing feature importance because it aligns with the GBDT framework by definition. Despite its popularity (used in 32 out of 39 studies), the theoretical compatibility between SHAP and GBDT is not explicitly acknowledged in any of the studies.

Similarly, RuleCOSI+, the most recent interpretability method, is designed for the GBDT mechanism and can simplify models by utilizing concepts inherent to its structure, considering both the coverage and accuracy. However, this method was not employed in the studies analyzed.

Interestingly, the only methods specifically developed for decision trees or ensembles of decision trees that were utilized relied on calculations based directly on the existing splits within the trees. Although decision rules can clearly elucidate decisions by a model, inTrees allow multiple analyses surrounding a GBDT structure, and Tree Space Prototype methods define representative points of data; these approaches were not applied in the studies reviewed. Instead, the focus was largely on ranking features based on their impact on predicted values for both the overall model and specific instances. With the exception of the study by Aguilar-Palacios *et al*. [2020], the other research examined the importance of features from a global perspective, even though not all of them used a method specifically designed for this purpose.

Furthermore, 19 studies employed multiple interpretability approaches. This diversity is advantageous, as different methods, even when addressing the same concept of feature importance, utilize distinct metrics. As such, it is common for different interpretability methods assessing feature importance to produce varied rankings of the features. Understanding the fundamental mechanisms of these methods is essential. Additionally, when applying the same interpretability methods to different machine learning models using the same dataset, the results can vary significantly due to the differing decision-making processes involved.

7 Conclusion

We have analyzed 18 methods that can enhance the understandability of Gradient Boosting Decision Trees and categorized them by technique. Additionally, we reviewed 39 studies that implemented explainability methods in the context of Gradient Boosting Decision Trees.

We noticed the prevalence of the adoption of SHAP Value, which effectively matches the Gradient Boosting Decision Trees mechanisms. However, the choice of SHAP values does not appear to be a consequence of the structural matching. Furthermore, most studies utilize more than one interpretability method, which is advantageous. We also noted a tendency to focus on ranking feature importance rather than understanding specific decisions through the analysis of decision rules or prototypes.

We recommend using SHAP values to assess the importance of different features. Additionally, we suggest employing Decision Rules to clarify specific decisions made by the model. For overall model simplification, RULECOSI+ is a good choice, while inTrees can be utilized for detailed analysis of the model’s structure. Furthermore, Tree Space Prototypes can help visualize representative data points effectively.

We recommend that future studies focus on evaluating the methods used for explainability processing. This evaluation should consider both the available tools and their associated costs, including processing time. By taking these factors into account, we can achieve a realistic assessment of the costs and benefits while considering practical execution possibilities.

Declarations

Authors’ Contributions

Victoria de Sousa Figueira Gonçalves is the main writer and also responsible for the conceptualization, literature reviewing and editing. Vinicius Renan de Carvalho is responsible for the methodology, conceptualization, supervision, reviewing and editing. All authors read and approved the final manuscript.

Competing interests

The authors declare that they have no competing interests

Availability of data and materials

The datasets (and/or software) generated and/or analyzed during the current study will be made upon request.

Table 5. Studies that apply a method. PFI stands for Permutation Feature Importance, and PVC for Prediction Values Change

