Leveraging Neural Networks and Calibration Measures for Confident Feature Selection

Hassan Gharoun, Navid Yazdanjue, Mohammad Sadegh Khorshidi, Fang Chen, *Member, IEEE*, and Amir H. Gandomi, *Senior Member, IEEE*

***Abstract*—With the surge in data generation, both vertically (i.e., volume of data) and horizontally (i.e., dimensionality) the burden of the curse of dimensionality has become increasingly palpable. Feature selection, a key facet of dimensionality reduction techniques, has advanced considerably to address this challenge. One such advancement is the Boruta feature selection algorithm, which successfully discerns meaningful features by contrasting them to their permutated counterparts known as shadow features. Building on this, this paper introduces NeuroBoruta, that extends the traditional Boruta approach by integrating neural networks and calibration metrics to improve prediction accuracy and reduce model uncertainty. By augmenting shadow features with noise and utilizing neural network-based perturbation for importance evaluation, and further incorporating calibration metrics alongside accuracy this evolved version of the Boruta method is presented. Experimental results demonstrate that NeuroBoruta significantly enhances the predictive performance and reliability of classification models across various datasets, including medical imaging and standard UCI datasets. This study underscores the importance of considering both feature relevance and model uncertainty in the feature selection process, particularly in domains requiring high accuracy and reliability.**

***Index Terms*—Neural networks, measurement uncertainty, feature selection, boruta, transfer learning, perturbation analysis, feature importance.**

I. Introduction

With the emergence of data centers and the advent of Big Data technologies in recent years, there has been a marked influence on the processes of data generation and storage. These advancements have acted as powerful enablers for high-throughput systems, substantially augmenting the capacity to generate data both in terms of the number of data points (sample size) and the range of attributes or features collected for each data point (dimensionality) [1]. The explosive surge in the volume of gathered data has heralded unprecedented opportunities for data-driven insights. AI technologies like machine learning (ML), and specially neural networks (NNs) are able to extract valuable insights from vast amounts of data.

High-dimensionality simultaneously poses distinct challenges that obstruct the success of ML algorithms. As data dimensions increase, irrelevant and redundant features may also be added to the data [2]. Moreover, as the number of dimensions in a dataset expands, the complexity of the information also tends to increase [3]. Therefore, more complex models are often required to handle the intricate information stemming from high-dimensional data [4]. Experiments have shown that more complex models can sometimes introduce challenges, leading to inconsistent predictions [5]. This inconsistent performance can be interpreted as uncertainty in the model's predictions. In ML, uncertainty typically stems from two primary sources: (I) data-related uncertainty, caused by inherent noise or variability in the data (aleatoric uncertainty), and (II) uncertainty related to the model itself, which occurs when the model's structure or parameters are not fully understood or properly captured (epistemic uncertainty) [6]. Thus, confidence can be understood as the inverse of the model's uncertainty. This issue is particularly concerning in the context of neural networks (NNs), which are known to produce overconfident predictions even when they are incorrect [7].

In high-stakes environments, decision-making systems aided by AI must not only be accurate, but also minimize uncertainty. This is increasingly important in the healthcare domain, where uncertain predictions risk guiding either a human operator or, in more severe scenarios, an automated controller, towards erroneous decisions [8].

In dealing with high-dimensional data, feature selection (FS) aims to identify the most significant subset of features, discarding those that are irrelevant or redundant. This process not only effectively reduces the dimensionality of datasets but also enhances the performance of classification tasks [9], [10]. Classic approaches to feature selection focus on improving predictive models in terms of accuracy. Accordingly, this paper introduces a novel FS method that aims to identify a subset of features which not only enhances model accuracy but also reduces model uncertainty.

The remainder of this paper is structured as follows: Next Section II reviews the related work. Section III offers a comprehensive discussion of the proposed algorithm. Section IV

Received 25 June 2024; revised 4 September 2024; accepted 4 November 2024. Date of publication 14 April 2025; date of current version 29 May 2025. *(Corresponding author: Amir H. Gandomi.)*

Hassan Gharoun, Navid Yazdanjue, Mohammad Sadegh Khorshidi, and Fang Chen are with the Faculty of Engineering & IT, University of Technology Sydney, Ultimo, NSW 2007, Australia (e-mail: hassan.gharoun@student.uts.edu.au; navid.yazdanjue@gmail.com; mohammadsadegh.khorshidialikordi@student.uts.edu.au; fang.Chen@uts.edu.au).

Amir H. Gandomi is with the Faculty of Engineering & IT, University of Technology Sydney, Ultimo, NSW 2007, Australia, and also with the University Research and Innovation Center (EKIK), Óbuda University, 1034 Budapest, Hungary (e-mail: gandomi@uts.edu.au).

This article has supplementary downloadable material available at https://doi.org/10.1109/TETCI.2025.3535659, provided by the authors.

Recommended for acceptance by N. Al Moubayed.

Digital Object Identifier 10.1109/TETCI.2025.3535659

details the dataset used and outlines the experimental design. The findings from the experiments are presented and analyzed in Section V. Lastly, Section VI provides concluding remarks and suggests avenues for future research.

II. Background

The curse of dimensionality, a term coined by Richard Bellman [11] which encapsulates the challenges faced in handling high-dimensional data spaces, is effectively addressed by employing a collection of techniques collectively referred to as dimensionality reduction. Dimensionality reduction can be categorized into two primary branches:

1) Feature extraction: the process of creating a smaller collection of new features from the original dataset while still preserving the majority of the vital information.
2) Feature selection: the process of identifying and choosing the most relevant features from the original dataset based on their contribution to the predetermined relevance criterion.

Feature selection, similar to ML models, is classified into supervised, unsupervised, and semi-supervised types, depending on the availability of well-labeled datasets. Furthermore, supervised feature selection is divided into four main subcategories, namely (interested readers in delving deeper into feature extraction and feature selection, and their various types, are encouraged to refer to [12]):

1) Filter methods: rank features based on statistical measures and select the top-ranked features.
2) Wrapper methods: evaluates subsets of features which best contribute to the accuracy of the model.
3) Hybrid methods: leverages the strengths of both filter and wrapper methods by first implementing a filter method to simplify the feature space and generate potential subsets, and then using a wrapper method to identify the most optimal subset [1].
4) Embedded methods: utilize specific ML models that use feature weighting functionality embedded in the model to select the most optimal subset during the model's training [13].

Random Forest (RF) is a widely used algorithm for embedded feature selection. The RF algorithm is a type of ensemble classifier that uses a concurrent set of decision trees, termed component predictors. RF applies a bootstrapping technique that randomly creates n training subsets from the main dataset, and this process is performed m times, leading to the construction of m independent decision trees. Each tree is built using a random subset of features. The ultimate decision is made based on the majority vote of the component predictors [14]. RF typically calculates feature importance using either the Mean Decrease in Impurity (MDI) or the Mean Decrease Accuracy (MDA) methods. MDI measures the importance of a feature by the total reduction in node impurity (e.g., Gini impurity or entropy) that it provides across all trees. MDA, on the other hand, assesses the importance of a feature by the decrease in model accuracy when the feature's values are randomly permuted.

Kursa [15] argued that the trustworthiness of the evaluation of feature significance is grounded in the presumption that the separate trees cultivated within the RF are unrelated while numerous analyses have occasionally demonstrated that this presupposition might not hold true for certain datasets. Furthermore, they contended that distinguishing genuinely important features becomes difficult when dealing with a large number of variables, as some may seem important due to random data correlations. Accordingly, the importance score by itself is inadequate to pinpoint significant associations between features and the target [15]. They address this issue by proposing the Boruta algorithm.

