Penerbit UTHM

JOURNAL OF SOFT COMPUTING AND DATA MINING
e-ISSN: 2716-621X

JSCDM

Vol. 5 No. 2 (2024) 96-109
https://publisher.uthm.edu.my/ojs/index.php/jscdm

Permuted Gini Importance – PaP Impurity Measurement for Tree-Based Models

Ifra Altaf[1], Manzoor Ahmed Chachoo[1]*

[1] *Department of Computer Sciences*
University of Kashmir, J&K, Srinagar, 190006, INDIA

*Corresponding Author: c_manzoor@yahoo.com

DOI: https://doi.org/10.30880/jscdm.2024.05.02.008

Article Info

Received: 17 September 2024
Accepted: 22 November 2024
Available online: 18 December 2024

Keywords

Feature selection, gini impurity, feature importance, impurity measurement, random forest

Abstract

One of the various advantages of tree-based models is that they come with feature importance measures intended for feature ranking. The significant advantage of tree ensemble importance measures is that they ensure the impact of each and every predictor variable distinctly, including multivariate interactions with other predictor variables. But when correlation increases, both Gini and Permutation importance are incapable of detecting relevant variables. Also, Gini's importance is biased toward certain features. The main aim of this research study is to reduce the feature selection bias arising from the correlated features in ensemble models. In this paper, we put forward an optimized impurity measurement approach for tree ensembles that comprises permuting the features sequentially throughout the training so as to destroy their extrapolative influence with the target without changing their marginal distribution. Subsequently, the resulting change in impurity is recorded and averaged over all trees to adjust the feature importance. Permuting one of the features leads to model extrapolation. In tree ensembles, the extrapolation quality is generally low. Therefore, the error value is high, which forces our proposed method to put weight on its importance. Experimental results show that our proposed approach achieves higher computational efficiency. Jupyter Notebook was employed as a primary platform to conduct the analysis and experimentation.

1. Introduction

Feature importance is an important theory in machine learning that discusses the comparative importance of each feature in the training or in-bag dataset. Feature importance is the degree of the impact of all input variables to predict the output target variable. Feature selection can be classified into three categories: wrapper, filter, and embedded. The wrapper approach evaluates the feature scores by taking the prediction performance of the proposed model into account. The filter approach calculates the feature importance scores without depending upon the model performance. The embedded feature selection approach tries to combine the feature selection as well as the prediction accuracy of the model.

Feature selection using the feature importance measures for tree ensembles have been getting better attention in disease diagnosis to select a subset of diagnostic features pertinent for the prediction of a certain disease [1]. The tree ensembles have better computational speed with little to no parameter tuning. They also provide the efficient way of handling the missing values. But they also act as black boxes without providing many insights about the internal working of arriving at a predicting outcome.

In the tree-based models the two most used feature importance measure are Gini Importance (GI) [2] and Permutation Importance (PI). Gini Importance is a quantitative measure of feature importance for estimating the target variable using the in-bag samples. Permutation Importance gives the decrease in accuracy based on measurements of the model error, therefore unseen out-of- bag samples are best suited to test the model. Through all the trees of the ensemble, the two techniques capably measure the feature importance to estimate the value of the target variable. Both the techniques are unit-less.

Most of the researchers have considered only one of the two measures to evaluate the feature importance. Gini importance can give high importance to the high cardinality features. It is biased towards certain features that promote over fitting of the model and have less predictive control regarding the target variable [3]. The reason is that these impurity-based importances are computed on in-bag samples statistics so are not fully able to generalize the out-of-bag samples. The PI measure has certain disadvantages if the dataset contains considerable correlations between its features [4] [5]. In that case, if one of the features gets permuted then the model starts to extrapolate. In a complicated ensemble like the black-box model, the extrapolation quality is generally very low. Therefore, the error value will be high that forces permutation importance to put more weight on its importance.

The weak point, as indicated by the previous works, emphasizes the fact that in highly correlated datasets, PI leads to poor extrapolation and is computationally intensive, attributable to the logarithmic calculations, while GI does not generalize the out-of-bag samples very well because it is dependent upon the in-bag samples to calculate impurity reduction. In our proposed research, we emphasize the feature selection bias of GI by means of methodological outlooks. To improve its biased split, we propose another method, Permuted Gini – a predict and permute (PaP) feature importance method that determines the importance by calculating the total reduction in the node impurity successively after randomly permuting the feature values. Permuted Gini is a simple resolution to the problem of bias in correlated features. The contribution of our proposed research work:

- Introduction of a novel feature importance measurement for tree ensembles.
- The implementation of the Permuted Gini feature importance approach, which handles the inherent feature selection bias of GI.
- This method helps to alleviate the bias that is introduced by correlated features.
- This method improves the accuracy of feature importance calculations.

In this research work, together with our proposed feature selection method, we classify the disease datasets using tree ensembles to determine the empirical efficiency of our proposed method. The remaining part of this research paper is organized as follows: Section 2 briefly signifies the literature survey of various random tree ensemble feature selection techniques. Section 3 presents the methodology, while Sections 4 and 5 present the results and discussion, respectively. Section 6 concludes our paper with some references.

2. Related Work

In comparison to the univariate screening methods, the key advantage of tree ensemble importance measures is that they ensure the influence of each and every predictor variable separately in addition to in multivariate interactions with other predictor variables [6].

Lunetta et al. [7] compared the random forest algorithm with some univariate screening methods like the Fisher test to detect the interactions between variables and found that the former approach detects efficiently. However, the algorithm was not efficient in handling the high-dimensional data compared to the other methods used for comparison. Hothorn et al. [8] used a modified tree construction namely cforest to grow classification trees using a conditional inference framework. Their approach reduced the bias in feature selection thereby increasing the accuracy of the classification trees but the method is computationally intensive. Strobl et al. [6] performed empirical studies and showed that GI assigns systematically higher feature importance values to those numerical or categorical features that have many categories. The authors also presented that the cforest has less feature selection bias. The authors in their work listed the sources for the biases and provided the solution to deal with them too. The most imperative problem that emerged here is inherent to GI that it assigned upper importance values to those features which were having many categories. This resulted in skewed results.