Ref.	Field	Methods
Herrera *et al*. [2023]	ICTs	SHAP
Christodoulou and Gregoriades [2023]	Language	SHAP
Shen *et al*. [2022]	Engineering	SHAP
Ma *et al*. [2023a]	Engineering	SHAP
Ma *et al*. [2023b]	Engineering	SHAP
Nguyen *et al*. [2023]	Finance	SHAP
Bedi *et al*. [2020]	Ecology	SHAP
Gramegna and Giudici [2020]	Finance	SHAP
Vaulet *et al*. [2022]	Healthcare	SHAP
Du *et al*. [2023]	Healthcare	SHAP
Lv *et al*. [2023]	Transport	SHAP
Tang *et al*. [2023]	Airport	SHAP
Boulitsakis Logothetis *et al*. [2023]	Healthcare	SHAP
He *et al*. [2023]	Ecology	SHAP
Feng *et al*. [2022]	Engineering	SHAP
Chu *et al*. [2024]	Healthcare	SHAP
Sokhansanj and Rosen [2022]	Language	SHAP, Gain
Li *et al*. [2023a]	Engineering	SHAP, Gain
Stojić *et al*. [2019]	Ecology	SHAP,Coverage, Split
Ho *et al*. [2022]	Chemistry	SHAP, Gain
GhoshRoy *et al*. [2023]	Healthcare	SHAP, LIME
Wang *et al*. [2023a]	Engineering	SHAP, ALE
Liang *et al*. [2022]	Engineering	SHAP, PDP
Lazaridis *et al*. [2023]	Engineering	SHAP, ALE, PDP
Wu *et al*. [2023]	Medicine	SHAP, PFI, PDP, ICE, ALE, LIME
Kookalani *et al*. [2022b]	Engineering	SHAP, ALE, PDP
Kookalani *et al*. [2022a]	Engineering	SHAP, ALE, PDP
Liu *et al*. [2024]	Finance	SHAP, PVC, Gain, Split
Settouti and Saidi [2024]	Healthcare	SHAP, PFI, LIME
Jas *et al*. [2024]	Ecology	SHAP, Split
Zhang *et al*. [2024]	Chemistry	SHAP, PDP
Wang *et al*. [2023b]	Healthcare	SHAP, PDP
Shimizu *et al*. [2022]	Healthcare	Gain
Tran *et al*. [2021]	Ecology	PVC
Calderón-Díaz *et al*. [2024]	Healthcare	Split
Aljadani *et al*. [2023]	Finance	LIME
Rhee *et al*. [2020]	Ecology	PFI
Aguilar-Palacios *et al*. [2020]	Sales	Surrogate Model, Contrastive Explanations
Shield and Houston [2022]	Ecology	PFI, ALE

References

Adadi, A. and Berrada, M. (2018). Peeking Inside the Black-Box: A Survey on Explainable Artificial Intelligence (XAI). *IEEE Access*, 6:52138–52160. DOI: 10.1109/ACCESS.2018.2870052.

Aguilar-Palacios, C., Munoz-Romero, S., and Rojo-Alvarez, J. L. (2020). Cold-Start Promotional Sales Forecasting through Gradient Boosted-Based Contrastive Explanations. *IEEE Access*, 8:137574–137586. DOI: 10.1109/ACCESS.2020.3012032.

Alex Goldstein, Adam Kapelner, J. B. and Pitkin, E. (2015). Peeking inside the black box: Visualizing statistical learning with plots of individual conditional expectation. *Journal of Computational and Graphical Statistics*, 24(1):44–65. DOI: 10.1080/10618600.2014.907095.

Aljadani, A., Alharthi, B., Farsi, M., Balaha, H., Badawy, M., and Elhosseini, M. (2023). Mathematical Modeling and Analysis of Credit Scoring Using the LIME Explainer: A Comprehensive Approach. *Mathematics*, 11(19). DOI: 10.3390/math11194055.

Apley, D. W. and Zhu, J. (2020). Visualizing the effects of predictor variables in black box supervised learning models. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 82(4):1059–1086. DOI: 10.1111/rssb.12377.

Barredo Arrieta, A., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., Garcia, S., Gil-Lopez, S., Molina, D., Benjamins, R., Chatila, R., and Herrera, F. (2020). Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Inf. Fusion*, 58(C):82–115. DOI:

10.1016/j.inffus.2019.12.012.

Başağaoğlu, H., Chakraborty, D., Do Lago, C., Gutierrez, L., Şahinli, M., Giacomoni, M., Furl, C., Mirchi, A., Moriasi, D., and Şengör, S. (2022). A Review on Interpretable and Explainable Artificial Intelligence in Hydroclimatic Applications. *Water (Switzerland)*, 14(8). DOI: 10.3390/w14081230.

Bedi, S., Samal, A., Ray, C., and Snow, D. (2020). Comparative evaluation of machine learning models for groundwater quality assessment. *Environmental Monitoring and Assessment*, 192(12). DOI: 10.1007/s10661-020-08695-3.

Bharati, S., Mondal, M. R. H., and Podder, P. (2023). A Review on Explainable Artificial Intelligence for Healthcare: Why, How, and When? *IEEE Transactions on Artificial Intelligence*, pages 1–15. DOI: 10.1109/TAI.2023.3266418.