In RF, the importance of features is calculated in comparison to each other. However, in Boruta, the main idea is to evaluate the importance of features in competition with a set of random features called shadow features. In this process, every feature in the dataset is duplicated and their values are shuffled randomly. The RF algorithm is applied repeatedly, randomizing the shadow features each time and calculating feature importance based on MDA method for all attributes (original features and shadow features). Initially, Boruta used a statistical test to determine feature importance. In this approach, if the importance of a given feature consistently exceeds the highest importance among all the shadow features, it is classified as important. The measure of consistency is established through a statistical test based on the binomial distribution, which quantifies how frequently the feature's importance overtakes the Maximum Importance of the Random Attributes (MIRA). If this count (called 'hits') significantly outnumbers or undershoots the expected count, the feature is deemed 'important' or 'unimportant' respectively. However, Boruta also offers a simplified selection criterion. This approach considers features that have at least one instance where their importance score is higher than the maximum importance score of the shadow features. If a feature achieves this threshold at least once, it is deemed 'important.' This process iterates until all features are conclusively categorized or a predetermined iteration limit is reached.

Since the introduction of Boruta, this algorithm has been extensively and successfully utilized in across diverse research domains, including medicine [16], [17], [18], cybersecurity [19], engineering [20], [21], and environmental [22], [23], [24] studies. Even the Boruta algorithm has been successfully employed to reduce the dimensionality of features extracted from images by deep networks [25]. While the Boruta algorithm has indeed been successful in feature selection, contributing to improved predictive performance as highlighted in the literature, it's crucial to note that in Boruta, features are merely permuted. This permutation does not alter the inherent attributes of a feature. A similar phenomenon occurs in the RF algorithm when calculating feature importance through permutation. However, The relevance of the feature is determined by the data's characteristics, not its value [26].

Accordingly, the primary aim of the proposed method is the introduction of a method, built upon the Boruta methodology, which effectively selects a subset of features that not only enhances model accuracy but also reduces the model's predictive uncertainty.

Considering the significance of uncertainty in the proposed method, it is important to highlight that an examination of the literature on feature selection shows that the concept of uncertainty is frequently explored and utilized. However, the definitions of uncertainty in the literature vary. The concept of entropy as an indicator of uncertainty has been widely used in filter methods. Many entropy-based filter strategies have been proposed, utilizing measures like information entropy, rough entropy, and mutual information to evaluate feature relevance. However, the type of uncertainty these methods target primarily pertains to the information gain and redundancy among features, rather than model predictive uncertainty. For instance, Symmetrical uncertainty has been widely used to evaluate the dependency between features and the target variable, aiming to reduce redundancy by selecting features that maximize information gain. For instance, [3] proposed a filter method using symmetrical uncertainty, while [27] introduced an M-Cluster feature selection (Mcfs) method using Symmetrical Uncertainty (SU) to enhance classification accuracy in medical datasets. [28] leveraged Symmetric Uncertainty (SU) to evaluate the relevance and redundancy of class-independent features, and [29] developed a graph-based filter method using symmetric uncertainty to visualize and rank features, enhancing feature selection in high-dimensional datasets.

Another group of methods utilizes entropy-based measures. [30] developed a composite entropy-based method that combined fuzzy set theory with entropy measures to evaluate feature relevance, and [31] presented an optimization-based filter method leveraging conditional mutual information to select features. [10] proposed the Uncertainty Change Ratio (UCR), combining conditional mutual information (CMI) and conditional entropy (CE) to measure feature importance, and [32] presented a framework using intuitionistic fuzzy entropy (IFE) to handle uncertainty in datasets.

Neighborhood mutual information and entropy measures have also been employed to address information uncertainty. [33] used neighborhood mutual information (NMI) combined with a forward greedy search to assess feature relevance, particularly in medical datasets, and [2] combined self-information measures with neighborhood rough sets to evaluate feature relevance. [34] utilized neighborhood entropy-based uncertainty measures for classifying gene expression data, incorporating neighborhood entropy (NE), decision neighborhood entropy (DNE), and neighborhood mutual information (NMI).

Additionally, semi-supervised and hybrid methods have been proposed. [35] introduced a semi-supervised feature selection method that iteratively selected the most informative pairs of data points, updating the similarity matrix and ranking features based on their ability to preserve must-link and cannot-link constraints. [36] proposed an uncertainty optimization-based feature subset selection model using rough set theory to minimize uncertainty in feature subsets, and [37] introduced a novel method for feature selection in a three heterogeneous information system (3HIS) using rough set theory and various uncertainty measures.

Recently, research has begun incorporating uncertainty metrics into wrapper methods to enhance feature selection. The type of uncertainty these methods target is typically related to the predictive uncertainty of the model, which differs from the information uncertainty addressed by filter methods. [38] utilized rough set theory to identify features that contributed to high certainty in predictions while filtering out those that added to uncertainty, thereby optimizing feature selection to enhance the accuracy and efficiency of stock market predictions. [39] introduced a feature selection method combining SU with Ant Colony Optimization (ACO), using a probabilistic sequence-based graph representation to enhance the selection of informative features and reduce redundancy. Similarly, [40] proposed an ensemble feature selection framework that combined SU with a Multi-Layer Perceptron (MLP) to enhance the classification of sonar targets. This method ranked features based on SU values, trained multiple MLP models on different subsets, and combined their predictions through an ensemble voting mechanism to address information uncertainty and feature relevance.

[41] introduced a feature selection method based on Bayesian learning to address the specific needs of healthcare data, focusing on reducing uncertainty for a single target of interest. This approach used Bayesian confidence measures to evaluate features based on their contribution to improving model confidence for the target class. The paper defined model confidence for a specific target as the average precision over multiple iterations, calculated by evaluating the accuracy of the model's predictions for a target class in each iteration. This approach provided a measure of the model's reliability in predicting the specific target without directly quantifying uncertainty in a probabilistic sense. [42] presented an enhanced version of the Instance-wise Variable Selection (INVASE) algorithm, called Uncertainty-aware INVASE, to improve predictive confidence in healthcare applications like breast cancer diagnosis. This model introduced an uncertainty quantification module and a reward shaping module, modifying the Predictor Network to output both a mean (predicted value) and a variance (uncertainty measure). The model was trained using a specialized loss function to balance prediction accuracy and uncertainty estimation.

The above reviews reveal the following limitations:

- The evaluation of feature significance in RF relies on the assumption that the trees are independent, but many studies have shown this assumption may not always hold true [15].
- In Boruta, features are simply permuted, with their relevance determined by the data's characteristics rather than their values. If the data is poorly distributed or exhibits inherent patterns, such as frequently repeating limited unique values, multicollinearity between original and shadow features can occur. This may result in misleading conclusions about the importance or predictiveness of the shadow features.
- Uncertainty measures in filter methods primarily target the reduction of information uncertainty and redundancy among features, not predictive uncertainty. Moreover, filter methods—typically employing statistical measures like information gain and entropy—are independent of predictive models, and consequently, they do not account for how the selected features will interact with a specific model in terms of predictive uncertainty, particularly epistemic uncertainty.

- While some recent feature selection methods address predictive uncertainty, model confidence has often been overlooked. Even the confidence introduced in [41] does not provide a probabilistic measure of prediction confidence.

Accordingly, this study proposes a novel feature selection method designed to enhance both epistemic and aleatoric prediction uncertainty. Epistemic uncertainty, which stems from the model's lack of knowledge about its structure and parameters, can be influenced by the selection of features, since high dimensionality potentially exacerbating this uncertainty by adding complexity to the model's structure. Aleatoric uncertainty, arising from inherent noise in the data, is also considered. To address these challenges, our approach enhances the Boruta algorithm by replacing the RF model with NNs, which are better suited to capturing complex relationships in the data and mitigate the risk associated with the lack of independence between trees in RF. Additionally, noise is introduced to the shadow features to simulate aleatoric uncertainty, ensuring that the competition between original and shadow features accounts for this uncertainty. The proposed method is designed to iteratively select features that, even in the presence of noisy shadow features, help to enhance the model's performance by improving both prediction accuracy and reducing predictive uncertainty, ultimately increasing the model's confidence in its predictions. Additionally, this new design incorporates calibration metrics—which assess the model's confidence in its predictions—thereby elevating the focus of conventional wrapper methods from solely error-based performance to a more comprehensive evaluation that includes both accuracy and confidence-based assessment.