Genuer et al. [9] [10] studied the impact of sample quantity, feature set and number of trees on PI along with the effect on the importance measure as well as the correlation. The authors were able to enhance the feature selection process as evident by their experimental results but the limitation of their methodological approach was that their generated results were specific to certain datasets and hence were not universally generalized. Nguyen et al. [11] put forward an optimal feature selection for support vector machines. The limitation of their optimal feature selection was regarding the inability not to generalize for different types of datasets. Louppe et al. [12] considered MDI for totally randomized trees and proved that GI of unrelated variables and noisy variables is zero and also proved that adding irrelevant variables do not impact the GI of relevant ones. This method was however not able to handle the noisy variables becoming the biggest drawback in their research. Louppe [13] proposed early stopping mechanisms by limiting the tree depth and having larger leaf sizes to lessen the feature selection bias. This affected the accuracy of the random forests; hence, it became the limitation of the study and decreased the performance.

Park et al. [14] pointed out a variable selection method based upon a sequential random k-nearest neighbor. The advantage of this method was that it worked effectively for high-dimensional data. The disadvantage of this method was that it did not work well for high-dimensional data, which had highly correlated features. Zhou, Zhou & Li [15] put forward a feature selection algorithm based upon the random forest to produce a low-cost feature subset where the experts estimated the real feature cost. This method ensured that only informative, relevant features got selected. The major disadvantage of this approach was that there was a bias due to the expert-based cost estimation.

However, [16] had earlier shown that the expert's elucidation could cause bias in epidemiological studies, but the study was not able to deal with all the complexities in other fields. Xiao Li [17] and Zhou et al. [18] put forward the de-biased and unbiased MDI feature importance respectively. The limitation associated with these was related to the level of complexities involved in their implementations. The former approach was sophisticated to understand and implement, while the latter required additional computational resources as compared to the other contemporary methods. In both approaches, the authors used the out-of-bag samples to perform the experiment of evaluating the decrease in impurity at each node. The advantage of these approaches however was in terms of MDI versions without bias. B´enard et al. [19] attempted the question of inconsistency of the PI and MDA, but they could not create general results for the GI consistency. Hence, the importance of using an impurity-based variable does not guarantee the selection of suitable features. Their solution was in terms of a practical one via Sobol-MDA but the inconsistency in MDA continued to be a challenge.

Ensemble randomized trees show high predictive accuracy [20] regardless of highly correlated variables. [21], [22], [23] have used ensemble trees for dimensionality reduction as well as to screen the feature set. Hapfelmeier et al. proposes to use an innovative variable selection technique for random forests but that required specific hyper parameter tuning for effective feature selection. Moreover, different datasets required different parameter settings. Similar approach was used by Ishak where the author compared the various feature selection methods with the help of random forests and support vector regression and random forest. The results however depended on a particular model or dataset. Yin et al. used an ensemble approach for feature selection which proved beneficial in terms of the performance but it face immense complexity while combining the two algorithms. There are only limited and sparse results on GI and PI from a theoretical perspective computed through tree procedures because the proofs are not straightforward at all.

From literature review it is clear that the random forests face the challenges with the high dimensional datasets. However the algorithm performs well in detecting the variable interactions effectively. Existing impurity measure for tree models lead to biased results due to many reasons that include overestimation of the features having many categories as well as difficulty with correlated features. Many research works have tried to reduce the bias and improved the accuracy of classification but these modified tree constructions are computationally intensive.

The researchers have worked with different approaches, such as using different sample sizes or different number estimators in tree ensembles to limit the bias in the feature selection. The review leads us to conclude that even though there are advanced tree construction methods recorded in the literature that help with reducing the bias and increasing the accuracy of classification but they come with a greater computational cost and vice-versa. As far as GI is concerned, the literature indicates that each time, the high GI values do not relate to the predictive relations between features and the result. This phenomenon is known as Gini Importance feature selection bias [17]. As it is dependent upon the in-bag samples that do not generalize well, the GI measurement may present the biased model's predictions by favoring specific features. The correct impact of features with respect to the model's predictions is not represented accurately. So, the features having high GI values don't always need to be the important ones. GI bias can lead to potential inaccuracies as well as diminish the overall performance.

The GI is further resource efficient as compared to PI in terms of the time because it does not involve the computation of logarithmic functions. Besides, PI faces problems when the features are highly correlated. In PI, the permutations of highly correlated features cause poor extrapolation which results in the inaccurate importance values. Table 1 gives a comparative overview of the feature importance measurement used for the tree ensembles, which includes a brief summary of the different methodologies used over time to measure feature importance.

Table 1 *Comparative overview of feature importances using tree ensembles*

Year	Author	Method	Dataset Used	Remark
2004	Lunetta et al. [7]	Univariate Screening Methods	Association Study Data (Large Scale)	Method did not efficiently detect interactions in large scale datasets.
2006	Hothorn et al. [8]	Conditional Inference Framework (cForest)	Disease Datasets including Breast Cancer Dataset	Reduced Feature Selection Bias
2007	Strobl et al. [6]	Gini Importance Measure	ILPD, Synthetic Datasets	Gini Importance feature selection bias
2008	Tuleau et al. [9]	Influence of Sample Quantity on Random Forest	PIMA, Wine and others	Could not generalize over all datasets
2010	Tuleau et al. [10]	Random Forest Variable Selection	Cancer Datasets	Efficient feature selection in high dimensional data
2010	Nguyen et al. [11]	SVM Feature Selection	PIDD, Iris and others	Improved performance with selected variables
2013	Louppe et al. [12]	Mean Decrease in Impurity for Random Forests	PIDD, Wine, Breast Cancer and others	Validated efficacy of Mean Decrease in Impurity (MDI) Gini Importance feature selection bias
2014	Louppe et al. [13]	Early Stopping Approach	PIDD, Wine, Breast Cancer and others	Presented theoretic insights
2015	Park et al. [14]	Sequential Random k-nearest Neighbor Feature Selection	Gene Expression Dataset	Improved accuracy for high-dimensional data
2016	Zhou et al. [15]	Cost-Sensitive Feature Selection Using Random Forest	Iris, Wine and others	Effective in terms of utility and cost
2019	Xiao et al. [17]	De-biased MDI Feature Importance Measure For Random Forests	Iris, Wine, PIDD, ILPD, and other benchmark datasets	Gini Importance feature selection bias
2019	Zhou et al. [18]	Unbiased Measurement of Feature Importance For Random Forests	UCI Adult Classification dataset	Need extra computational resources
2021	Benard et al. [19]	Sobol MDA and MDA	PIDD, German Credit Dataset	Improved consistency of the feature selection measurement Computationally intensive (computation of logarithmic functions)
2023	Yuhua et al. [23]	IGRF-RFE	UNSW-NB15 dataset	Increased the complexity of the model

