University of Maribor Press

DOI: https://doi.org/10.18690/um.fov.5.2026.1
ISBN: 978-961-299-160-9

Tree-based Machine Learning Methods for Wind Farm Data

Manohar Gowdru Shridhara,
Ľubomír Antoni, and
Gabriel Semanišin

Abstract
Environmental and energy datasets are typically characterized by nonlinear dependencies and a combination of numerical and categorical variables. Such characteristics require more adaptable computational approaches. In this context, we explore tree-based machine learning methods since they provide a high predictive performance and a high level of interpretability. In this chapter, we present a comparative study of selected tree-based regression models applied to real-world environmental data from the United States Wind Turbine Database. The evaluated methods include a single regression decision tree, a bagging-based Random Forest ensemble, and modern gradient boosting implementations represented by CatBoost and LightGBM. All models are trained within a unified framework using standard regression performance metrics. We demonstrate that ensemble-based approaches substantially outperform a single decision tree in our experimental results. In particular, boosting-based models achieve higher predictive accuracy, with LightGBM providing the best overall performance in terms of squared error metrics and coefficient of determination. Feature importance analysis further highlights the key role of technical turbine characteristics and categorical descriptors. The findings confirm that modern gradient boosting frameworks represent a powerful and effective solution for regression tasks involving large-scale environmental and energy-related datasets.

Keywords: tree-based learning; regression; ensemble methods; gradient boosting; Random Forest; LightGBM; environmental data; wind energy

Manohar Gowdru Shridhara
Pavol Jozef Šafárik University in Košice, Faculty of Science, Jesenná 5, 04001 Košice, Slovakia
e-mail: manohar.gowdru.shridhara@student.upjs.sk
Ľubomír Antoni
Pavol Jozef Šafárik University in Košice, Faculty of Science, Jesenná 5, 04001 Košice, Slovakia
e-mail: lubomir.antoni@upjs.sk
Gabriel Semanišin
Pavol Jozef Šafárik University in Košice, Faculty of Science, Jesenná 5, 04001 Košice, Slovakia
e-mail: gabriel.semanisin@upjs.sk

1 Introduction

The rapid growth of environmental and energy-related datasets has significantly increased the demand for interpretable machine learning methods capable of modeling nonlinear relationships. In particular, data-driven approaches play an increasingly important role in the analysis and optimization of renewable energy systems, where large-scale heterogeneous datasets are commonly generated. Among various machine learning paradigms, decision tree–based methods have gained an important role due to their flexibility, interpretability, and strong performance in regression and prediction tasks [1, 2].

Decision trees represent a class of supervised learning models that recursively partition the input space in order to minimize prediction error. Their natural ability to capture nonlinear dependencies and interactions among variables makes them particularly appropriate for environmental data, which often contains complex spatial and technical characteristics. However, single decision trees tend to overfit and have high variance, which has motivated the development of ensemble-based approaches such as bagging and boosting [3, 4, 5].

Bagging methods, most notably Random Forests, address the variance issue by constructing an ensemble of decorrelated trees trained on bootstrap samples of the data [4]. Random Forests have proven to be highly effective in a wide range of regression problems, including applications in environmental monitoring and renewable energy. In contrast, boosting techniques build ensembles sequentially, where each subsequent model focuses on correcting the errors of its predecessors. Gradient Boosting and its modern variants have demonstrated strong predictive performance, mainly in structured tabular data [5].

In addition to classical tree-based ensembles, fuzzy decision trees extend the standard framework by incorporating fuzzy logic principles [6]. By allowing soft partitioning of the feature space, fuzzy trees provide a natural mechanism for covering uncertainty and gradual transitions between decision regions, which are common in real-world environmental data. These properties make fuzzy tree-based models attractive not only from a predictive point of view but also in terms of interpretability and robustness.

In this paper, we focus on a theoretical overview and experimental evaluation of selected tree-based machine learning methods for regression tasks in the context of environmental data analysis. The methods considered include classical decision trees, Random Forests, boosting-based ensembles, and fuzzy decision tree approaches. The experimental part of the study is based on real-world data from the United States Wind Turbine Database, which provides detailed technical and geographical information about wind energy installations across the United States. By combining theoretical insights with empirical results, we aim to highlight the strengths and limitations of tree-based learning methods when applied to large-scale environmental datasets. The paper is structured in the following way: In Section 2, we present theoretical foundations and principles of decision tree-based learning methods. In Section 3, we provide a description of the wind farm dataset. In Section 4, we present the methodology and our experimental setup. We summarize the results and discussion in Section 5.

2 Learning methods of decision trees

2.1 Decision trees for regression

Decision trees are hierarchical, non-parametric supervised learning models that recursively partition the input feature space into disjoint regions in order to approximate an underlying target function. In the case of regression tasks, the goal is to predict continuous output values by minimizing an error criterion within each partition. Regression trees are particularly attractive due to their intuitive structure, simplicity of interpretation, and ability to model nonlinear relationships without requiring explicit assumptions about the data distribution [3].