III. Methodology

The method proposed in this paper extends conventional Boruta feature selection by incorporating noise augmentation into the creation of shadow features, substituting the RF feature importance method typically used in Boruta with NNs and perturbation techniques. Additionally, it advances beyond merely assessing accuracy for feature importance, incorporating calibration metrics to enhance the evaluation process. In the subsequent sections, initially, the conventional Boruta feature selection is reviewed to establish a baseline, followed by a detailed presentation of the proposed methodology.

A. Preliminary: Introduction to Boruta

The Boruta algorithm is an effective, straightforward technique for selecting the most relevant subset of features in a dataset. It achieves this by performing a comparative analysis with synthetically created shadow features. Here's a brief outline of the process:

- *Creation of Shadow Features:* The algorithm begins by duplicating each feature within the dataset. These duplicates are then randomly shuffled to create shadow features, ensuring they mirror the structure but not the exact sequence of the original features.
- *Importance Assessment Using RF:* Both original and shadow features undergo an evaluation to determine their importance, which is carried out repeatedly through the RF model.
- *Comparison and Iteration:* Each original feature's importance is compared against the highest score obtained by any shadow feature in each iteration. An original feature is considered significant if it consistently surpasses the best shadow feature in terms of importance over several iterations.
- *Iterative Process:* The algorithm iteratively refines the comparison, aiming to isolate the features that demonstrate real predictive capabilities.
- *Feature Selection:* At the end of this process, any feature that consistently shows greater importance than the corresponding shadow features is selected as significant and kept for further model development.

The method outlined in the following sections builds upon the original concept of Boruta algorithm.

B. Proposed Method: NeuroBoruta

The proposed method enhances the Boruta algorithm in several significant ways: Firstly, it enhances the creation of shadow features by augmenting them with noise. Secondly, it replaces the RF and its embedded feature importance method with NNs and perturbation analysis. Lastly, it elevates the feature importance metric from solely focusing on accuracy to include both accuracy and calibration metrics. Each of these strategies is elaborated in the following.

1) Noise-Augmented Shadow Features: Shadow features in conventional Boruta algorithm bearing the same characteristics as the original ones, even considering the permutation of these shadow features disrupts the original relationship with the target variable. However, in predictive modeling, the overall characteristics of the dataset are often considered more insightful than the value of any single feature [26], while in the Boruta algorithm each original feature competes with new generated features mirroring their own characteristics.

Let's consider one feature denoted as X, its shadow feature X' can be defined as $X' = \text{shuffle}(X)$, where shuffle represents a random permutation of the entries in X. Although the order of data points in X' is randomized, the overall statistical distribution remains the same as X. This could lead to scenarios where some statistical relationship (like correlation) between X and X' exists simply due to the similar distribution of values, even though the logical or causal relationship intended in the model is disrupted.

To assess the potential for multicollinearity, consider the correlation coefficient between X and X'. The correlation coefficient ρ between two variables X and X' using the Pearson correlation coefficient is given by:

$$\rho_{X,X'} = \frac{\text{cov}(X, X')}{\sigma_X \sigma_{X'}} \tag{1}$$

Where:

- $\text{cov}(X, X')$ is the covariance between X and X',
- σ_X and $\sigma_{X'}$ are the standard deviations of X and X', respectively.

The covariance $\text{cov}(X, X')$ can be calculated as:

$$\text{cov}(X, X') = \mathbb{E}[(X - \mu_X)(X' - \mu_{X'})] \tag{2}$$

where μ_X and $\mu_{X'}$ are the means of X and X', respectively. Given that X' is a shuffled version of X, the means μ_X and $\mu_{X'}$ will be equal, and the standard deviations σ_X and $\sigma_{X'}$ will also be equal because shuffling does not change the distribution of the data.

However, calculating this directly for a permuted version is not straightforward because permutation disrupts the pairing of the values that contributes to the covariance. For a sufficiently large dataset and truly random permutation, the theoretical expectation of $\text{cov}(X, X')$ should be zero because each pair of values (x, x') from X and X' would likely be unrelated. However, if X is not well-distributed, or has inherent patterns such as limited unique values that repeat frequently, multi-collinearity among X and X' might be observed. This can lead to misleading conclusions about shadow features importance or predictiveness.

This potential issue is addressed by adding a scaled noise component to the original features. This ensures that the shadow features will differ marginally from their originals, reducing the likelihood that they'll maintain same statistical distributions.

In the context of NeuroBoruta, the introduction of noise into shadow features serves a dual purpose. Firstly, it deviates shadow features slightly from the originals in terms of their distribution, as discussed previously. Secondly, and crucially, it aligns with the goal of evaluating calibration metrics, which is discussed in the next section.4 In ML and deep learning, noise is recognized as a fundamental source of uncertainty that can affect model predictions. Calibration metrics, which assess how well the probabilistic predictions of a model correspond to actual outcomes, are particularly sensitive to the integrity of the feature set in the presence of noise. By incorporating noise into the shadow features, NeuroBoruta effectively simulates more challenging and realistic scenarios where features must demonstrate not only predictive accuracy but also robustness in maintaining calibration. This approach ensures that selected features are not only impactful but also reliable under varying conditions, enhancing the overall confidence in the model's predictions. Thus the use of noise-augmented shadow features complements the calibration focus of NeuroBoruta, providing a stringent benchmark for feature selection.

Algorithm 1 clearly outlines the steps involved in the generation of noise-augmented shadow features. In this approach, each original feature undergoes a process of augmentation with a factor of white noise – a random value possessing zero mean and standard deviation equal to that computed from the original feature. Subsequently, a random permutation is applied.

2) Uncertainty-Aware Feature Importance Via NNs Perturbation Analysis: The concept of perturbation analysis offers a solution to quantify the influence of each variable within the framework of NN models. In the procedure, perturbations are intentionally introduced to the NN's inputs. To maintain control over the experiment, only one input variable is altered during each iteration, keeping the remainder unchanged. The variable (in other words feature) that, when disturbed, yields the most significant impact on the dependent variable (in other words target or output) is then recognized as the variable of greatest importance [43].

Algorithm 1: Noise-Augmented Shadow Features.

```
Let D be the set of all features
for each feature f in F do
    δ_f ← std(f)
    Noise_f ← N(0, δ_f)
    Shadow_f ← shuffle(f + C × Noise_f)
end for
return D_NS ← set of noise-augmented shadow features
```

In this paper, the impact on the dependent variable is assessed using both accuracy and calibration metrics. Given the biased performance of accuracy metrics in imbalanced datasets, the F1 score is selected to gauge the impact in terms of accuracy. Among calibration metrics, the Brier score and Expected Calibration Error are chosen to further assess the impact in terms of uncertainty.

The Expected Calibration Error (ECE) is a widely recognized metric for assessing the calibration accuracy of probabilistic models. In practice, ECE (3) is computed by dividing the dataset into M bins and evaluating the absolute difference between the observed frequency of the positive class in each bin and the mean predicted confidence for that bin. This difference is then weighted by the proportion of the total observations that fall into each bin, providing a weighted average of these discrepancies across all bins [44]. In simple words, ECE is the average discrepancy between the observed accuracy and the predicted probability within each of M defined buckets [45].

$$ECE = \sum_{m=1}^{M} \frac{|B_m|}{n} |Acc(B_m) - Conf(B_m)| \tag{3}$$

$$Acc(B_m) = \sum_{m=1}^{M} \frac{1}{|B_m|} \mathbf{1}(\hat{y}_i = y_i) \tag{4}$$

$$Conf(B_m) = \sum_{m=1}^{M} \frac{1}{|B_m|} \tag{5}$$

where B_m is the number of predictions in bucket m, n is the total number of data points, $Acc(B_m)$ (4) and $Conf(B_m)$ (5) are the accuracy and confidence of bucket m, respectively. Also $\mathbf{1}(\cdot)$ is the indicator function. A lower ECE value indicates better calibration, with a value of 0 representing perfect calibration, where the predicted probabilities precisely reflect the observed frequencies.