Boulitsakis Logothetis, S., Green, D., Holland, M., and Al Moubayed, N. (2023). Predicting acute clinical deterioration with interpretable machine learning to support emergency care decision making. *Scientific Reports*, 13(1). DOI: 10.1038/s41598-023-40661-0.

Breiman, L. (1996). Bagging predictors. *Machine Learning*, 24:123–140. DOI: 10.1007/BF00058655.

Burkart, N. and Huber, M. F. (2021). A Survey on the Explainability of Supervised Machine Learning. *J. Artif. Int. Res.*, 70:245–317. DOI: 10.1613/jair.1.12228.

Calderón-Díaz, M., Silvestre Aguirre, R., Vásconez, J. P., Yáñez, R., Roby, M., Querales, M., and Salas, R. (2024). Explainable Machine Learning Techniques to Predict Muscle Injuries in Professional Soccer Players through Biomechanical Analysis. *Sensors*, 24(1). DOI: 10.3390/s24010119.

Carvalho, D. V., Pereira, E. M., and Cardoso, J. S. (2019). Machine learning interpretability: A survey on methods and metrics. 8(8). DOI: 10.3390/electronics8080832.

Chen, T. and Guestrin, C. (2016). Xgboost: A scalable tree boosting system. *CoRR*, abs/1603.02754. DOI: 10.1145/2939672.2939785.

Christodoulou, E. and Gregoriades, A. (2023). Applying Machine Learning in Personality-based Persuasion Marketing. In *2023 IEEE International Conference on Data Mining Workshops (ICDMW)*, pages 16–23. DOI: 10.1109/ICDMW60847.2023.00010.

Chu, L., Nelen, J., Crivellari, A., Masiliūnas, D., Hein, C., and Lofi, C. (2024). Relationships between geo-spatial features and COVID-19 hospitalisations revealed by machine learning models and SHAP values. *International Journal of Digital Earth*, 17(1). DOI: 10.1080/17538947.2024.2358851.

Covert, I., Lundberg, S., and Lee, S.-I. (2022). Explaining by removing: A unified framework for model explanation.

Das, S., Agarwal, N., Venugopal, D., Sheldon, F. T., and Shiva, S. (2020). Taxonomy and Survey of Interpretable Machine Learning Method. In *2020 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 670–677. DOI: 10.1109/SSCI47803.2020.9308404.

De Bock, K., Coussement, K., Caigny, A., Słowiński, R., Baesens, B., Boute, R., Choi, T.-M., Delen, D., Kraus, M., Lessmann, S., Verbeke, W., and Weber, R. (2023). Explainable AI for Operational Research: A defining framework, methods, applications, and a research agenda. *European Journal of Operational Research*. DOI: 10.1016/j.ejor.2023.09.026.

de Oliveira, R. and Martens, D. (2021). A framework and benchmarking study for counterfactual generating methods on tabular data. *Applied Sciences (Switzerland)*, 11(16). DOI: 10.3390/app11167274.

Delgado-Panadero, Á., Hernández-Lorca, B., García-Ordás, M. T., and Benítez-Andrades, J. A. (2022). Implementing local-explainability in Gradient Boosting Trees: Feature Contribution. *Inf. Sci.*, 589(C):199–212. DOI: 10.1016/j.ins.2021.12.111.

Deng, H. (2014). Interpreting Tree Ensembles with inTrees. *CoRR*. DOI: 10.48550/arXiv.1408.5456.

Doshi-Velez, F. and Kim, B. (2017). Towards a rigorous science of interpretable machine learning. DOI: 10.48550/arXiv.1702.08608.

Du, J., Chang, X., Ye, C., Zeng, Y., Yang, S., Wu, S., and Li, L. (2023). Developing a hypertension visualization risk prediction system utilizing machine learning and health check-up data. *Scientific Reports*, 13(1). DOI: 10.1038/s41598-023-46281-y.

Feng, J., Zhang, H., Gao, K., Liao, Y., Yang, J., and Wu, G. (2022). A machine learning and game theory-based approach for predicting creep behavior of recycled aggregate concrete. *Case Studies in Construction Materials*, 17. DOI: 10.1016/j.cscm.2022.e01653.