A regression decision tree is typically constructed using a top-down, greedy algorithm. At each internal node, the algorithm selects a splitting attribute and a corresponding threshold that best separates the data according to a predefined impurity or error measure. Commonly used criteria for regression trees include the mean squared error (MSE) or variance reduction, where the optimal split minimizes the weighted sum of variances in the resulting child nodes [7]. The tree-growing process continues recursively until a stopping condition is met, such as a minimum number of samples in a leaf node or a maximum tree depth.

Let d be the number of input features. A regression tree partitions the feature space by selecting, at each internal node, a feature index $j \in \{1, \dots, d\}$ and a threshold $t \in \mathbb{R}$ that minimizes the within-node squared error after the split. Let S denote the set of training indices reaching the current node. The split induces two subsets

$$S_L(j,t) = \{i \in S : x_{ij} \le t\}, \qquad S_R(j,t) = \{i \in S : x_{ij} > t\}.$$

A common objective for regression trees is the sum of squared errors (SSE) in the two children:

$$(j^*, t^*) \in \arg\min_{j,t} \left(\sum_{i \in S_L(j,t)} (y_i - \bar{y}_L)^2 + \sum_{i \in S_R(j,t)} (y_i - \bar{y}_R)^2 \right),$$

where $\bar{y}_L$ and $\bar{y}_R$ are the mean target values in the left and right child, respectively. When a stopping condition is met (e.g., maximum depth, minimum number of samples in a node, or no improvement in the objective), the node becomes a leaf and outputs the constant prediction

$$\hat{y}(x) = \bar{y}_S = \frac{1}{|S|} \sum_{i \in S} y_i.$$

Algorithm 1 summarizes a simplified top-down procedure for constructing a regression tree which can be found e.g. in [3].

Each terminal node, or leaf, represents a local model that outputs a constant prediction, usually computed as the mean of the target values of the samples contained in that leaf. While this piecewise-constant approximation enables regression trees to capture complex nonlinear dependencies and interactions between variables, it also makes them

Algorithm 1 Simplified procedure for building a regression tree [3]

Require: Training data $\{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$, current node index set S, depth D
Ensure: A regression tree node (either internal split or leaf)
if $D \geq D_{\max}$ **or** $|S| < n_{\min}$ **then**
  return leaf with prediction $\bar{y}_S$
end if
Find (j^*, t^*) minimizing SSE over candidate splits (feature j, threshold t)
Compute $S_L \leftarrow \{i \in S : x_{ij^*} \leq t^*\}$ and $S_R \leftarrow \{i \in S : x_{ij^*} > t^*\}$
if $S_L = \emptyset$ **or** $S_R = \emptyset$ **then**
  return leaf with prediction $\bar{y}_S$
end if
Create internal node storing (j^*, t^*)
Left child $\leftarrow$ BuildTree($\{(\mathbf{x}_i, y_i)\}, S_L, D + 1$)
Right child $\leftarrow$ BuildTree($\{(\mathbf{x}_i, y_i)\}, S_R, D + 1$)
return internal node with left and right children

highly sensitive to small perturbations in the training data. As a result, regression trees are known to suffer from high variance and a tendency to overfit, especially when grown to full depth [3].

Despite these limitations, regression trees provide several important advantages in the context of environmental and energy-related datasets. They naturally handle mixed data types, are robust to outliers, and allow for straightforward assessment of variable importance through split statistics. These properties make decision trees a fundamental building block for more advanced ensemble methods such as Random Forests and boosting algorithms, which aim to improve predictive performance by addressing the instability of individual trees.

2.2 Bagging and random forests

One of the main limitations of single regression trees is their high variance, which results from their sensitivity to small changes in the training data. Ensemble learning methods aim to address this issue by combining multiple base learners in order to obtain more stable and accurate predictions. Bagging, short for bootstrap aggregating, is one of the earliest and most widely used ensemble techniques designed to reduce variance without increasing bias [3, 8].

The bagging approach constructs multiple training datasets by sampling with replacement from the original dataset. A separate regression tree is trained on each bootstrap sample, and the final prediction is obtained by averaging the predictions of all individual trees. Since each tree is trained on a slightly different subset of the data, the ensemble benefits from reduced variance while preserving the expressive power of deep trees. Bagging is particularly effective for unstable learners such as decision trees, making it a natural extension of the regression tree model [3].

Random Forests further extend the bagging principle by introducing an additional layer of randomness during tree construction. In addition to bootstrap sampling of the

training data, Random Forests randomly select a subset of input features at each split node, thereby reducing the correlation between individual trees in the ensemble [4]. This decorrelation effect significantly improves generalization performance, especially in high-dimensional settings or when strong predictor variables dominate the splitting process.

Let $\{(\mathbf{x}_i, y_i)\}_{i=1}^N$ denote the training dataset. In the bagging framework, B bootstrap samples are generated by sampling with replacement from the original dataset. For each bootstrap sample $b \in \{1, \ldots, B\}$, an independent regression tree $f_b(\mathbf{x})$ is trained.

The bagging predictor is defined as the average of individual tree predictions:

$$\hat{f}_{\text{bag}}(\mathbf{x}) = \frac{1}{B} \sum_{b=1}^{B} f_b(\mathbf{x}).$$

Random Forests extend the bagging principle by introducing additional randomness during tree construction. At each split node, only a randomly selected subset of features $\mathcal{M} \subset \{1, \ldots, d\}$ with $|\mathcal{M}| = m \ll d$ is considered when determining the optimal split. This feature subsampling mechanism reduces correlation among individual trees and improves ensemble generalization performance.