3. Methods

3.1 Preliminaries

In the tree ensembles, the process of tree building is obtained by dividing the feature space recursively by selected splitting criteria and fitting a prediction function. If the input dataset consists of k inputs of in-bag samples and the response is denoted by z_i equaling to (x_i, y_i) for i = 1, 2 … n, it means xi = (x_{i1}; x_{i2}, …, x_{ik}) ,. Let P represent the data at a node m. Let sv and sp be the splitting variable and a splitting point, respectively then the resulting children node of this binary classification tree are:

$$P_1 = \{(x, y)|\ x_{sv}\} \leq sp \tag{1}$$

$$\mathrm{P_2} = \{(\mathrm{x}, \mathrm{y})|\ \mathrm{x_{sv}}\} > \mathrm{sp} \tag{2}$$

Gini Impurity is a quantifiable measure that is used to predict the probability that a randomly selected variable would be wrongly classified by a particular node. Since it indicates how the model diverges from a pure division, it is called impurity metric. In feature importance it approximates the target variable. Stated originally by Leo Breiman in 1984, the Gini impurity of the node in a tree is calculated as:

$$I_{node\_j} = 1 - \sum_{i=1}^{n} p_c^2 \tag{3}$$

where *n* is the total number of classes in the target variable and p_c represents the probability of choosing a data sample having class *c* [24].

The impurity degree ranges from 0 to 1. In terms of GI, it can be used to calculate the feature importance. Gini decrease is calculated by means of the average decrease in Gini every time when the tree in the ensemble model is split on that specific feature. The improvement in the split-criterion associated with each node split is naturally aggregated and recorded in the tree building procedure [25]. It relies on the tree structure intensely; therefore, it can be computed for trees only [26]. Greater Gini importance value corresponds to the greater feature importance score and vice versa.

Gini Importance also known as Mean Decrease in Impurity [2] is the total decrease in node impurity weighted by the proportion of samples reaching that node averaged over all that make up the forest [27]. The best split of any node 'm' in a tree is given by θ_m which splits at the variable *i*. The splitting results in two children nodes, represented as *l* and *r*. *I* represent the impurity function for classification. A decrease in impurity is denoted as:

$$\Delta(\theta_m^*) = \mathrm{g_m I(m)} - (\mathrm{g_l\, I(l)} + \mathrm{g_r\, I(r)}) \tag{4}$$

where g is the fraction of samples falling into each node, $\mathrm{g_m} = \frac{\mathrm{n_m}}{\mathrm{n}}$, $\mathrm{g_l} = \frac{\mathrm{n_l}}{\mathrm{n}}$ and $\mathrm{g_r} = \frac{\mathrm{n_r}}{\mathrm{n}}$. We sum up all the $\Delta(\theta_m^*)$ of non-terminal nodes where the split includes ith variable, to get the importance of the feature *I* in a single tree.

$$\mathrm{GI_i^T} = \sum_{\mathrm{m,i}\in\theta_m^*} \Delta(\theta_m^*) \tag{5}$$

For tree ensembles, where we have N number of base learners, the Gini importance can be calculated by taking the average across all the trees

$$\mathrm{GI_i^{ET}} = \frac{1}{\mathrm{N}} \sum_{\mathrm{n=1}}^{\mathrm{N}} \mathrm{GI_i^T} \tag{6}$$

The permutation importance, also known as the Mean Decrease in Accuracy [2] PI_i of feature fi of the tree ensemble can be computed by:

$$\mathrm{PI_i^{ET}} = \mathrm{a} - \frac{1}{\mathrm{N}} \sum_{\mathrm{n=1}}^{\mathrm{N}} \mathrm{a_{k,j}} \tag{7}$$

where *a* is the accuracy score of the model before permuting one of its features and $\mathrm{a_{k,j}}$ is the accuracy score on corrupted permuted data.

3.2 Proposed Method

The benefit of using GI and PI features is their capability to consider the interaction between features [28]. Being computationally efficient, GI has been extensively used in a wide range of applications.

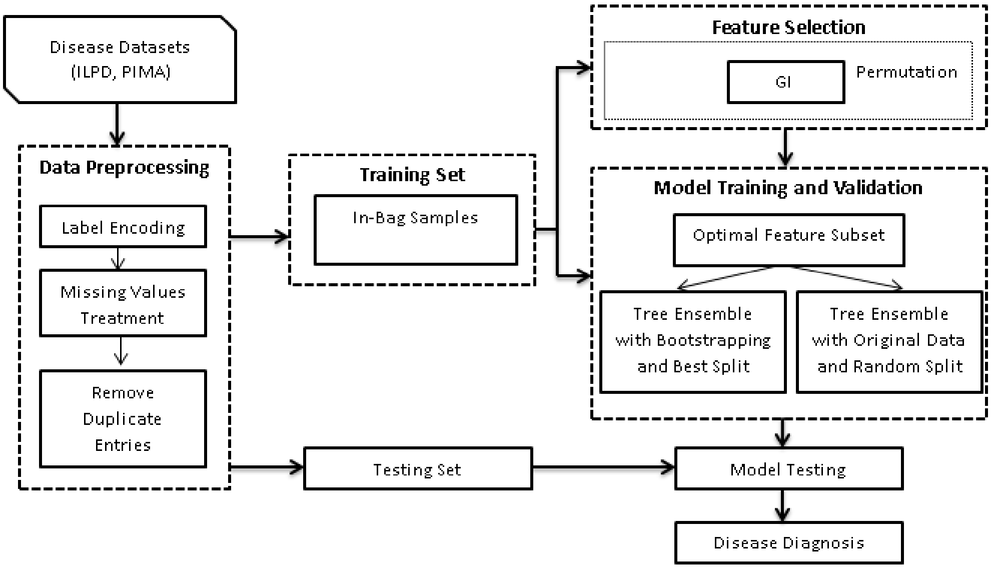


Fig. 1 *Proposed method*

To mitigate the GI feature selection bias we have applied a simple approach to correct its bias. We permute the features one by one while training the ensemble model. The relationship amongst the target feature and each feature in the in-bag dataset is destroyed by randomly shuffling or permuting the data of the specific feature taken into consideration. This removes the effect of the correlated features. When the n^{th} tree is grown, the i^{th} variable is randomly permuted in the in-bag samples and passed down the tree. The impurity after permuting is recorded and is averaged over all trees. This procedure gives the measure of the importance of variable i in the tree ensemble. The methodology proposed in the research work is depicted in Fig. 1.