Fisher, A., Rudin, C., and Dominici, F. (2019). All models are wrong, but many are useful: Learning a variable's importance by studying an entire class of prediction models simultaneously.

Freund, Y. and Schapire, R. E. (1997). A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. *Journal of Computer and System Sciences*, 55(1):119–139. DOI: 10.1006/jcss.1997.1504.

Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, 29(5):1189 – 1232. DOI: 10.1214/aos/1013203451.

GhoshRoy, D., Alvi, P., and Santosh, K. (2023). Explainable AI to Predict Male Fertility Using Extreme Gradient Boosting Algorithm with SMOTE. *Electronics (Switzerland)*, 12(1). DOI: 10.3390/electronics12010015.

Giudici, P. and Raffinetti, E. (2021). Shapley-Lorenz eXplainable Artificial Intelligence. *Expert Systems with Applications*, 167. DOI: 10.1016/j.eswa.2020.114104.

Gramegna, A. and Giudici, P. (2020). Why to buy insurance? An explainable artificial intelligence approach. *Risks*, 8(4):1–9. DOI: 10.3390/risks8040137.

Guidotti, R., Monreale, A., Ruggieri, S., Turini, F., Giannotti, F., and Pedreschi, D. (2018). A survey of methods for explaining black box models. *ACM Computing Surveys*, 51(5). DOI: 10.1145/3236009.

Hara, S. and Hayashi, K. (2017). Making tree ensembles interpretable: A bayesian model selection approach.

He, S., Niu, G., Sang, X., Sun, X., Yin, J., and Chen, H. (2023). Machine Learning Framework with Feature Importance Interpretation for Discharge Estimation: A Case Study in Huitanggou Sluice Hydrological Station, China.

Water (Switzerland), 15(10). DOI: 10.3390/w15101923.

Herrera, G. P., Constantino, M., Su, J.-J., and Naranpanawa, A. (2023). The use of ICTs and income distribution in Brazil: A machine learning explanation using SHAP values. *Telecommun. Policy*, 47(8). DOI: 10.1016/j.telpol.2023.102598.

Ho, I.-T., Matysik, M., Herrera, L., Yang, J., Guderlei, R., Laussegger, M., Schrantz, B., Hammer, R., Miranda-Quintana, R., and Smiatek, J. (2022). Combination of explainable machine learning and conceptual density functional theory: applications for the study of key solvation mechanisms. *Physical Chemistry Chemical Physics*, 24(46):28314–28324. DOI: 10.1039/d2cp04428e.

Islam, S. R., Eberle, W., Ghafoor, S. K., and Ahmed, M. (2021). Explainable artificial intelligence approaches: A survey. DOI: 10.48550/arXiv.2101.09429.

Jas, K., Mangalathu, S., and Dodagoudar, G. R. (2024). Evaluation and analysis of liquefaction potential of gravelly soils using explainable probabilistic machine learning model. *Computers and Geotechnics*, 167. DOI: 10.1016/j.compgeo.2023.106051.

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., and Liu, T.-Y. (2017). LightGBM: a highly efficient gradient boosting decision tree. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, pages 3149–3157, Red Hook, NY, USA. Curran Associates Inc.. DOI: 10.5555/3294996.3295074.

Kök, I., Okay, F. Y., Muyanlı, Ö., and Özdemir, S. (2023). Explainable Artificial Intelligence (XAI) for Internet of Things: A Survey. *IEEE Internet of Things Journal*, 10(16):14764–14779. DOI: 10.1109/JIOT.2023.3287678.

Kookalani, S., Cheng, B., and Torres, J. (2022a). Structural performance assessment of GFRP elastic gridshells by machine learning interpretability methods. *Frontiers of Structural and Civil Engineering*, 16(10):1249–1266. DOI: 10.1007/s11709-022-0858-5.

Kookalani, S., Nyunn, S., and Xiang, S. (2022b). Form-finding of lifting self-forming GFRP elastic gridshells based on machine learning interpretability methods. *Structural Engineering and Mechanics*, 84(5):605–618. DOI: 10.12989/sem.2022.84.5.605.

Lazaridis, P., Kavvadias, I., Demertzis, K., Iliadis, L., and Vasiliadis, L. (2023). Interpretable Machine Learning for Assessing the Cumulative Damage of a Reinforced Concrete Frame Induced by Seismic Sequences. *Sustainability (Switzerland)*, 15(17). DOI: 10.3390/su151712768.