The Random Forest regression predictor is therefore given by

$$\hat{f}_{\text{RF}}(\mathbf{x}) = \frac{1}{B} \sum_{b=1}^{B} f_b^{(\mathcal{M})}(\mathbf{x}),$$

where $f_b^{(\mathcal{M})}$ denotes a regression tree trained using random feature selection at each split.

Algorithm 2 Simplified Random Forest regression algorithm [3]

Require: Training data $\{(\mathbf{x}_i, y_i)\}_{i=1}^N$, number of trees B, number of features per split m
Ensure: Random Forest regression model

```
for b = 1 to B do
    Draw a bootstrap sample S_b from the training data
    Train a regression tree f_b on S_b
    for each split node in f_b do
        Randomly select m features from {1, ..., d}
        Choose the best split using only the selected features
    end for
end for
return ensemble predictor f̂_RF(x) = (1/B) Σ_{b=1}^{B} f_b(x)
```

From a regression perspective, Random Forests approximate the target function by averaging predictions across a large number of randomized trees. This averaging mechanism leads to robust performance, strong resistance to overfitting, and stable predictions even in the presence of noise. Moreover, Random Forests provide built-in mechanisms for estimating feature importance, typically based on impurity reduction or permutation-

based measures, which are particularly valuable in environmental applications where understanding the influence of individual variables is essential [3].

Due to their favorable bias–variance trade-off, minimal parameter tuning requirements, and ability to process large heterogeneous datasets, Random Forests have become a standard baseline method in environmental data analysis and renewable energy research.

2.3 Boosting methods

While bagging-based ensembles primarily aim to reduce variance by training base learners independently, boosting methods adopt a different strategy by constructing models sequentially. The central idea of boosting is to iteratively focus on training instances that are difficult to predict, thereby progressively improving the overall performance of the ensemble. This adaptive learning process enables boosting methods to reduce both bias and variance, making them particularly effective for difficult regression tasks [9].

Boosting regression methods construct an additive model by sequentially combining weak learners, typically shallow regression trees. Let M denote the number of iterations. The resulting model can be expressed as

$$f_M(\mathbf{x}) = \sum_{m=1}^{M} \gamma_m h_m(\mathbf{x}),$$

where each $h_m(\mathbf{x})$ denotes a weak regression tree and γ_m is a scaling parameter controlling its contribution to the ensemble.

In gradient boosting, the learning process is formulated as an optimization problem in function space. Given a differentiable loss function $L(y, f(\mathbf{x}))$, the model is built iteratively by fitting each new base learner to the negative gradient of the loss with respect to the current model predictions [5]. For the commonly used squared error loss,

$$L(y, f(\mathbf{x})) = (y - f(\mathbf{x}))^2,$$

the negative gradient corresponds to the residual

$$r_{im} = y_i - f_{m-1}(\mathbf{x}_i).$$

At iteration m, a regression tree h_m is fitted to the residuals $\{r_{im}\}_{i=1}^{N}$, and the ensemble model is updated according to

$$f_m(\mathbf{x}) = f_{m-1}(\mathbf{x}) + \eta\, h_m(\mathbf{x}),$$

where $\eta \in (0, 1]$ denotes the learning rate that controls the contribution of each newly added tree. This iterative refinement allows boosting models to progressively

reduce prediction error by concentrating learning capacity on previously mispredicted instances.

Algorithm 3 Simplified Gradient Boosting algorithm for regression [3]

Require: Training data $\{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$, number of iterations M, learning rate η
Ensure: Gradient Boosting regression model
1: Initialize model with constant prediction

$$f_0(\mathbf{x}) = \arg\min_c \sum_{i=1}^{N} L(y_i, c)$$

2: **for** $m = 1$ to M **do**
3: Compute residuals $r_{im} \leftarrow y_i - f_{m-1}(\mathbf{x}_i)$
4: Fit regression tree h_m to $\{(\mathbf{x}_i, r_{im})\}_{i=1}^{N}$
5: Update model: $f_m(\mathbf{x}) \leftarrow f_{m-1}(\mathbf{x}) + \eta\, h_m(\mathbf{x})$
6: **end for**
7: **return** final model $f_M(\mathbf{x})$

Modern boosting frameworks extend the classical gradient boosting paradigm by incorporating advanced regularization strategies, efficient tree-growing mechanisms, and specialized handling of categorical features. Notable examples include CatBoost and LightGBM, which have demonstrated state-of-the-art performance on structured tabular datasets [10, 27].

CatBoost introduces ordered boosting and target-based encoding schemes to reduce bias in predictions caused by many-category features [10]. These properties make CatBoost particularly suitable for real-world datasets containing mixed numerical and categorical variables, as commonly encountered in environmental and energy-related applications.

LightGBM employs histogram-based feature binning and a leaf-wise tree growth strategy that enable efficient modeling of complex feature interactions with reduced computational cost [27]. By choosing splits that most reduce the loss, LightGBM often achieves higher predictive performance compared to level-wise boosting approaches, especially on large-scale datasets.