Algorithm 1: Optimal Feature Selection – Permuted GINI

```
t[n]: trees base learners {n = 1 to N}
$x_i$: features in dataset {i = 1 to k}
T = Threshold
for features xi from i = 1 to n do
    for tree base learners t[m] ($x_i$) from n= 1 to N do
        N: all nodes in tree t[n] that use feature $x_i$ ($x_{i1}$; $x_{i2}$, ..., $x_{ik}$)
        for node m ϵ N do
            permute tuples of the ith feature
            Compute the score s as impurity decrease at m
            Weight the score s by number of tuples
            Add up these scores to S
        end for
    end for
    PGI = Average score S over all trees t[n] using the $x_i$ feature.
end for
Select features whose x importance is larger than threshold T to obtain an optimal feature subset.
```

Fig. 2 *Proposed PaP impurity measurement for tree algorithm*

For each permuted feature, we count the times this feature is utilized to split a node across all the trees in the ensemble, weighted proportionally by the number of samples that it splits. According to our proposed method:

$$\Delta'(\theta^*_m) = g_m I'(m) - (g_l I'(l) + g_r I'(r)) \tag{8}$$

where $I'(m)$is the impurity function at node m after permuting the ith column according to some mechanism. Similarly $I'^{(l)}$ and $I'(r)$ is for left and right two child nodes respectively. Suppose there are N base learners in the forest, the split improvement of xi in a tree ensemble can be defined as:

$$PGI_i^{ET} = \frac{1}{N}\sum_{n=1}^{N} GI\,'^{T}_{i} \tag{9}$$

$$PGI_i^{ET} = \frac{1}{N}\sum_{n=1}^{N}\sum_{m,i\in\theta_m^*}\Delta\,'\,(\theta_m^*) \tag{10}$$

Eq. (10) represents our proposed impulsive method to design a simple fix to solve the bias problem in PI by permuting the features. Our proposed method intends to give a measure of how each permuted feature adds to the similarity of node and leaves in the resulting tree ensemble. Higher the value of permuted Gini Importance, the higher is the importance of the feature in the ensemble. The permuted in-bag samples are used directly to assess the GI. Fig. 2 presents the proposed optimal unbiased feature selection approach.

Due to the shuffling of the features, results vary and randomness gets added to the feature importance measurement. By averaging the importance measures over repeated permutations evens out the measure. Although there is an increase in the computation time, every time the feature is shuffled, GI does not need model retraining. Also, reduced data for model training is meaningless for the feature importance. Threshold to select k number of features is kept entirely dependent upon the accuracy outcome during training phase of the model.

3.3 Dataset

In this research study, the datasets used are attained from Kaggle data-science community. The datasets consist of the Indian Liver Patient Dataset (ILPD) [29] and the PIDD Indians diabetes dataset (PIDD) [30]. The ILPD dataset was gathered from North East of Andhra Pradesh, India containing 583 observations having ten predictive features and one target output. PIDD contains 768 observations with eight predictive features and one target output. Table 2 and Table 3 give the statistical description of ILPD and PIDD respectively.

Table 2 *ILPD dataset description*

	Age	gender	TB	DB	ALP	SGPT	SGOT	TP	ALB	AG_RATIO	Target
count	583.00	583.00	583.00	583.00	583.00	583.00	583.00	583.00	583.00	579.00	583.00
mean	44.74	0.75	3.29	1.48	290.57	80.71	109.9	6.48	3.14	0.94	1.28
std	16.18	0.42	6.20	2.80	242.93	182.62	288.91	1.08	0.79	0.31	0.45
min	4.00	0.00	0.40	1.00	63.00	10.00	10.00	2.70	0.90	0.30	1.00
25%	33.00	1.00	0.80	0.20	175.50	23.00	25.00	5.80	2.60	0.70	1.00
50%	45.00	1.00	1.00	0.30	208.00	35.00	42.00	6.60	3.10	0.93	1.00
75%	58.00	1.00	2.60	1.30	298.00	60.50	87.50	7.20	3.80	1.10	2.00
max	90.00	1.00	75.00	19.70	2110.00	2000.00	4929.00	9.60	5.50	2.80	2.00

Table 3 *PIDD dataset description*

	preg	plas	pres	skin	insu	mass	pedi	age	Target
count	768.00	768.000000	768.000000	768.00	768.00	768.00	768.00	768.00	768.00
mean	3.84	120.89	69.10	20.53	79.79	31.99	0.47	33.24	1.34
std	3.36	31.97	19.35	15.95	115.24	7.88	0.33	11.76	0.47
min	0.00	0.00	0.00	0.00	0.00	0.00	0.07	21.00	1.00
25%	1.00	99.00	62.00	0.00	0.00	27.30	0.24	24.00	1.00
50%	3.00	117.00	72.00	23.00	30.50	32.00	0.37	29.00	1.00
75%	6.00	140.25	80.00	32.00	127.25	36.60	0.62	41.00	2.00
max	17.00	199.00	122.00	99.00	846.00	67.10	2.42	81.00	2.00

3.4 Data Processing

Our experiments are carried out in an open online cloud-based Jupyter Notebook environment Google Colab using Python 3 as a programming language. Data processing and visualizations were provided by *Scikit-learn, pandas, matplotlib*, *numpy*, and other packages. Label Encoding is used to transform categorical data into numerical ones.

The outliers were handled using a multivariate distance-based method, which separates the divergent data points on the basis of dissimilarity measures. The missing values were imputed by means of a multivariate imputation technique, which calculates every column's mean and uses it as a placeholder. A series of regression models are run sequentially to impute each missing value.