Brier score (BS) defined as the average gap between predicted probabilities and ground truth and demonstrates the accuracy of predictions in probabilistic form [46]. Brier score is formulated as (6):

$$BS = \frac{1}{N} \sum_{i=1}^{N} (p_i - o_i)^2 \tag{6}$$

where N is the total number of samples, p_i is the predicted probability for each sample, and o_i is the actual outcome for each sample. A model demonstrating a lower Brier Score indicates better calibration.

The ECE and Brier score complement each other by providing two distinct but equally important perspectives on the quality of probabilistic predictions. ECE measures how well a model's predicted probabilities align with actual outcomes. For instance, if a model assigns a 70% probability to a particular outcome, the ECE evaluates whether that outcome occurs approximately 70% of the time across all instances where the model makes such a prediction. This assessment ensures that the predicted probabilities accurately reflect the true likelihood of outcomes, thus measuring the model's calibration.

On the other hand, Brier score measures the accuracy of probabilistic predictions by evaluating not just whether the prediction is correct, but also how close the predicted probability is to the actual outcome. For example, if a model assigns an 80% probability to a specific outcome and that outcome occurs, the Brier score will indicate that the prediction was well-calibrated. Conversely, if the model assigns an 80% probability to an outcome and it does not occur, the Brier score penalizes this prediction, with the penalty proportional to the difference between the predicted probability and the actual outcome.

By selecting these two metrics, the proposed method ensures that the feature selection process not only enhances predictive accuracy but also improves the model's confidence.

3) Neuroboruta: Considering the integration of NNs into the methodology, the proposed method is hereafter referred to as NeuroBoruta for ease of reference. Algorithm 2 offers a step-by-step delineation of the proposed method.

Consider a dataset, $D = \{(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)\}$, where x_i represents the i^{th} observation vector in a d-dimensional feature space, and y_i corresponds to the label of the i^{th} observation. The first stage involves the creation of training and testing datasets, denoted by D_{train} and D_{test}, respectively.

In this proposed variant, D_{train} is solely used for feature selection, while D_{test} is reserved exclusively to evaluate the performance of the selected features. Thus, the feature selection process does not have any access to or influence from the test dataset, thereby ensuring an unbiased assessment of the feature selection process.

Given D_{train} a new train set $\mathcal{D}'$ is constructed by a combination of original features and their noise-augmented counterparts (shadow features). This set is then normalized to prepare it for a shallow neural network. The shallow ANN model is trained on this normalized dataset. Once the model is trained, it evaluates three metrics: the F1 score, Brier score (BS), and ECE.

After training, each feature in $\mathcal{D}'$ undergoes perturbation analysis. Here, individual features are perturbed by adding noise scaled to a factor of their standard deviation $(C * \sigma)$, and the network's performance is re-evaluated. The change in performance metrics due to perturbation—specifically, the reduction in F1 score and increases in BS and ECE—are calculated for each feature.

To determine the significance of the original features in D_{train}, the maximum delta values among the shadow features are established as thresholds. If the performance degradation of an original feature exceeds these thresholds, it is marked as a 'hit,' indicating its significant role in model performance.

The iterative process repeats for a predefined number of iterations ($maxIter$), allowing for multiple evaluations and adjustments to the model and feature set. At the end of these iterations, the features that have accumulated at least one hit are selected.

Algorithm 2: Proposed Method: NeuroBoruta.

Let D_{train} be the train set with feature set F
Let D_{NS} be the set of noise-augmented shadow features
Let $\mathcal{H}$ be the empty list to store the hit history
Let $maxIter$ be the maximum number of iterations
Require: Shallow ANN, D_{train}, D_{NS}, $maxIter$
for $iter = 1$ to $maxIter$ **do**
  Create $\mathcal{D}' = D_{train} \cup D_{NS}$
  Normalize $\mathcal{D}'$
  $Model \leftarrow$ Shallow ANN
  Train the model on the dataset $\mathcal{D}'$
  Compute $F1S, BS, ECE \leftarrow$ the training f1 score, Brier score, and ECE of model on $\mathcal{D}'$
  for each feature f in $\mathcal{D}'$ **do**
    Perturb & shuffle feature f by adding $(C * \sigma)$ while keeping other features unchanged
    Compute $F1S'_f, BS'_f, ECE'_f \leftarrow$ the f1 score, Brier score, and ECE of trained model on $\mathcal{D}'$ with perturbed feature f
    Compute decrease in f1 score: $F1S''_f \leftarrow \max(F1S - F1S', 0)$
    Compute increase in Brier score and ECE: $BS''_f \leftarrow \max(BS' - BS, 0)$ $ECE''_f \leftarrow \max(ECE' - ECE, 0)$
  end for
  Normalizing every $F1S''_f$, BS''_f, ECE''_f via *Min-Max* scaler method
  $F1S_{MaxShadow} \leftarrow \max(F1S'')$ among D_{NS}
  $BS_{MaxShadow} \leftarrow \max(BS'')$ among D_{NS}
  $ECE_{MaxShadow} \leftarrow \max(ECE'')$ among D_{NS}
  for every feature f in D_{train} (original feature) **do**
    if $FS''_f > I_{Maxshadow}$ or $BS''_f > I_{Maxshadow}$ or $ECE''_f > I_{Maxshadow}$ **then**
      Add a hit to $\mathcal{H}_f$, the hit history for feature f
    end if
  end for
end for
return The set of features with at least one hit

IV. Experimental Setup

A. Data Sets

This study utilizes the FracAtlas dataset [47], comprises 4,083 X-ray images gathered for analyzing musculoskeletal injuries (i.e., fracture). The dataset includes a broad age range of patient, from 8 months to 78 years old, to accommodate variations

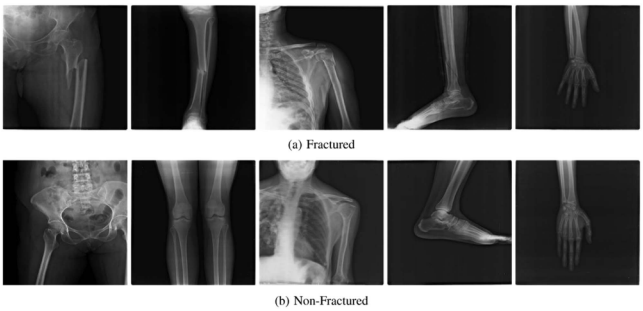


Fig. 1. Examples of the X-ray images from FracAtlas dataset.

in bone structure that could affect fracture analysis. It was noted that younger subjects might display features in their bone structure that resemble fractures due to lower bone density, while older subjects could have rough bone surfaces misinterpreted as fractures by analytical models. Gender distribution within the dataset shows 62% male and 38% female representation. Notably, the dataset includes 713 images categorized as 'Fractured' and 3,366 images classified as 'Non-fractured.' Analyzing the subset of images that show fractures, the gender distribution is markedly different: 85.4% of these abnormal images are from male and 14.6% are from female patients. The dataset encompasses detailed imaging of various body parts, with 1,538 hand scans, 2,272 leg scans, 338 hip scans, and 349 shoulder scans across multiple views [47]. Fig. 1 shows samples of FracAtlas dataset.

B. Transfer Learning

Training a deep learning model with convolutional layers from image data typically demands a substantial dataset and involves computationally intensive parameter tuning. Given the limited number of images in the FracAtlas dataset, this study adopted the transfer learning (TL) approach to prepare the dataset for the proposed NeuroBoruta method from the FracAtlas dataset.