Li, Y., Jia, C., Chen, H., Su, H., Chen, J., and Wang, D. (2023a). Machine Learning Assessment of Damage Grade for Post-Earthquake Buildings: A Three-Stage Approach Directly Handling Categorical Features. *Sustainability (Switzerland)*, 15(18). DOI: 10.3390/su151813847.

Li, Z., Zhu, Y., and Van Leeuwen, M. (2023b). A Survey on Explainable Anomaly Detection. *ACM Transactions on Knowledge Discovery from Data*, 18(1). DOI: 10.1145/3609333.

Liang, S., Shen, Y., and Ren, X. (2022). Comparative study of influential factors for punching shear resistance/failure of RC slab-column joints using machine-learning models. *Structures*, 45:1333–1349. DOI: 10.1016/j.istruc.2022.09.110.

Liu, Y., Huang, F., Ma, L., Zeng, Q., and Shi, J. (2024). Credit scoring prediction leveraging interpretable ensemble learning. *Journal of Forecasting*, 43(2):286 – 308. DOI: 10.1002/for.3033.

Lundberg, S., Erion, G., Chen, H., DeGrave, A., Prutkin, J., Nair, B., Katz, R., Himmelfarb, J., Bansal, N., and Lee, S.-I. (2020). From local explanations to global understanding with explainable AI for trees. *Nature Machine Intelligence*, 2(1):56–67. DOI: 10.1038/s42256-019-0138-9.

Lundberg, S. and Lee, S.-I. (2017). A unified approach to interpreting model predictions. DOI: 10.48550/arXiv.1705.07874.

Lundberg, S. M., Erion, G. G., and Lee, S.-I. (2019). Consistent individualized feature attribution for tree ensembles. DOI: 10.48550/arXiv.1802.03888.

Lv, H., Li, H., Chen, Y., and Feng, T. (2023). An origin-destination level analysis on the competitiveness of bike-sharing to underground using explainable machine learning. *Journal of Transport Geography*, 113. DOI: 10.1016/j.jtrangeo.2023.103716.

Ma, C., Wang, S., Zhao, J., Xiao, X., Xie, C., and Feng, X. (2023a). Prediction of shear strength of RC deep beams based on interpretable machine learning. *Construction and Building Materials*, 387. DOI: 10.1016/j.conbuildmat.2023.131640.

Ma, C., Wang, W., Wang, S., Guo, Z., and Feng, X. (2023b). Prediction of shear strength of RC slender beams based on interpretable machine learning. *Structures*, 57. DOI: 10.1016/j.istruc.2023.105171.

Minh, D., Wang, H. X., Li, Y. F., and Nguyen, T. N. (2022). Explainable artificial intelligence: a comprehensive review. *Artificial Intelligence Review*, 55(5):3503–3568. DOI: 10.1007/s10462-021-10088-y.

Moustafa, N., Koroniotis, N., Keshk, M., Zomaya, A., and Tari, Z. (2023). Explainable Intrusion Detection for Cyber Defences in the Internet of Things: Opportunities and Solutions. *IEEE Communications Surveys and Tutorials*, 25(3):1775–1807. DOI: 10.1109/COMST.2023.3280465.

Nadeem, A., Vos, D., Cao, C., Pajola, L., Dieck, S., Baumgartner, R., and Verwer, S. (2023). SoK: Explainable Machine Learning for Computer Security Applications. In *2023 IEEE 8th European Symposium on Security and Privacy (EuroS&P)*, pages 221–240. DOI: 10.1109/EuroSP57164.2023.00022.

Nagahisarchoghaei, M., Nur, N., Cummins, L., Nur, N., Karimi, M., Nandanwar, S., Bhattacharyya, S., and Rahimi, S. (2023). An Empirical Survey on Explainable AI Technologies: Recent Trends, Use-Cases, and Categories from Technical and Application Perspectives. *Electronics (Switzerland)*, 12(5). DOI: 10.3390/electronics12051092.

Nguyen, H., Viviani, J.-L., and Ben Jabeur, S. (2023). Bankruptcy prediction using machine learning and Shapley additive explanations. *Review of Quantitative Finance and Accounting*. DOI: 10.1007/s11156-023-01192-x.