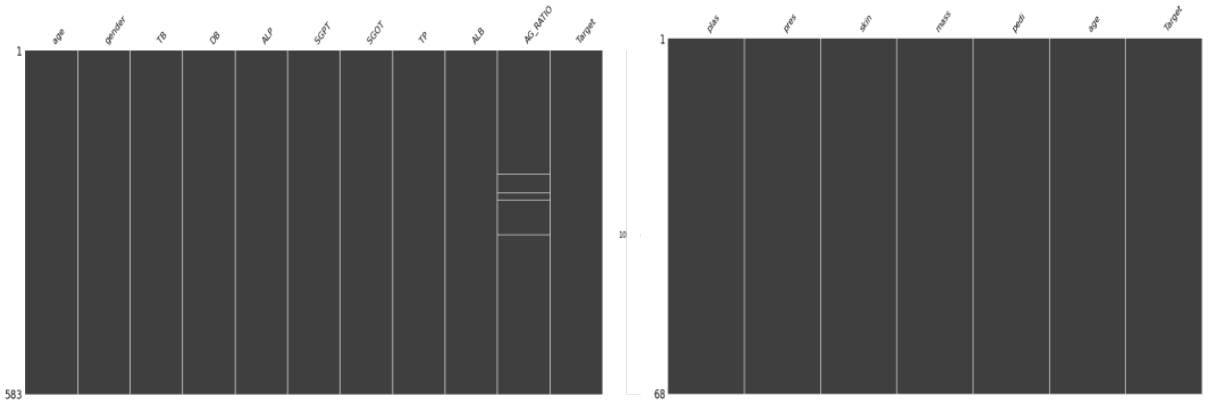


Fig. 3 *Missing values plot in ILPD and PIDD*

We used the *missingno* library of python to create a matrix in order to find a pattern of missing data values in the dataset. Fig. 3 depicts that missing values were only found in ILPD dataset.

3.5 Training

We divide the training set into two parts – in-bag samples and out-of-bag samples. The in-bag samples are used to compute the scores of features using GI and the out-of-bag samples are used to compute the scores of features using PI. Since the test and training data follow different distributions, in order to correctly estimate the training loss of the model, we construct the out-of-bag samples in such a manner that they have the same distribution as that of test data. For the Permuted GI ranking technique, we simply use the in-bag samples and test them on the testing data. The train – test ratio is taken as 70:30. Each time the number of features is reduced one by one, accuracy is calculated to get the optimal number of features (threshold) where the accuracy remains at peak. Eq. (6) and Eq. (7) are used to calculate the Gini importance and Permutation Importance for the tree ensembles respectively. We use Eq. (11) to implement our proposed Permuted Gini Importance ranking method. The unbiased results is easily extended to randomized as well as extra randomized forests as it is just an average across all the base learners used in the ensemble.

3.6 Evaluation Metric

As our work is a binary classification task and we are supposed to check whether our proposed approach works well with reaching a good fit, we use accuracy, mean square error (MSE), bias and variance. Accuracy is calculated by dividing the number of accurate predictions by the total number of predictions. Figure 4 shows the classification of machine learning errors.

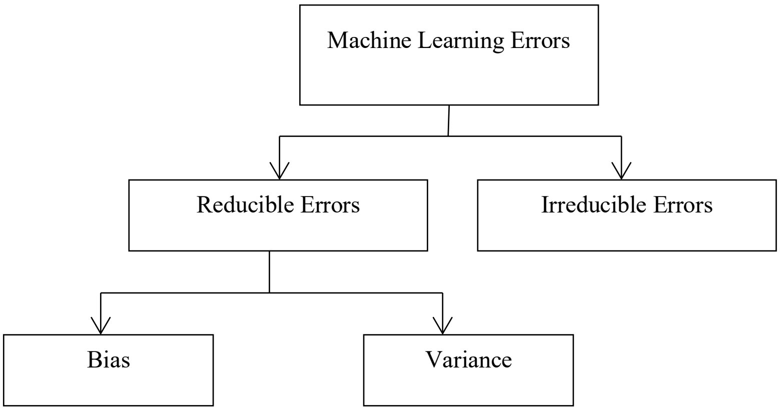


Fig. 4 *Classification of machine learning errors*

$$\text{Accuracy} = \frac{\text{TrueNegative+TruePositive}}{\text{TruePositive+FalsePositive+TrueNegative+FalseNegative}} \quad (12)$$

MSE is calculated as the average of the squared distance between the predicted and observed values. The formula for MSE is:

$$\text{MSE} = \frac{\Sigma(y_i - y_i')^2}{n} \quad (13)$$

where n is the number of data points in the dataset, y_I is the observed value, and y_i' is the predicted value. The reducible errors present in the machine learning model (Fig. 4) can be optimized. The reducible errors are divided into bias and variance. Bias can be defined as the incapability of machine learning models to learn the actual relationship between the data samples whereas variance measures the changes in the model while using different training sets.

$$\text{MSE} = \text{Bias}^2 + \text{Variance} \quad (14)$$

4. Results

There have been few feature selection techniques for tree models based upon both the in-bag and out-of-bag samples. Feature selection using feature importance ranking methods is not effective for all cases. The main motivation of this study is to propose a de-biased feature ranking technique that can help classify disease datasets efficiently.