Despite their strong predictive capabilities, boosting-based models are generally more sensitive to noise and hyperparameter configuration than bagging-based ensembles. Overfitting may occur if there are too many boosting iterations or the individual trees are too complex. Nevertheless, when appropriately regularized and tuned, modern boosting methods frequently outperform Random Forests in regression tasks that require capturing nonlinear relationships and interactions between attributes.

2.4 Fuzzy decision trees and fuzzy ensembles

Classical decision trees rely on crisp, binary splits of the input feature space, which may be insufficient for modeling gradual relationships and uncertainty commonly present in

real-world data. Fuzzy decision trees address this limitation by incorporating concepts from fuzzy logic [6], allowing instances to belong to multiple decision nodes with varying degrees of membership. This soft partitioning of the feature space enables smoother decision boundaries and improved robustness to noise [12].

In fuzzy decision trees, splitting criteria are based on fuzzy membership functions rather than hard thresholds. Continuous attributes are typically represented using linguistic terms, such as *low*, *medium*, and *high*, each associated with a corresponding membership function. During the tree construction process, samples propagate through multiple branches simultaneously, weighted by their membership degrees. Leaf nodes aggregate these contributions to produce final predictions, often through weighted averaging mechanisms [12].

The use of fuzzy logic enhances interpretability by enabling rule-based representations that are similar to human reasoning. Each root-to-leaf path can be interpreted as a fuzzy rule, making fuzzy decision trees particularly attractive in domains where transparency and explainability are essential. Moreover, fuzzy trees have increased tolerance to measurement uncertainty and imprecise attribute values, which are typical characteristics of environmental and energy-related datasets.

Fuzzy ensemble methods extend the principles of fuzzy decision trees by combining multiple fuzzy trees into an ensemble framework. Similar to classical bagging and boosting, fuzzy ensembles aim to improve predictive performance and stability. Approaches such as fuzzy random forests and fuzzy boosting incorporate randomness or sequential learning while preserving fuzzy partitions of the feature space [13]. These hybrid models aim to balance predictive accuracy with interpretability, using the strengths of both ensemble learning and fuzzy reasoning.

In fuzzy decision trees, the partitioning of the feature space is based on fuzzy sets rather than crisp thresholds. Let x_{ij} denote the value of feature j for instance i. Each continuous attribute is associated with a collection of fuzzy sets $\{A_{j1}, \ldots, A_{jk}\}$ characterized by membership functions

$$\mu_{jl} : \mathbb{R} \to [0, 1],$$

which quantify the degree to which x_{ij} belongs to the linguistic term A_{jl}.

At a given internal node, an instance is propagated to multiple child nodes simultaneously, weighted by its membership degrees. For a node corresponding to fuzzy set A_{jl}, the weighted subset of instances is defined as

$$S_{jl} = \{(i, \mu_{jl}(x_{ij})) : i \in S\},$$

where S denotes the set of instances reaching the parent node.

For regression tasks, the prediction at a fuzzy leaf node is typically computed as a weighted average of target values:

$$\hat{y}_A = \frac{\sum_{i \in S} \mu_A(\mathbf{x}_i)\, y_i}{\sum_{i \in S} \mu_A(\mathbf{x}_i)},$$

where $\mu_A(\mathbf{x}_i)$ represents the aggregated membership degree of instance $\mathbf{x}_i$ in the corresponding fuzzy rule. This formulation allows smooth transitions between decision regions and provides robustness against measurement uncertainty.

Algorithm 4 Simplified construction of a fuzzy regression tree

Require: Training data $\{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$, fuzzy partitions for each feature
Ensure: Fuzzy regression tree
if stopping criterion satisfied **then**
    Compute weighted prediction $\hat{y}$ using membership degrees
    return fuzzy leaf node
end if
Select feature j^* and corresponding fuzzy sets $\{A_{j^*l}\}$
for each fuzzy set A_{j^*l} **do**
    Compute membership degrees $\mu_{j^*l}(x_{ij^*})$ for all instances
    Propagate instances to child node with weights μ_{j^*l}
    Recursively build fuzzy subtree
end for
return internal fuzzy node with fuzzy children

Fuzzy ensemble models combine multiple fuzzy regression trees by aggregating their predictions. Given an ensemble of B fuzzy trees $\{f_b\}_{b=1}^{B}$, the final prediction is commonly obtained as

$$\hat{f}_{\text{fuzzy}}(\mathbf{x}) = \frac{1}{B} \sum_{b=1}^{B} f_b(\mathbf{x}),$$

analogously to classical ensemble methods, while preserving fuzzy membership propagation within individual trees.

Fuzzy tree-based methods provide a sound alternative for regression tasks characterized by uncertainty, vagueness, and nonlinear interactions. As such, fuzzy decision trees and fuzzy ensembles can represent a valuable extension of classical tree-based learning methods in environmental data analysis.

3 Dataset description

This section describes the dataset used in the experimental part of the study. The focus is placed on the origin of the data, its scope, and its relevance for modeling regression tasks in the context of environmental and renewable energy analysis. Particular attention is given to the structure of the dataset and its suitability for evaluating tree-based machine learning methods.