Here, the idea of TL is to transfer knowledge at the parametric level from models that had been initially trained and optimized on massive datasets such as ImageNet [48]. By stripping away the final fully-connected layers and freezing the rest of networks, these networks were repurposed as fixed feature extractors for the FracAtlas dataset, effectively leveraging their pre-established computational intelligence for new data applications.

The output from these pre-trained models was set to produce 256-dimensional feature vectors per image. These feature representations were then utilized as input for the proposed method. Fig. S1 from the Supplementary Document illustrates the use of TL in this study. By employing a variety of pre-trained models, from CNNs to transformers, the study aimed to mitigate the influence of any specific model's pre-training on the final results. Five pre-trained models employed in this study to extract features, are briefly presented below.

- InceptionResNetV2: InceptionResNetV2 is constructed by integrating the strengths of two advanced architectures: Inception and ResNet. The Inception modules employ multiple convolutions and pooling layers to aggregate features at different spatial scales from one layer to the next, facilitating multi-level feature extraction. This structure allows the model to capture a broad range of features efficiently. Residual connections, a key component from ResNet, enhance the gradient flow across numerous layers. These connections create shortcut paths that directly sum features from preceding layers to subsequent layers, effectively addressing the vanishing gradient problem and allowing for significantly deeper networks [49].
- DenseNet121: DenseNet-121, abbreviated from Dense Convolutional Network, comprises 121 layers. This architecture is renowned for its 'dense blocks,' which connect each layer directly to every other subsequent layer. These dense blocks concatenate outputs from all preceding layers and feed them as inputs to subsequent layers [50]. This structure maximizes information and gradient flow throughout the network, allowing it to be both deeper and more accurate while efficiently reusing features.
- EfficientNetB0: EfficientNet-B0 is part of the EfficientNet family of models (B0 to B7) introduced by [51]. EfficientNetB0 represents a breakthrough in scaling up convolutional neural networks (CNNs) through the novel compound scaling method, which uniformly scales the network's depth, width, and resolution with a set coefficient. EfficientNet architecture utilizes multiple components, including Mobile Inverted Bottleneck Convolutions (MBConv) and Squeeze-and-Excitation (SE) optimization, yet it remains relatively lightweight [51].
- BigTransfer (BiT): BiT employs a modified ResNet architecture, extensively pre-trained on extensive datasets such as ImageNet-21 k to enhance its transfer learning capabilities. This model utilizes deep residual networks along with advanced training techniques like Group Normalization and Weight Standardization, which are crucial for maintaining performance stability across varied batch sizes and image resolutions. These features allow BiT to be effectively adapted to new tasks while preserving learned features from its extensive pre-training. Specifically, this paper focuses on the BiT version using the ResNet50 architecture trained on the ImageNet-21 k dataset [52].
- Vision Transformer (ViT): ViT shifts from conventional CNN approaches to using the transformer architecture, which was originally developed for natural language processing tasks. It starts by splitting an image into fixed-size patches, then linearly embedding each of them (i.e., flattened them), akin to tokens in a natural language model. These embeddings then pass through a series of transformer layers that use self-attention mechanisms to integrate information from different patches [53]. ViT undergoes pre-training on extensive datasets like ImageNet to adapt to new tasks.

Table I provides general information about proposed pre-trained models. It worths to mention that the number of parmeters reported here are less than in the original architecture. This discrepancy is because the top classification layer, typically used for classifying large number of classes in ImageNet, has been

TABLE I
ARCHITECTURES' SUMMARY OF UTILIZED PRE-TRAINED MODELS

Architecture	Input Size	Output Size	Number of Parameters
InceptionResNetV2	299*299	256	54,730,208
Densenet121	224*224	256	7,299,904
EfficientNetB0	224*224	256	4,377,507
BiT	224*224	256	24,024,896
ViT	224*224	256	86,389,248

removed to facilitate transfer learning. For the InceptionResNetV2 model in this research, the required input image size is set at 299 $\times$ 299 pixels, whereas the other models use an image size of 244 $\times$ 244 pixels. Additionally, the output feature size has been fixed to a 256-dimensional vector, further modifying the network from its standard setup.

C. Experiment Configurations

In this study, the performance of the proposed method has been compared with the original Boruta algorithm. For feature selection using the Boruta algorithm, RF with 200 predictors is utilized.

In configuring the method proposed in this study, more parameters need to be decided upon. The first set of these parameters pertains to the shallow neural network used inside the NeuroBoruta which is solely used for feature importance calculation. Given that in the NeuroBoruta, this network is solely used for feature selection, and features are chosen based on the impact their perturbation has on reducing model accuracy, thus fine-tuning the learner at this stage is not critical. What is required here is to select a network architecture that can generate a minimum accuracy above 50 percent. Accordingly, the network used here consists of two fully connected layers, containing 64 and 8 neurons respectively, both utilizing the ReLU activation function. The epoch was set to 100. Class weights are dynamically adjusted based on the inverse frequency of each class within the dataset to address class imbalances, thereby promoting equitable learning across both classes. For the perturbation analysis which is used for feature importance during the NeuroBoruta references suggests that typical perturbations in neural network analysis involve changes ranging from 10 to 50% of the values of individual variables [43], [54]. This study adapted the perturbation scale to include a factor of standard deviations to appropriately model these changes within the context of each dataset. Here, a scaling factor of 3 is applied to the noise added to each feature during perturbation analysis. In a similar manner, a scaling factor of 3 is used during the shadow feature generation step. Maximum iterations for NeuroBruta and Boruta are set 100. In this study, both NeuroBoruta and Boruta select features that have accumulated at least one hit. By using the one-hit threshold, it is ensured that all features demonstrating any significant impact on the model's performance are retained for further consideration. The experiments were conducted in a virtualized environment with the following hardware configuration: an Intel(R) Xeon(R) CPU @ 2.20 GHz with 2 virtual CPUs, 12 GB of memory, running on a Linux Kernel version 6.1.85+, and operating on an $x86\_64$ architecture.

V. RESULTS AND DISCUSSION

This section is organized into two distinct parts. Section V-A presents and discusses the results obtained from the proposed method, NeuroBoruta, comparing its performance to the original Boruta algorithm. The comparison extends to include performance evaluations using all features extracted from pre-trained models without any feature selection. Section V-B broadens the performance analysis by applying the proposed method to three additional non-image tabular datasets from the University of California, Irvine (UCI) ML repository. This extension aims to provide further insights into the model's performance beyond the primary FracAtlas dataset.

A. Results and Analysis

The performance evaluation process consists of two main phases: feature selection and model tuning, followed by generalization evaluation as illustrated by Fig. S2 from the Supplementary Document. Initially, the dataset is split into training and testing sets at a ratio of 70% to 30%, with stratified sampling from the target variable. Feature selection is performed on the training set, resulting in a reduced set of features that are deemed most relevant. The training set is then filtered to include only these selected features. Subsequently, multi-layer perceptron (MLP) is tuned using the filtered training set. The fine-tuned model is evaluated to ensure it meets the desired performance criteria. For feature selection, the NeuroBoruta and the original Boruta algorithm are each run on the training dataset, with a maximum iteration limit of 100 times. To extend the performance evaluation and comparison, Recursive Feature Elimination (RFE) feature selection was also considered. RFE operates by recursively eliminating features and evaluating the model performance to identify the optimal subset of features that yield the highest performance score. This choice of RFE was made for its similarity to Boruta and NeuroBoruta, as all three are wrapper methods that iteratively evaluate feature importance to improve model performance but with different mechanisms. In this study, RFE was implemented using a RF with 100 predictors as the Classifier, cross-validation with five folds and F1 score as the evaluation metric during the RFE process. Fig. S3 from the Supplementary Document illustrates the relationship between the number of features retained and the F1 score achieved, aiding in identifying the optimal number of features that balance complexity and performance across pre-trained models by RFE.