Obregon, J. and Jung, J.-Y. (2023). RuleCOSI+: Rule extraction for interpreting classification tree ensembles. *Inf. Fu-*

sion, 89(C):355–381. DOI: 10.1016/j.inffus.2022.08.021.

Obregon, J., Kim, A., and Jung, J. Y. (2019). Rule-COSI: Combination and simplification of production rules from boosted decision trees for imbalanced classification. *Expert Systems with Applications*, 126:64–82. DOI: 10.1016/j.eswa.2019.02.012.

Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A. V., and Gulin, A. (2019). Catboost: unbiased boosting with categorical features. DOI: 10.48550/arXiv.1706.09516.

Quinlan, J. R. (1993). Chapter 2 - constructing decision trees. In Quinlan, J. R., editor, *C4.5*, pages 17–26. Morgan Kaufmann, San Francisco (CA). DOI: 10.1016/B978-0-08-050058-4.50007-3.

Rawal, A., McCoy, J., Rawat, D. B., Sadler, B. M., and Amant, R. S. (2022). Recent Advances in Trustworthy Explainable Artificial Intelligence: Status, Challenges, and Perspectives. *IEEE Transactions on Artificial Intelligence*, 3(6):852–866. DOI: 10.1109/TAI.2021.3133846.

Rhee, J., Park, K., Lee, S., Jang, S., and Yoon, S. (2020). Detecting hydrological droughts in ungauged areas from remotely sensed hydro-meteorological variables using rule-based models. *Natural Hazards*, 103(3):2961–2988. DOI: 10.1007/s11069-020-04114-5.

Ribeiro, M. T., Singh, S., and Guestrin, C. (2016). ”Why should I trust you?” Explaining the predictions of any classifier. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, volume 13-17-August-2016, pages 1135–1144. Association for Computing Machinery. DOI: 10.1145/2939672.2939778.

Rjoub, G., Bentahar, J., Abdel Wahab, O., Mizouni, R., Song, A., Cohen, R., Otrok, H., and Mourad, A. (2023). A Survey on Explainable Artificial Intelligence for Cybersecurity. *IEEE Transactions on Network and Service Management*, 20(4):5115–5140. DOI: 10.1109/TNSM.2023.3282740.

Sahakyan, M., Aung, Z., and Rahwan, T. (2021). Explainable Artificial Intelligence for Tabular Data: A Survey. *IEEE Access*, 9:135392–135422. DOI: 10.1109/ACCESS.2021.3116481.

Schapire, R. E. (1990). The strength of weak learnability. *Machine learning*, 5:197–227. DOI: 10.1109/SFCS.1989.63451.

Settouti, N. and Saidi, M. (2024). Preliminary analysis of explainable machine learning methods for multiple myeloma chemotherapy treatment recognition. *Evolutionary Intelligence*, 17(1):513 – 533. DOI: 10.1007/s12065-023-00833-3.

Shen, Y., Wu, L., and Liang, S. (2022). Explainable machine learning-based model for failure mode identification of RC flat slabs without transverse reinforcement. *Engineering Failure Analysis*, 141. DOI: 10.1016/j.engfailanal.2022.106647.

Shield, S. and Houston, A. (2022). Diagnosing Supercell Environments: A Machine Learning Approach. *Weather and Forecasting*, 37(5):771–785. DOI: 10.1175/waf-d-21-0098.1.

Shimizu, H., Enda, K., Shimizu, T., Ishida, Y., Ishizu, H., Ise, K., Tanaka, S., and Iwasaki, N. (2022). Machine Learning Algorithms: Prediction and Feature Selection for Clinical Refracture after Surgically Treated Fragility Fracture. *Journal of Clinical Medicine*, 11(7). DOI: 10.3390/jcm11072021.

Sokhansanj, B. and Rosen, G. (2022). Predicting Institution Outcomes for Inter Partes Review (IPR) Proceedings at the United States Patent Trial &amp; Appeal Board by Deep Learning of Patent Owner Preliminary Response Briefs. *Applied Sciences (Switzerland)*, 12(7). DOI: 10.3390/app12073656.