3.1 US Wind turbine database

The experimental evaluation is based on data obtained from the United States Wind Turbine Database (USWTDB), a comprehensive and publicly available repository maintained by the U.S. Geological Survey in collaboration with national laboratories and governmental agencies [14]. The database provides detailed information on utility-scale wind turbines installed across the United States and represents one of the most authoritative sources of wind energy infrastructure data.

The USWTDB contains records for tens of thousands of wind turbines, covering a wide geographical area and multiple generations of wind energy technologies. Each turbine is described by a rich set of attributes, including geographical coordinates, technical specifications, operational status, and installation metadata. The dataset is continuously updated to reflect newly installed turbines as well as revisions to existing records, ensuring a high level of accuracy and temporal relevance.

Due to its scale and heterogeneity, the USWTDB constitutes a representative example of real-world environmental data characterized by nonlinear relationships, mixed attribute types, and potential measurement uncertainty. These properties make the dataset particularly suitable for evaluating decision tree–based regression models and ensemble learning methods. In this study, a selected subset of numerical and categorical attributes is used to formulate a regression task aimed at modeling relationships between turbine characteristics and target variables of interest.

3.2 Data attributes

The United States Wind Turbine Database provides a comprehensive set of attributes describing individual wind turbines from both technical and geographical perspectives. Attribute definitions and metadata are publicly documented in a structured XML format, which ensures consistency and transparency across dataset versions [15]. In this study, we use a selected subset of these attributes relevant to regression-based modeling of wind turbine characteristics.

From a structural perspective, the attributes can be grouped into several logical categories. The first group consists of geographical attributes, including latitude and longitude coordinates, state identifiers, and county-level information. These variables capture the spatial distribution of wind turbines and indirectly reflect environmental and regulatory conditions that may influence turbine design and deployment.

The second group contains the technical and physical characteristics of wind turbines. These attributes include hub height, rotor diameter, nameplate capacity, and turbine manufacturer information. Such variables are directly related to turbine performance and are therefore particularly relevant for regression tasks aimed at modeling capacity-related or structural properties. The numerical nature of these attributes makes them well-suited for tree-based learning methods, which naturally deal with nonlinear interactions and threshold-based relationships.

Additional attributes describe installation and operational metadata, such as the year of installation, turbine status, and project identifiers. While some of these variables are categorical or temporal in nature, they provide contextual information that may improve predictive performance when appropriately encoded. In the experimental setup, categorical attributes are transformed into numerical representations when required, while attributes with limited analytical relevance are excluded.

Overall, the selected attributes form a heterogeneous feature space that combines spatial, technical, and contextual information. An overview of the selected attributes from the US Wind Turbine Database used in our regression task is shown in Table 1. The dataset contains 70 697 instances.

Table 1: Overview of selected attributes from the US Wind Turbine Database used in our regression task.

Attribute	Type	Description
Latitude	Numerical	Geographic latitude of the turbine location
Longitude	Numerical	Geographic longitude of the turbine location
State	Categorical	U.S. state of installation
County	Categorical	County-level administrative region
Year of Installation	Numerical	Year when the turbine became operational
Hub Height	Numerical	Height of the turbine hub above ground (m)
Rotor Diameter	Numerical	Diameter of the turbine rotor (m)
Nameplate Capacity	Numerical	Rated turbine power capacity (kW)
Turbine Manufacturer	Categorical	Manufacturer of the wind turbine
Turbine Model	Categorical	Model designation of the turbine

3.3 Data preprocessing

Before model training and evaluation, we applied a series of preprocessing steps to ensure data consistency and suitability for tree-based regression models. Given the heterogeneous structure of the United States Wind Turbine Database, particular attention was paid to attribute selection in order to avoid introducing unnecessary bias into the learning process.

We started the preprocessing pipeline by selecting a subset of attributes relevant to the regression task. The target variable was defined as the turbine nameplate capacity, while the input features consisted of selected technical, temporal, and geographical attributes. Specifically, numerical features included hub height, rotor diameter, total turbine height, longitude, latitude, year of installation, and the number of turbines within a plant. In addition, categorical attributes describing turbine state, manufacturer, and model were retained to capture design- and location-specific characteristics.

Rows with missing values in the target variable were removed from the dataset. Subsequently, observations containing missing values in any of the selected input features were also excluded. This conservative strategy was adopted to ensure that all models were trained and evaluated on complete cases only, thereby simplifying the experimental setup and improving result reproducibility.

Categorical attributes were explicitly converted to string representations to guarantee consistent data types and to prevent issues arising from mixed numerical and textual encodings. Numerical attributes were retained in their original scale, as tree-based learning methods are naturally insensitive to feature scaling and monotonic transformations.

After preprocessing, the dataset was randomly partitioned into training and testing subsets using an 80%–20% split with a fixed random seed. The training set was used exclusively for model fitting, while the testing set was reserved for performance evaluation. This separation enables an unbiased assessment of generalization performance and ensures a fair comparison of the evaluated regression models.

Overall, the preprocessing procedure reflects practical conditions commonly encountered in data analysis, prioritizing robustness and transparency over aggressive data transformation. The resulting dataset provides a reliable basis for the experimental comparison of decision tree–based regression methods presented in the following sections.

4 Methodology

This section describes the experimental design used to evaluate the performance of selected tree-based machine learning methods on the wind turbine dataset. The focus is placed on the formulation of the regression task, the definition of input and output variables, and the methodological considerations ensuring a fair and reproducible comparison of different models (Fig. 1).