Considering that evaluating the model only once in phase one does not guarantee reproducible results. This is due to the fact that most ML and deep learning (DL) models assume that training and testing datasets have the same distribution [55]. Consequently, their performance can suffer under data distribution shifts [56], [57]. Accordingly, phase two is designed to measure the model's generalization power. For this evaluation, the dataset is repeatedly divided into training and test sets with a 70% to 30% ratio. The model, using the architecture obtained from the initial tuning, is trained from scratch on each training

set. Importantly, only the architecture of the tuned model from phase one is used, without transferring the optimal weights. The model is then tested on each test set. Performance metrics are calculated and stored for each iteration, with a number of iterations set to 30. While phase one is designed for feature selection and hyperparameter tuning, phase two aims to train the model each time with a different training set and evaluate it with a different test set. This ensures the model is exposed to various data distributions during training and testing, and the stored results allow for assessment of the model's generalization ability across different possible data distributions. The evaluation metrics used in this study are ECE, Overconfidence Error (OE), and F1 score. The choice of the F1 score is due to the imbalanced nature of the dataset. F1, by combining precision and recall, allows for a comprehensive performance assessment. The OE is another metric used to assess calibration performance. This metric specifically penalizes predictions where the level of confidence surpasses the actual accuracy, applying a weighting based on the degree of confidence [46]. OE is formulated as follows [58] (7):

$$OE = \sum_{m=1}^{M} \frac{|B_m|}{n} \left[\text{conf}(B_m) \cdot \max\left(\text{conf}(B_m) - \text{acc}(B_m), 0\right)\right] \tag{7}$$

Where, B_m represents the set of samples within the m-th bin, $\text{conf}(B_m)$ denotes the average confidence in bin m, $\text{acc}(B_m)$ represents the accuracy within bin m, and n is the total number of samples across all bins. Lower ECE and OE values indicate better calibration of the model, meaning the predicted probabilities of outcomes are more accurate.

In this study, the hyperparameter tuning of a neural network model was performed using the Keras Tuner, a library built upon TensorFlow and Keras. The search space for the hyperparameters included the number of layers, ranging from 1 to 4, and the number of units per layer, varying from 16 to 1024 with a step size of 32 units. This step size indicates the interval at which the number of units in each layer was incremented during the search. The hyperparameter search was conducted using Keras Tuner's Hyperband algorithm, an efficient method that evaluates models with varying resource allocation, balancing exploration and exploitation. The objective was to maximize validation accuracy with a validation split of 20%, meaning that 20% of the training data was used to evaluate the model's performance during the hyperparameter search. Class weights were included to ensure balanced learning across all classes. The search process set a maximum of 100 epochs for training each model, with a reduction factor of 3, meaning that the training resources were scaled down by this factor during the iterative search process. Early stopping was implemented to prevent overfitting, halting the training if the validation loss did not improve for 30 consecutive epochs. Upon completion of the hyperparameter search, the optimal hyperparameters were identified, including the best number of units in the input layer and the optimal number of layers. These model characteristics are subsequently utilized in Phase 2, with an epoch size of 1000. Table II summarized the optimal architecture and the number of selected features by NeuroBoruta, Boruta and RFE across different pre-trained models.

TABLE II
SUMMARY OF NO. OF SELECTED FEATURES AND MODELS' OPTIMAL ARCHITECTURE

Method	Pre-trained	No. of Features	Fine-tuned MLP architecure*
NeuroBoruta	BiT	136	(656, 592)
	ViT	108	(112, 320)
	Densenet121	142	(944, 272)
	EfficientNetB0	163	(592, 656, 592, 944)
	InceptionResNetV2	146	(592, 432)
Boruta	BiT	54	(336, 912, 592)
	ViT	82	(592, 976)
	Densenet121	50	(656, 784, 1008, 368, 176)
	EfficientNetB0	63	(624, 976, 592)
	InceptionResNetV2	46	(688, 208)
All Features	BiT	256	(880, 16, 368)
	ViT	256	(144, 304, 272)
	Densenet121	256	(592, 720, 912)
	EfficientNetB0	256	(144, 752, 272, 304)
	InceptionResNetV2	256	(176, 208)
RFE	BiT	12	(560, 144, 16, 752)
	ViT	9	(496, 688, 688)
	Densenet121	17	(688, 752, 688, 16)
	EfficientNetB0	16	(496, 336, 464)
	InceptionResNetV2	19	(208, 784, 976, 144, 784)

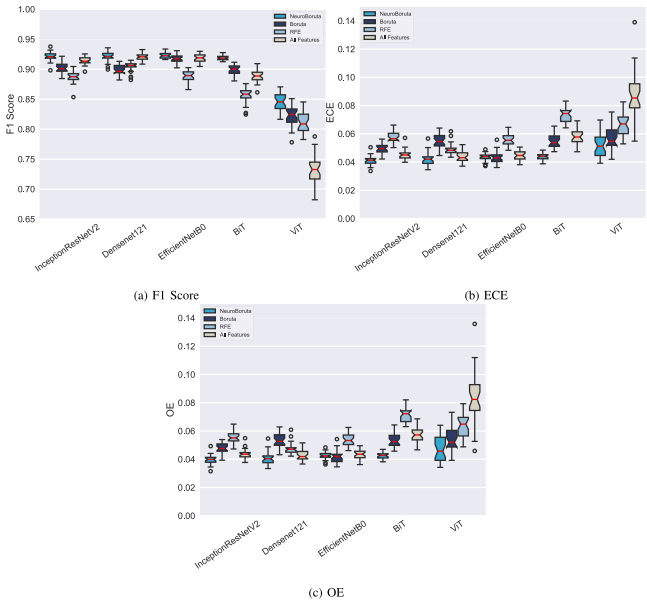
* Each number signifies the size of neurons in a layer. In the cases where a sequence of numbers is presented, such as (i_1, i_2, i_3), these correspond to multiple hidden layers within the network.

As reflected in Table II, NeuroBoruta consistently selected a higher number of features compared to the original Boruta and RFE methods across all pre-trained models. NeuroBoruta's selection of a relatively higher number of features compared to traditional methods like Boruta or RFE can be attributed to its approach, which emphasizes not just the predictive accuracy but also the calibration of the model. This method may retain features that, while not significantly boosting accuracy, do contribute to reducing prediction uncertainty. For instance, some features might show a weaker direct correlation with the target variable but may provide essential context that helps the model make more reliable probability estimations, thereby improving the model's overall confidence. Boruta showed conservative feature selection, particularly noticeable in models like InceptionResNetV2 and Densenet121, where only 46 and 50 features were selected, respectively. The conservative nature of Boruta might limit the model's performance in complex scenarios by potentially omitting features that contribute to the prediction but have less apparent statistical importance. RFE demonstrated the most stringent selection criteria, selecting fewer than 20 features for each model. This extreme reduction could lead to models that are very efficient computationally but may miss out on performance if some of the discarded features hold subtle yet crucial information. The approach of using all features serves as a benchmark when all 256 features are retained.