Stepin, I., Alonso, J. M., Catala, A., and Pereira-Fariña, M. (2021). A Survey of Contrastive and Counterfactual Explanation Generation Methods for Explainable Artificial Intelligence. *IEEE Access*, 9:11974–12001. DOI: 10.1109/ACCESS.2021.3051315.

Stojić, A., Stanić, N., Vuković, G., Stanišić, S., Perišić, M., Šoštarić, A., and Lazić, L. (2019). Explainable extreme gradient boosting tree-based prediction of toluene, ethylbenzene and xylene wet deposition. *Science of the Total Environment*, 653:140–147. DOI: 10.1016/j.scitotenv.2018.10.368.

Tan, S., Soloviev, M., Hooker, G., and Wells, M. T. (2020). Tree Space Prototypes: Another Look at Making Tree Ensembles Interpretable. In *FODS 2020 - Proceedings of the 2020 ACM-IMS Foundations of Data Science Conference*, pages 23–34. Association for Computing Machinery, Inc. DOI: 10.1145/3412815.3416893.

Tang, H., Yu, J., Lin, B., Geng, Y., Wang, Z., Chen, X., Yang, L., Lin, T., and Xiao, F. (2023). Airport terminal passenger forecast under the impact of COVID-19 outbreaks: A case study from China. *Journal of Building Engineering*, 65. DOI: 10.1016/j.jobe.2022.105740.

Tran, D. A., Tsujimura, M., Ha, N. T., Nguyen, V. T., Binh, D. V., Dang, T. D., Doan, Q. V., Bui, D. T., Anh Ngoc, T., Phu, L. V., Thuc, P. T. B., and Pham, T. D. (2021). Evaluating the predictive power of different machine learning algorithms for groundwater salinity prediction of multi-layer coastal aquifers in the Mekong Delta, Vietnam. *Ecological Indicators*, 127. DOI: 10.1016/j.ecolind.2021.107790.

Vaulet, T., Al-Memar, M., Fourie, H., Bobdiwala, S., Saso, S., Pipi, M., Stalder, C., Bennett, P., Timmerman, D., Bourne, T., and De Moor, B. (2022). Gradient boosted trees with individual explanations: An alternative to logistic regression for viability prediction in the first trimester of pregnancy. *Comput. Methods Prog. Biomed.*, 213(C). DOI: 10.1016/j.cmpb.2021.106520.

Wang, S., Ma, C., Wang, W., Hou, X., Xiao, X., Zhang, Z., Liu, X., and Liao, J. (2023a). Prediction of Failure Modes and Minimum Characteristic Value of Transverse Reinforcement of RC Beams Based on Interpretable Machine Learning. *Buildings*, 13(2). DOI: 10.3390/buildings13020469.

Wang, X., Qiao, Y., Cui, Y., Ren, H., Zhao, Y., Linghu, L., Ren, J., Zhao, Z., Chen, L., and Qiu, L. (2023b). An explainable artificial intelligence framework for risk prediction of COPD in smokers. *BMC Public Health*, 23(1). DOI: 10.1186/s12889-023-17011-w.

Wu, H., Ruan, W., Wang, J., Zheng, D., Liu, B., Geng, Y., Chai, X., Chen, J., Li, K., Li, S., and Helal, S. (2023). Interpretable Machine Learning for COVID-19: An Empirical Study on Severity Prediction Task. *IEEE Transactions on Artificial Intelligence*, 4(4):764–777. DOI: 10.1109/TAI.2021.3092698.

Yasodhara, A., Asgarian, A., Huang, D., and Sobhani, P. (2021). On the Trustworthiness of Tree Ensemble Explainability Methods. In *Machine Learning and Knowledge Extraction: 5th IFIP TC 5, TC 12, WG 8.4, WG 8.9, WG 12.9 International Cross-Domain Conference, CD-MAKE 2021, Virtual Event, August 17–20, 2021, Proceedings*, pages 293–308, Berlin, Heidelberg. Springer-Verlag. DOI: 10.1007/978-3-030-84060-0_19.

Zhang, Y., Cheng, L., Pan, A., Hu, C., and Wu, K. (2024). Phase Transformation Temperature Prediction in Steels via Machine Learning. *Materials*, 17(5). DOI: 10.3390/ma17051117.