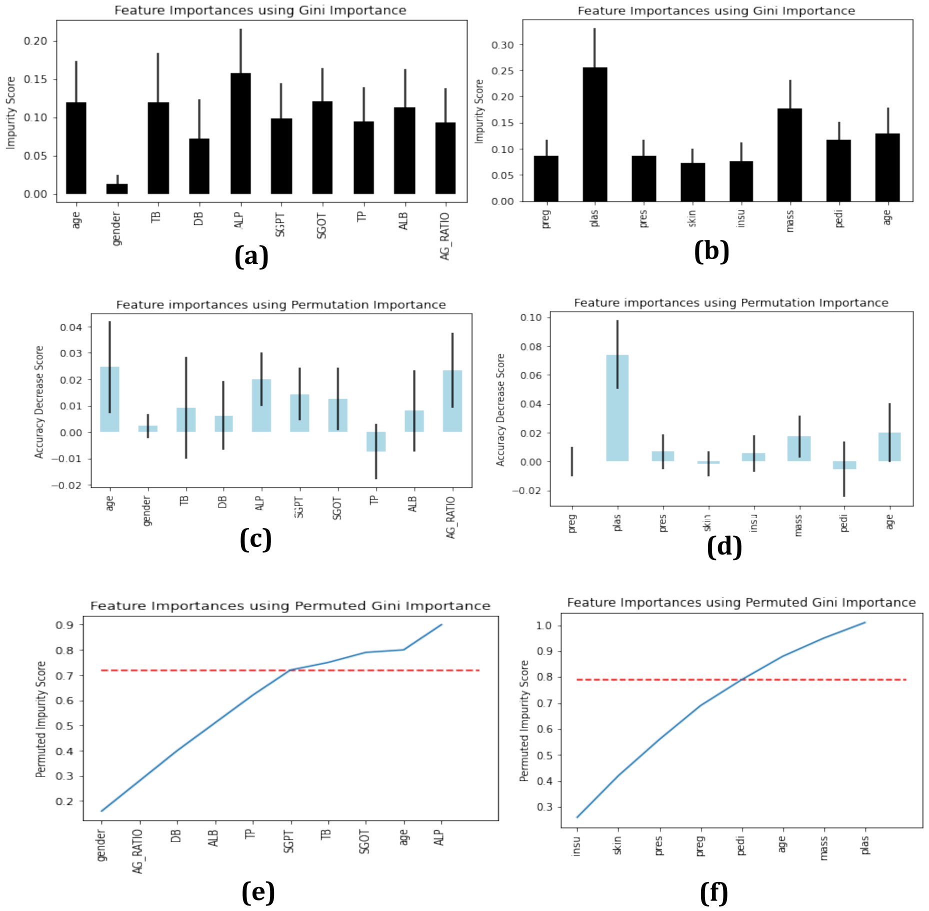


Fig. 5 *Significant features score using different types of feature ranking techniques (A), (C) and (E) represent feature importance ranking for ILPD, (B), (D) and (F) represent feature importance ranking for PIDD*

Fig. 5 gives the pictorial representation of various feature ranking techniques used in our experiment. In the proposed Permuted Gini Importance, the number of features to be taken depends upon the threshold (T). The threshold for the number of features to be taken is problem specific. Initially we included all the available features. Later we experimented with different number of features reducing them one by one and evaluated the impact on model performance in order to determine the optimal number of features for our scenario. Fig. 6 shows the performance of randomized trees with different number of feature count in feature subsets. Table 4 depicts the time taken to calculate the feature importance scores of the two datasets taken into consideration.

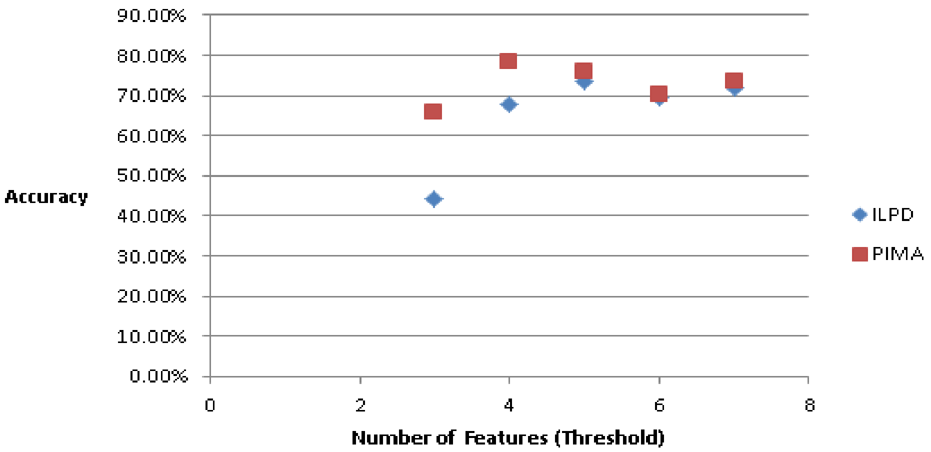


Fig. 6 *Accuracy calculation with different feature subset*

Table 4 *Elapsed time (seconds) to compute the feature importance using different feature subsets*

Dataset	GI	PI	Permuted GINI
ILPD	0.032	1.180	1.042
PIDD	0.040	1.120	1.035

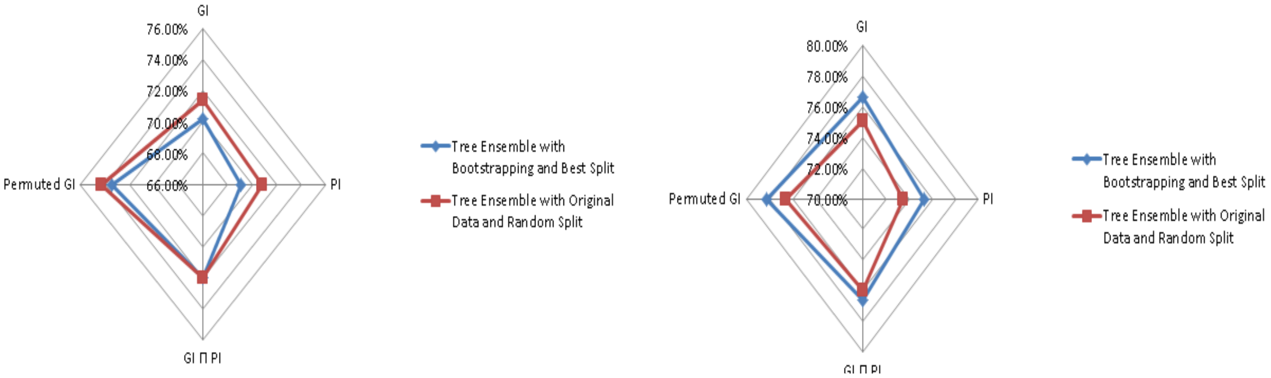
Table 5 presents the selected feature list using various feature selection techniques.