Table III summarized the performance result over 30 run across different pre-trained models and feature selection methods. Additional metrics, including accuracy, precision, and recall, are provided in Supplementary Table S.I for further reference. Fig. 2 illustrates the distribution of the evaluation

TABLE III
PERFORMANCE COMPARISON OF NEUROBORUTA, BORUTA, RFE AND ALL FEATURES

		F1 Score	ECE	OE
NeuroBoruta	BiT	% 91.8975 ± 0.3850	0.0438 ± 0.0022	0.0425 ± 0.0021
	ViT	% 84.4447 ± 1.4769	0.0516 ± 0.0084	0.0476 ± 0.0091
	Densenet121	% 92.1251 ± 0.8174	0.0417 ± 0.0045	0.0404 ± 0.0045
	EfficientNetB0	% 92.2934 ± 0.4577	0.0435 ± 0.0026	0.0426 ± 0.0025
	InceptionResNetV2	% 92.1289 ± 0.7164	0.0409 ± 0.0033	0.0396 ± 0.0034
Boruta	BiT	% 89.8880 ± 0.8532	0.0549 ± 0.0046	0.0534 ± 0.0048
	ViT	% 82.1482 ± 1.6535	0.0573 ± 0.0088	0.0544 ± 0.0090
	Densenet121	% 87.5310 ± 12.5129	0.0653 ± 0.0568	0.0635 ± 0.0570
	EfficientNetB0	% 91.7173 ± 0.7864	0.0430 ± 0.0041	0.0416 ± 0.0042
	InceptionResNetV2	% 90.1771 ± 0.8341	0.0497 ± 0.0038	0.0479 ± 0.0038
All Features	BiT	% 88.8248 ± 0.9852	0.0582 ± 0.0052	0.0574 ± 0.0052
	ViT	% 73.2837 ± 2.2897	0.0883 ± 0.0169	0.0838 ± 0.0177
	Densenet121	% 92.0658 ± 0.5763	0.0440 ± 0.0037	0.0429 ± 0.0035
	EfficientNetB0	% 91.8203 ± 0.6447	0.0447 ± 0.0034	0.0435 ± 0.0034
	InceptionResNetV2	% 91.4029 ± 0.6043	0.0452 ± 0.0033	0.0438 ± 0.0032
RFE	BiT	% 85.6458 ± 1.2413	0.0731 ± 0.0052	0.0712 ± 0.0050
	ViT	% 81.1578 ± 1.9433	0.0666 ± 0.0079	0.0637 ± 0.0078
	Densenet121	% 90.4590 ± 0.7896	0.0489 ± 0.0038	0.0476 ± 0.0038
	EfficientNetB0	% 88.8263 ± 0.9355	0.0557 ± 0.0041	0.0536 ± 0.0043
	InceptionResNetV2	% 88.7183 ± 0.9240	0.0573 ± 0.0036	0.0556 ± 0.0037



(a) F1 Score (b) ECE

(c) OE

Fig. 2. Comparison box plots of F1 score, ECE, and OE.

TABLE IV
STATISTICAL COMPARISON RESULTS OF DIFFERENT FEATURE SELECTION METHODS ACROSS VARIOUS PRE-TRIANED

		P-Value			Adjusted P-Value		
Model	Comparison	F1 Score	ECE	OE	F1 Score	ECE	OE
BiT	NeuroBoruta v. Boruta	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	NeuroBoruta v. RFE	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	NeuroBoruta v. All Feat.	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	Boruta v. RFE	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	Boruta v. All Feat.	0.0000	0.0128	0.0032	0.0003	**0.0770**	0.0193
	RFE v. All Feat.	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
ViT	NeuroBoruta v. Boruta	0.0000	0.0113	0.0026	0.0000	**0.0678**	0.0154
	NeuroBoruta v. RFE	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	NeuroBoruta v. All Feat.	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	Boruta v. RFE	0.0364	0.0000	0.0000	**0.2186**	0.0003	0.0003
	Boruta v. All Feat.	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	RFE v. All Feat.	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Densenet121	NeuroBoruta vs Boruta	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	NeuroBoruta vs RFE	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	NeuroBoruta v. All Feat.	**0.7000**	**0.0606**	0.0405	**1.0000**	**0.3634**	**0.2429**
	Boruta vs RFE	0.0028	0.0000	0.0000	0.0166	0.0001	0.0002
	Boruta v. All Feat.	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	RFE v. All Feat.	0.0000	0.0000	0.0000	0.0000	0.0000	0.0001
EfficientNetB0	NeuroBoruta vs Boruta	0.0020	**0.4522**	**0.2286**	0.0121	**1.0000**	**1.0000**
	NeuroBoruta vs RFE	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	NeuroBoruta v. All Feat.	0.0043	**0.1403**	**0.2206**	0.0260	**0.8417**	**1.0000**
	Boruta vs RFE	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	Boruta v. All Feat.	**0.6120**	**0.1241**	**0.0841**	**1.0000**	**0.7448**	**0.5044**
	RFE v. All Feat.	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
InceptionResNetV2	NeuroBoruta vs Boruta	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	NeuroBoruta vs RFE	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	NeuroBoruta v. All Feat.	0.0002	0.0000	0.0000	0.0011	0.0002	0.0001
	Boruta vs RFE	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	Boruta v. All Feat.	0.0000	0.0000	0.0000	0.0000	0.0000	0.0001
	RFE v. All Feat.	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

Highlighted values indicates no significant difference (P-value $>$ 0.05).

metrics across pre-trained models. In the comparative analysis of feature selection methods across various pre-trained models NeuroBoruta consistently demonstrates superior performance in terms of F1 score possibly due to a more comprehensive feature set that captures relevant information. This method also exhibits lower ECE and OE values, suggesting it not only enhances prediction accuracy but also improves the confidence and reliability of these predictions. Boruta shows moderate performance with F1 scores generally lower than those of NeuroBoruta, particularly in models such as ViT and Densenet121. The ECE and OE values are comparatively higher. RFE tends to select fewer features and shows the lowest F1 scores among the methods, especially noticeable in models like ViT and BiT. The higher ECE and OE values suggest that the minimal feature set may affect the model's ability to generalize and calibrate accurately. Using all features often results in lower performance compared to feature selection methods, particularly in calibration metrics, as seen in the ViT model. This suggests that including all features can introduce noise, leading to poorer model performance and calibration.

To statistically validate the differences in model performance across various metrics, the Wilcoxon signed-rank test was employed. This non-parametric test is designed to compare two related samples to determine whether their population mean ranks differ. The Wilcoxon signed-rank test was performed on each pair of methods for each metric, yielding a p-value for each comparison. If the p-value was less than 0.05, the null hypothesis (which states that there is no difference between the methods) was rejected, indicating a statistically significant difference.

To account for multiple comparisons and control the family-wise error rate, the Bonferroni correction was applied to the p-values. This correction adjusts the threshold for significance based on the number of comparisons being made, ensuring that the likelihood of Type I errors is minimized. The results, including the original and corrected p-values, Table IV presents the original and corrected p-values across various models and metrics.

It is worth mentioning that the performance comparisons presented in this study are confined to the feature sets output by each individual pre-trained model. In other words, cross pre-trained model comparisons, such as contrasting the feature selection methods applied to BiT with those applied to

InceptionResNet, have not been undertaken. This decision is based on the understanding that each feature set output from the different pre-trained models is considered as a distinct dataset. The primary objective of this approach is to have the efficacy of the discussed feature selection methods meticulously examined within the unique context of each dataset. Consequently, this analysis has been specifically designed to assess the performance enhancements that are brought by each feature selection method on a per-model basis, rather than to evaluate or compare the inherent quality of the features generated by each pre-trained model. This focus ensures that a thorough and relevant evaluation of the methods in question is conducted, tailored to the specific characteristics and challenges presented by each dataset.

In this section, the discussion is centered on the statistical analysis conducted to evaluate the performance of NeuroBoruta in comparison with other feature selection methods, specifically Boruta, RFE, and the strategy of using all features. The focus of this analysis is particularly tailored to underscore the effectiveness of NeuroBoruta, as it represents the core innovation of this study. Comparisons between Boruta and RFE, while noted, fall outside the primary interest of this paper and are mentioned only to contextualize the overall landscape of feature selection methods relative to the proposed NeuroBoruta approach. The detailed statistical tests provided in Table IV reinforce that NeuroBoruta provides significant improvements in terms of predictive accuracy and confidence. For the BiT model, NeuroBoruta demonstrates a profound improvement over Boruta, RFE, and the strategy of using all features, with all comparisons showing p-values significantly lower than 0.05. In the case of ViT, while NeuroBoruta shows improvements over Boruta and RFE in F1 Score and calibration metrics, the adjustments for multiple comparisons reveal that the improvements in ECE are not statistically significant (adjusted p-values > 0.05). However, it is noteworthy that NeuroBoruta shows a significant improvement in OE (adjusted p-values < 0.05).