4.1 Regression task definition

The considered learning problem is formulated as a supervised regression task, where the objective is to model the relationship between a set of input attributes describing wind turbine characteristics and a continuous target variable. Let $\mathbf{x}_i \in \mathbb{R}^d$ denote the feature vector associated with the i-th wind turbine, where d represents the number of selected attributes, and let $y_i \in \mathbb{R}$ denote the corresponding target value.

Given a training dataset $\{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$, the goal of the regression model is to learn a function $f : \mathbb{R}^d \to \mathbb{R}$ that approximates the underlying dependency between turbine features and the target variable. The learned function is subsequently used to predict the target values for unseen instances in the test dataset.

In this study, the input feature vector $\mathbf{x}_i$ comprises a combination of geographical attributes (such as latitude and longitude) and technical parameters (including hub height, rotor diameter). The target variable (turbine capacity) is selected to represent a contin-

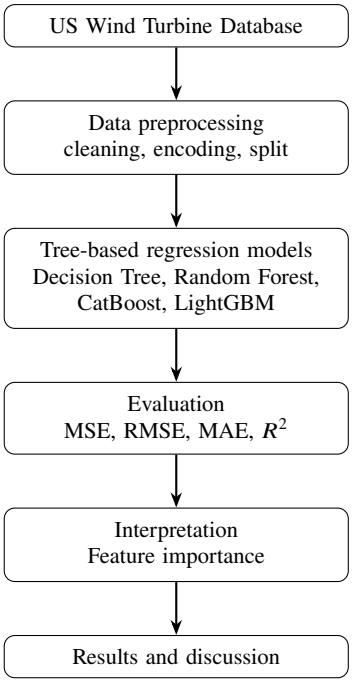

Fig. 1: Simplified compact workflow of the proposed approach.

uous turbine-related characteristic of practical relevance in the context of wind energy analysis. This formulation allows the evaluation of model performance in capturing nonlinear interactions between spatial and technical factors.

The regression task is designed to reflect realistic modeling conditions encountered in environmental datasets, where the relationships between variables are often complex and influenced by multiple interacting factors.

4.2 Implemented models

To evaluate the effectiveness of decision tree-based learning approaches for environmental regression tasks, several representative models were implemented and compared within a unified experimental framework. The selected models cover both classical tree learners and modern ensemble-based methods that are widely adopted in contemporary machine learning practice for tabular data.

As a baseline model, a single regression decision tree was employed. This model serves as a reference point for assessing the benefits of ensemble learning techniques. The regression tree was trained using variance-based splitting criteria and constrained by regularization parameters, such as a minimum of 5 samples per leaf to reduce overfitting and ensure meaningful generalization.

Bagging-based ensemble learning was represented by the Random Forest regressor. In this approach, 300 regression trees are trained independently on bootstrap samples of the training data, with additional randomization introduced through feature subsampling at each split. The final prediction is obtained by averaging the outputs of all individual trees. Random Forests are known for their robustness, stability, and relatively low sensitivity to hyperparameter tuning, making them a strong baseline for heterogeneous environmental datasets.

Boosting-based ensemble methods represent the third category of evaluated models. In particular, two modern gradient boosting implementations were employed. The first is CatBoost (3000 iterations, learning rate of 0.05, depth of 8, and RMSE loss function), which constructs ensembles of decision trees in a sequential manner and is specifically designed to handle categorical attributes effectively. By employing ordered boosting and target-based encoding, CatBoost reduces prediction bias and mitigates overfitting when working with high-cardinality categorical features commonly found in real-world datasets.

The second boosting-based model is LightGBM, a highly efficient gradient boosting framework that employs histogram-based feature binning and leaf-wise tree growth strategies. This design enables LightGBM to capture complex nonlinear relationships and feature interactions with high computational efficiency. Model complexity was controlled through parameters such as 5000 boosting iterations, a learning rate of 0.03, a minimum of 30 child samples, and regularization terms to balance predictive accuracy and generalization.

All models were trained and evaluated using identical training and testing splits to ensure a fair comparison. Hyperparameters were selected based on recommended practices for each algorithm and validated through preliminary experimentation.

4.3 Evaluation metrics

The predictive performance of the implemented regression models was assessed using several commonly adopted evaluation metrics for continuous-valued targets. These metrics quantify different aspects of prediction error and together provide a comprehensive view of model accuracy and robustness.

The primary evaluation metric used in this study is the mean squared error (MSE), defined as

$$\mathrm{MSE} = \frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2 ,$$

where y_i denotes the true target value and $\hat{y}_i$ represents the corresponding model prediction. MSE penalizes larger errors more strongly due to the squared term and is therefore sensitive to outliers.

To provide an error measure expressed in the same units as the target variable, the root mean squared error (RMSE) is also reported. RMSE is obtained as the square root of MSE and is often easier to interpret in practical applications. Additionally, the mean

absolute error (MAE) is included as a complementary metric that measures the average magnitude of prediction errors without emphasizing extreme deviations.

Finally, the coefficient of determination (R^2) is used to evaluate the proportion of variance in the target variable that is explained by the model. The R^2 score provides a normalized measure of goodness-of-fit and enables comparison across different models and datasets.