Table 5 *Selected features by various feature selection techniques*

Dataset	Feature Selection Technique	Selected Feature
ILPD	**GI**	alp, age, tb, sgot,alb,sgpt
	PI	age, tb, alp, sgpt, sgot,ag-ratio
	GI Π PI	age, tb, sgot,alp,sgpt
	PGI	sgpt, tb, sgot, age, alp
PIDD	**GI**	plas, mass, age, pedi, pres, preg
	PI	age, mass, plas, pres, insu, pedi
	GI Π PI	mass, plas, age, pres, pedi
	PGI	plas, mass, age, pedi

The different acquired subsets are given to tree ensembles with bootstrapping and best split to record the accuracy, bias, and variance for both datasets. Subsequently, the feature subsets are given to tree ensembles with original data and random split to record their accuracy statistics. Fig. 7 shows the accuracy comparison for ILPD and PIDD by means of the proposed feature selection method. In Table 6, the performance of our proposed method is presented along with other feature selection subsets.

Permuted Gini Importance gives the highest accuracy value in tree ensembles for both bootstrapped data with the best split and original data with a random split for both datasets. For ILPD, the accuracy obtained by our proposed method is the highest among all others (74.28%) using a tree ensemble with original data and a random split. For PIDD, the highest accuracy (78.30%) is achieved by using a tree ensemble with bootstrapped data with the best split.



(a) **(b)**

Fig. 7 *Accuracy comparison of different feature subsets on (a) ILPD; (b) PIDD*

5. Discussion and Comparison

GI gives the measure of the frequency at which any element from the dataset is mislabeled [31]. From the results (Table 4) it is apparent that the time taken for computing GI is less because at processor level log requires more computation than simple multiplication. The motivation behind GI is that it is computationally fast and less expensive since it does not require the computation of logarithmic functions [32]. It works better when the target variable is a binary variable [33]. GI allots high importance to several noisy features in the dataset that can lead to systematic bias and over fitting during feature selection. Permuting the features destroys the extrapolative influence of the feature without changing its marginal distribution.

Table 6 *Evaluation metrics of classification of disease datasets ILPD and PIDD*

Dataset	Feature Subset	Tree Ensemble with Bootstrapping and Best Split					Tree Ensemble with Original Data and Random Split				
		Accuracy	Bias	Variance	MSE	AEL	Accuracy	Bias	Variance	MSE	AEL
ILPD	GI	70.15%	0.224	0.078	0.303	0.302	71.42%	0.175	0.125	0.286	0.300
	PI	69.13%	0.223	0.079	0.314	0.302	70.82%	0.181	0.125	0.257	0.306
	GI Π PI	72.01%	0.223	0.077	0.291	0.300	72.00%	0.176	0.127	0.286	0.303
	Permuted GI	73.42%	0.221	0.077	0.269	0.299	74.28%	0.211	0.074	0.286	0.285
PIDD	GI	76.60%	0.165	0.062	0.216	0.226	75.04%	0.205	0.054	0.264	0.258
	PI	75.30%	0.199	0.049	0.247	0.248	73.50%	0.172	0.110	0.264	0.282
	GI Π PI	76.60%	0.164	0.061	0.234	0.225	76.01%	0.169	0.121	0.238	0.291
	Permuted GI	78.30%	0.167	0.061	0.203	0.228	76.62%	0.165	0.121	0.234	0.287

The compulsion of our proposed method is that when an unimportant feature is permuted, the impurity has a slight effect. Still, when an important feature is permuted, it results in a large increase in the impurity measure, which appears authentic in Table 7.

Table 7 *Accuracy comparison of other feature selection methods on ILPD and PIDD datasets*

Literature Work	Dataset	Classifier	Feature Selection Method	Number of Features	Accuracy
Singh et al. [34]	ILPD	Logistic Regression	Correlation-based Feature Selection	5	74.36%
Muheet et al. [35]	ILPD	Hard Voting Meta Classifier	Gini Importance	6	74.03%
Victor et al.[36]	PIDD	Naïve Bayes	Principal Component Analysis	5	77.83%
Caliskan et al.[37]	PIDD	Deep Neural Network	Auto-encoder	-	77.09%

In Table 7, we compare our results with the performance of various feature selection techniques that we used in our experiments. It is quite noticeable that our proposed method outperforms all other methods. We have also assessed the performance of feature selection techniques using separate GI and PI, and then we have used their intersection as well. The intersection increases the accuracy, but it is still low compared to our proposed method. We have also compared our approach's performance with that of the existing ones in Table 7.

6. Conclusion

Feature selection is a very beneficial preprocessing tool that aids practitioners in understanding the fundamental reasons for certain diseases. Gini needs to capture higher levels of feature interactions. Gini struggles more, it would seem. Permutation seems to capture importance better, although this difference is small. In our proposed method, after permuting the features one by one, we use GI to calculate the importance of each feature by taking the average or mean of a feature's total decrease in node impurity assessed by the probability of samples reaching that node in every single distinct decision tree in the ensemble forest. Hence, in the absence of input dependence and correlation, our proposed method computed with random tree ensembles provides a good assessment of the variable importance. This paper provides an innate basis for the flaws found in GI. The experimental results show that the suggested measurement provides a more efficient way for assessing the feature importance practically. In future we will use the theoretical and empirical implications of our proposed technique with other classification models.

Acknowledgement

The authors would like to show their appreciation to the Department of Computer Sciences, University of Kashmir.

Conflict of Interest

Authors declare that there is no conflict of interests regarding the publication of the paper.

Author Contribution

The authors confirm their contribution to the paper as follows: ***study conception and design:*** *Ifra Altaf, Manzoor Ahmad Chachoo;* ***data collection:*** *Ifra Altaf;* ***analysis and interpretation of results:*** *Ifra Altaf;* ***draft manuscript preparation:*** *Ifra Altaf. All authors reviewed the results and approved the final version of the manuscript.*

References

[1] Altaf, I., Butt, M. A., & Zaman, M. (2022, July). Machine learning techniques on disease detection and prediction using the hepatic and lipid profile panel data. In Congress on Intelligent Systems: Proceedings of CIS 2021, Volume 2 (pp. 189-203). Singapore: Springer Nature Singapore.

[2] Fayaz, S. A., Zaman, M., & Butt, M. A. (2022). Performance evaluation of GINI index and information gain criteria on geographical data: An empirical study based on JAVA and Python. In International Conference on Innovative Computing and Communications: Proceedings of ICICC 2021, Volume 3 (pp. 249-265). Springer Singapore.

[3] Kim, H., & Loh, W. Y. (2001). Classification trees with unbiased multiway splits. Journal of the American Statistical Association, 96(454), 589-604.

[4] Strobl, C., Boulesteix, A. L., Zeileis, A., & Hothorn, T. (2006, June). Bias in random forest variable importance measures. In Workshop on statistical modelling of complex systems. Citeseer.

[5] Strobl, C., Boulesteix, A. L., Kneib, T., Augustin, T., & Zeileis, A. (2008). Conditional variable importance for random forests. BMC bioinformatics, 9, 1-11.

[6] Strobl, C., Boulesteix, A. L., Zeileis, A., & Hothorn, T. (2007). Bias in random forest variable importance measures: Illustrations, sources and a solution. BMC bioinformatics, 8, 1-21.

[7] Lunetta, K. L., Hayward, L. B., Segal, J., & Van Eerdewegh, P. (2004). Screening large-scale association study data: exploiting interactions using random forests. BMC genetics, 5, 1-13.

[8] Hothorn, T., Hornik, K., & Zeileis, A. (2006). Unbiased recursive partitioning: A conditional inference framework. Journal of Computational and Graphical statistics, 15(3), 651-674.

[9] Genuer, R., Poggi, J. M., & Tuleau, C. (2008). Random Forests: some methodological insights. arXiv preprint arXiv:0811.3619.

[10] Genuer, R., Poggi, J. M., & Tuleau-Malot, C. (2010). Variable selection using random forests. Pattern recognition letters, 31(14), 2225-2236.

[11] Nguyen, M. H., & De la Torre, F. (2010). Optimal feature selection for support vector machines. Pattern recognition, 43(3), 584-591.

[12] Louppe, G., Wehenkel, L., Sutera, A., & Geurts, P. (2013). Understanding variable importances in forests of randomized trees. Advances in neural information processing systems, 26.

[13] Louppe, G. (2014). Understanding random forests: From theory to practice. arXiv preprint arXiv:1407.7502.

[14] Park, C. H., & Kim, S. B. (2015). Sequential random k-nearest neighbor feature selection for high-dimensional data. Expert Systems with Applications, 42(5), 2336-2342.

[15] Zhou, Q., Zhou, H., & Li, T. (2016). Cost-sensitive feature selection using random forest: Selecting low-cost subsets of informative features. Knowledge-based systems, 95, 1-11.

[16] Walter, S., & Tiemeier, H. (2009). Variable selection: current practice in epidemiological studies. European journal of epidemiology, 24, 733-736.

[17] Li, X., Wang, Y., Basu, S., Kumbier, K., & Yu, B. (2019). A debiased MDI feature importance measure for random forests. Advances in Neural Information Processing Systems, 32.

[18] Zhengze Zhou and Giles Hooker. Unbiased measurement of feature importance in tree-based methods. arXiv preprint arXiv:1903.05179, 2019.

[19] Bénard, C., Da Veiga, S., & Scornet, E. (2022). Mean decrease accuracy for random forests: inconsistency, and a practical solution via the Sobol-MDA. Biometrika, 109(4), 881-900.

[20] Fayaz, S. A., Zaman, M., & Butt, M. A. (2022). A hybrid adaptive grey wolf Levenberg-Marquardt (GWLM) and nonlinear autoregressive with exogenous input (NARX) neural network model for the prediction of rainfall. International Journal of Advanced Technology and Engineering Exploration, 9(89), 509.

[21] Hapfelmeier, A., & Ulm, K. (2013). A new variable selection approach using random forests. Computational Statistics & Data Analysis, 60, 50-69.

[22] Ben Ishak, A. (2016). Variable selection using support vector regression and random forests: A comparative study. Intelligent Data Analysis, 20(1), 83-104.

[23] Yin, Y., Jang-Jaccard, J., Xu, W., Singh, A., Zhu, J., Sabrina, F., & Kwak, J. (2023). IGRF-RFE: a hybrid feature selection method for MLP-based network intrusion detection on UNSW-NB15 dataset. Journal of Big Data, 10(1), 15.

[24] Menze, B. H., Kelm, B. M., Masuch, R., Himmelreich, U., Bachert, P., Petrich, W., & Hamprecht, F. A. (2009). A comparison of random forest and its Gini importance with standard chemometric methods for the feature selection and classification of spectral data. BMC bioinformatics, 10, 1-16.

[25] Hastie, T., Tibshirani, R., Friedman, J. H., & Friedman, J. H. (2009). The elements of statistical learning: data mining, inference, and prediction (Vol. 2, pp. 1-758). New York: springer.

[26] Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. Annals of statistics, 1189-1232.

[27] Breiman, L. (2017). Classification and regression trees. Routledge.

[28] Wright, M. N., Ziegler, A., & König, I. R. (2016). Do little interactions get lost in dark random forests?. BMC bioinformatics, 17, 1-10.

[29] https://www.kaggle.com/rahulrajpandey31/ilpd-indian-liver-patient-dataset-data-set, last accessed 2022/02/01

[30] https://www.kaggle.com/uciml/pima-indians-diabetes-database, last accessed 2022/02/01

[31] Han, H., Guo, X., & Yu, H. (2016, August). Variable selection using mean decrease accuracy and mean decrease gini based on random forest. In 2016 7th ieee international conference on software engineering and service science (icsess) (pp. 219-224). IEEE.

[32] Altaf, I., Butt, M. A., & Zaman, M. (2022). ETL for disease indicators using brute force rule-based NLP algorithm and metadata exploration. International Journal of Advanced Technology and Engineering Exploration, 9(90), 644.

[33] Kumar, A. (2016). Learning predictive analytics with Python. Packt Publishing Ltd.

[34] Singh, J., Bagga, S., & Kaur, R. (2020). Software-based prediction of liver disease with feature selection and classification techniques. Procedia Computer Science, 167, 1970-1980.

[35] Altaf, I., Butt, M. A., & Zaman, M. (2022). Hard voting meta classifier for disease diagnosis using mean decrease in impurity for tree models. Rev Comput Eng Res, 9(2), 71-82.

[36] Chang, V., Bailey, J., Xu, Q. A., & Sun, Z. (2023). Pima Indians diabetes mellitus classification based on machine learning (ML) algorithms. Neural Computing and Applications, 35(22), 16157-16173.

[37] Caliskan A, Yuksel ME, Badem H, Basturk A. Performance improvement of deep neural network classifiers by a simple training strategy. Eng Appl Artif Intell.2018;67:14–23