The combination of these metrics allows for balanced evaluation of model performance, capturing both overall accuracy and sensitivity to large prediction errors. All evaluation measures are computed on the test dataset, which remains unseen during model training and parameter selection.

5 Results and discussion

In this section, we present and discuss the experimental results obtained from applying selected decision tree-based regression models to the United States Wind Turbine Database. The evaluated models include a single regression tree, a Random Forest ensemble, a CatBoost regressor, and a LightGBM regressor. Model performance is assessed using the evaluation metrics defined in Section 4.3, and interpretability is examined through feature importance analysis.

5.1 Quantitative results

Table 2 summarizes the predictive performance of all evaluated models in terms of mean squared error (MSE), root mean squared error (RMSE), mean absolute error (MAE), and the coefficient of determination (R^2). All results are reported on the held-out test dataset.

Table 2: Regression performance of tree-based models on the USWTDB dataset.

Model	MSE	RMSE	MAE	R^2
Decision Tree	141.25	11.88	0.72	0.99978
Random Forest	105.19	10.26	**0.71**	0.99983
CatBoost	83.34	9.13	1.33	0.99987
LightGBM	**77.50**	**8.80**	0.74	**0.99988**

The results clearly demonstrate the advantages of ensemble-based learning methods over a single regression tree. While the baseline decision tree achieves relatively high predictive accuracy, its performance is consistently worse than that of ensemble models across all error-based metrics except for MAE, where the Decision Tree (0.72) outperforms CatBoost (1.33).

Among the evaluated approaches, LightGBM achieves the best overall performance, obtaining the lowest MSE and RMSE as well as the highest R^2 value. CatBoost also performs strongly, outperforming both the single tree and the Random Forest ensemble in terms of squared error metrics. These results indicate that boosting-based methods are particularly effective in capturing complex nonlinear relationships present in the wind turbine dataset.

Interestingly, the Random Forest model, while robust and competitive, is outperformed by both boosting-based approaches in this experimental setting in RMSE and R^2 value. This observation suggests that the sequential error-correction mechanism used in boosting methods provides an advantage over variance reduction alone when modeling structured tabular data with strong dominant predictors.

5.2 Model comparison

A more detailed comparison of the evaluated models highlights the practical implications of different ensemble learning strategies. The Random Forest regressor exhibits stable and reliable performance, benefiting from bootstrap aggregation and feature-level randomization. Its relatively low MAE indicates strong robustness to moderate prediction errors, making it suitable for applications where stability and interpretability are prioritized.

In contrast, boosting-based models, particularly LightGBM, achieve higher performance in terms of squared error metrics. By constructing trees in a sequential manner and optimizing leaf-wise growth, LightGBM is able to model fine-grained interactions between numerical and categorical attributes more effectively than bagging-based ensembles. The improved performance of boosting methods reflects their ability to focus learning on hard examples and reduce residual errors iteratively.

While Random Forests remain a strong baseline for heterogeneous environmental datasets, modern gradient boosting implementations such as LightGBM can offer strong performance when properly configured.

5.3 Feature importance analysis

We conducted the feature importance to provide model interpretability for the Random Forest and LightGBM models. It should be noted that feature importance values are derived differently for each model.

For the Random Forest model, feature importance is based on impurity reduction aggregated across all trees in the ensemble. We found that rotor diameter is by far the most influential predictor, accounting for the majority of explained variance. This result is consistent with domain knowledge in wind energy systems, where the rotor diameter affects the energy capture potential of a turbine. Additional contributions are observed from the turbine model and manufacturer, reflecting design-specific characteristics, while temporal and geographical attributes play a secondary role.

In contrast, LightGBM computes feature importance based on split frequency and gain across leaf-wise tree growth. The resulting importance distribution highlights a broader set of influential features, including geographical coordinates, turbine identifiers, and temporal variables. This indicates that LightGBM exploits more complex interactions between spatial, technical, and contextual attributes when forming predictions.

Despite differences in importance ranking mechanisms, both models consistently identify turbine-specific technical attributes and categorical descriptors as key drivers of predictive performance. Overall, the results confirm that modern boosting-based methods, particularly LightGBM, offer state-of-the-art performance for regression tasks involving large-scale environmental and energy-related datasets.

6 Conclusion and future work

In this chapter, we present a systematic study of selected decision-tree-based machine learning methods applied to regression tasks in the context of environmental data analysis. The theoretical background of classical decision trees, ensemble learning techniques based on bagging and boosting, as well as fuzzy decision tree approaches, was complemented by an experimental evaluation using real-world data from the United States Wind Turbine Database.

The experimental results demonstrate that ensemble-based methods significantly outperform a single regression tree in terms of predictive accuracy and robustness. Random Forests provide a balance between performance and stability, while boosting-based models achieve higher accuracy by capturing fine-grained nonlinear relationships among turbine characteristics. Feature importance analysis further highlights the interpretability of tree-based models and confirms the dominant influence of technical turbine attributes on model predictions, in agreement with domain knowledge from wind energy systems.

Despite the strong performance of the evaluated models, we acknowledge several limitations. The analysis focuses on a static snapshot of turbine data and does not explicitly account for temporal dynamics or spatial dependencies. Moreover, the experimental setup prioritizes methodological comparison over exhaustive hyperparameter optimization, which may further improve predictive performance.

In our future work, we aim to explore extensions of the presented approach in several directions. These include the incorporation of fuzzy ensemble methods to better handle uncertainty and vagueness in environmental data, the integration of spatial and temporal modeling techniques, and the application of advanced explainability tools to further enhance model transparency. Additionally, the proposed methodology may be extended to other types of renewable energy datasets and environmental monitoring applications.

Data availability

Data are available from U.S. Wind Turbine Database [16], provided by the U.S. Geological Survey, American Clean Power Association, and Lawrence Berkeley National Laboratory via https://energy.usgs.gov/uswtdb

Acknowledgement

This work was supported by the VVGS ESGD grant (Early Stage Grants – Pavol Jozef Šafárik University in Košice), led by Ing. Manohar Gowdru Shridhara, ŠPP element 0HV040126 / FS 190180 / 2887 – VVGS ESGD – Gowdru, Faculty of Science, within the project "Early Stage Grants – Pavol Jozef Šafárik University in Košice", code 09I03-03-V05-00008. (M. Gowdru Shridhara).

References

1. Zhang, Y., Wang, J. X., & Wang, X. F. (2014). Review on probabilistic forecasting of wind power generation. *Renewable and Sustainable Energy Reviews*, *32*, 255–270. https://doi.org/10.1016/j.rser.2014.01.033
2. Abisoye, B. O., Sun, Y., & Wang, Z. (2024). A survey of artificial intelligence methods for renewable energy forecasting: Methodologies and insights. *Renewable Energy*, *221*, Article 100529. https://doi.org/10.1016/j.renene.2023.100529
3. Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The elements of statistical learning: Data mining, inference, and prediction*. Springer. https://doi.org/10.1007/978-0-387-84858-7
4. Breiman, L. (2001). Random forests. *Machine Learning*, *45*(1), 5–32. https://doi.org/10.1023/A:1010933404324
5. Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, *29*(5), 1189–1232. https://doi.org/10.1214/aos/1013203451
6. Zadeh, L. A. (1965). Fuzzy sets. *Information and Control*, *8*(3), 338–353. https://doi.org/10.1016/S0019-9958(65)90241-X
7. Breiman, L., Friedman, J. H., Olshen, R. A., & Stone, C. J. (1984). *Classification and regression trees*. Wadsworth International Group.
8. Breiman, L. (1996). Bagging predictors. *Machine Learning*, *24*(2), 123–140. https://doi.org/10.1007/BF00058655
9. Schapire, R. E. (1990). The strength of weak learnability. *Machine Learning*, *5*(2), 197–227. https://doi.org/10.1007/BF00116037
10. Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A. V., & Gulin, A. (2018). CatBoost: Unbiased boosting with categorical features. *Advances in Neural Information Processing Systems*, *31*, 6638–6648.
11. Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., & Liu, T.-Y. (2017). LightGBM: A highly efficient gradient boosting decision tree. *Advances in Neural Information Processing Systems*, *30*, 3146–3154.
12. Yuan, Y., & Shaw, M. J. (1995). Induction of fuzzy decision trees. *Fuzzy Sets and Systems*, *69*(2), 125–139. https://doi.org/10.1016/0165-0114(94)00229-Z
13. Pal, S. K., & Mitra, P. (2004). Case generation using rough sets with fuzzy representation. *IEEE Transactions on Knowledge and Data Engineering*, *16*(3), 292–300. https://doi.org/10.1109/TKDE.2003.1262181

14. U.S. Geological Survey. (2024). *United States wind turbine database (USWTDB)* (Version 7.1) [Data set]. https://eerscmap.usgs.gov/uswtdb/
15. U.S. Geological Survey. (2024). *United States wind turbine database – Attribute definitions (XML metadata)* (Version 7.1) [Data set]. https://data.usgs.gov/datacatalog/metadata/USGS.6001e327d34e592d8671fae0.xml
16. Hoen, B. D., Diffendorfer, J. E., Rand, J. T., Kramer, L. A., Garrity, C. P., & Hunt, H. E. (2025). *United States wind turbine database (USWTDB)* (Version 8.2, December 10, 2025) [Data set]. U.S. Geological Survey, American Clean Power Association, & Lawrence Berkeley National Laboratory.

About authors

Manohar Gowdru Shridhara is a PhD student at the Faculty of Science, Pavol Jozef Šafárik University in Košice. His research interests include machine learning and optimization techniques, mainly in the fields of energetics and wind farms.

Ľubomír Antoni is an associate professor at the Institute of Computer Science, Faculty of Science, Pavol Jozef Šafárik University in Košice. His research interests include artificial intelligence, fuzzy systems, data mining, and applied machine learning.

Gabriel Semanišin is a professor of Computer Science at Faculty of Science, Pavol Jozef Šafárik University in Košice. As part of his research activities, he focuses mainly on algorithmic graph theory and its application in various areas of theoretical and applied informatics. He is a co-guarantor of the study programs Applied Informatics, Data Analysis and Artificial Intelligence, and Computer Science. He was a supervisor of six PhD students in the study programs Computer Science, Discrete Mathematics and Theory of Teaching Informatics.