NeuroBoruta's impact on DenseNet121 shows significant improvements in F1 Score when compared to other methods. However, the results for ECE and OE are not significantly better than using all features after adjusting for multiple comparisons. This indicates that while NeuroBoruta can refine the predictive accuracy by selecting relevant features, the dense connectivity of DenseNet121 may diminish the effectiveness of these features in enhancing the model's calibration. The dense layers may blend feature information in a way that marginalizes the benefits of selective feature inclusion. Further insights can be gleaned from comparing Boruta's performance to using all features within the same DenseNet121 architecture. In this comparison, Boruta underperforms relative to using all features, indicating that Boruta's method of feature selection does not effectively capitalize on the model's capabilities. In contrast, even though NeuroBoruta does not significantly improve calibration metrics over using all features, it maintains similar performance levels with fewer features. This outcome highlights that NeuroBoruta still manages to maintain a balance between reducing feature dimensionality and sustaining model performance. This efficiency in feature usage without a loss in performance underscores the capability of NeuroBoruta in handling complex architectures where traditional methods like Boruta may fail to deliver optimal results. For EfficientNetB0, NeuroBoruta significantly improves the F1 Score compared to Boruta and RFE, yet the improvements in calibration metrics (ECE and OE) are not statistically significant after correction. This outcome suggests that the efficiency-focused architecture of EfficientNetB0, which is designed to minimize computational expenses, may inherently limit the impact of expanded feature sets on calibration improvements. The model's smaller parameter count and optimized processing paths could mean that the quality and precise targeting of features are more crucial than the mere quantity of features. Additional insights can be derived from examining the comparison between Boruta and using all features, which also yields no significant differences. This finding indicates that even Boruta's attempt at feature selection does not provide a statistically significant enhancement over using all available features. This result is critical as it implies that within the architecture of EfficientNetB0, the process of feature selection itself does not substantially contribute to improving model performance, either in terms of predictive accuracy or calibration. While the architecture of EfficientNetB0 may inherently limit the benefits of feature selection for calibration improvements—given its emphasis on efficiency and a smaller parameter count—the ability of NeuroBoruta to still outperform Boruta in terms of F1 Score is notable. This suggests that NeuroBoruta effectively balances feature reduction with the retention of critical predictive features. With InceptionResNetV2, NeuroBoruta shows a clear advantage in terms of F1 Score and calibration metrics over Boruta and RFE, with all comparisons yielding p-values significantly below 0.05.

The detailed model-by-model analysis reveals that NeuroBoruta generally enhances F1 Scores and model confident compares to Boruta and RFE feature selections across models. While this is of lesser interest to the main focus of this paper, it is still informative to consider the comparative performance of Boruta, RFE, and the strategy of using all features. Evaluating these methods alongside their respective number of selected features reveals insights into their relative effectiveness. Generally, Boruta tends to perform better than RFE, which can be attributed to its more sophisticated feature shadowing technique that tends to preserve more informative features than RFE's more straightforward elimination process. In terms of achieving a balance between feature reduction and maintaining high model performance, Boruta appears to be superior to RFE. Nonetheless, the method of using all features, despite its lack of selectivity, sometimes offers competitive performance, particularly in models that can benefit from larger datasets. This indicates that the effectiveness of a feature selection strategy can be highly dependent on the specific characteristics and capabilities of the underlying model architecture.

Having examined the performance metrics, understanding the computational demands provides a more comprehensive view of the trade-offs involved in feature selection methods. Although the integration of neural networks and noise-augmented shadow features increases computational demands, the time complexity of NeuroBoruta is comparable to the original Boruta algorithm. Both methods begin with creating shadow features, with a time

complexity of $O(d)$, where d is the number of features. The next step involves model training—neural networks for NeuroBoruta and Random Forest (RF) for Boruta.

Neural network training, the most computationally intensive part of NeuroBoruta, has a complexity of $O(N \times L \times M^2 \times E)$, where N is the number of samples, L the number of layers, E the number of epochs, and M the average number of neurons per layer. Perturbation Analysis further increases the complexity to $O(d \times N \times L \times M^2 \times E)$.

For Boruta, training an RF has a complexity of $O(T \times N \times \log(N) \times d)$, where T is the number of trees. The iterative process, common to both methods, repeats for a given number of iterations I. Thus, NeuroBoruta's complexity is $O(d \times N \times L \times M^2 \times E)$, compared to $O(T \times N \times \log(N) \times d)$ for Boruta.

While neural networks may lead to higher computational costs, this is mitigated by using a shallow learner without fine-tuning in NeuroBoruta. Boruta's reliance on RF offers scalability but may not handle complex feature interactions as effectively.

B. Extended Comparative Study

This section explores the versatility of the proposed method beyond the image data from the FracAtlas dataset. To this end, three distinct datasets from the UCI ML Repository, each with unique characteristics, were selected to evaluate the performance of the proposed method compared to the conventional Boruta method. Brief descriptions of each dataset are presented below:

- Smartphone-based recognition of human activities and postural transitions (SB-RHAPT) [59]: This dataset contains 10 299 instances and 561 features collected from smartphone sensors, recording six different activities and postural transitions.
- Epileptic seizure recognition (ESR) [60]: With 11 500 instances and 179 features, this dataset records EEG values to distinguish between seizure and non-seizure activities. The original five-category target variable is simplified to a binary classification task, creating an imbalanced dataset.
- Parkinson's disease classification (PDC) [61]: This dataset includes 756 instances and 755 features of biomedical voice measurements, classifying instances into Parkinson's Disease and Healthy categories.

The same evaluation process and statistical tests from previous section were used for these datasets. Number of selected features, the performance results and statistical analysis results are summarized in Tables V, VI, VII respectively. Also, additional metrics, including accuracy, precision, and recall, are provided in Supplementary Table S.II for further reference. Fig. S4 from the Supplementary Document illustrates the distribution of the evaluation metrics across ESR, PD, and RHAPT datasets.

For the ESR dataset, it is observed that Boruta selected 178 features, equivalent to the total number of features available, indicating no reduction or selection was effectively made. In contrast, NeuroBoruta successfully reduced the feature set to 118, demonstrating its capability to discern and retain the most relevant features for modeling. In the PD dataset, NeuroBoruta selected 123 features, which is a significant increase compared to Boruta's selection of 78 features. Similarly, for the RHAPT dataset, NeuroBoruta identified 492 features compared to Boruta's 478, indicating a preference for a slightly larger feature set which may contribute to model robustness and accuracy. These findings are consistent with those reported in the previous section, where NeuroBoruta consistently selects more features compared to Boruta.

As indicated by Table VI, NeuroBoruta consistently achieves high F1 Scores, notably outperforming the other methods in most cases. For instance, in the ESR dataset, NeuroBoruta achieves an F1 Score of 97.5342% with notably low ECE and OE, indicating not only high accuracy but also reliable calibration and confidence in its predictions. In the PD dataset, while NeuroBoruta's F1 Score is comparably high, the differences in ECE and OE are not as pronounced compared to the Boruta

TABLE V
SUMMARY OF NO. OF SELECTED FEATURES AND OPTIMAL ARCHITECTURES OF MODELS ACROSS THREE DATASETS: ESR, PD, AND RHAPT

Method	Dataset	No. of Features	Fine-tuned MLP architecure
NeuroBoruta	ESR	118	(408, 946, 364, 534)
	PD	123	(176, 784, 16)
	RHAPT	492	(272, 976, 528)
Boruta	ESR	178	(336 624)
	PD	78	(48, 944, 752, 944, 528)
	RHAPT	478	(688, 624, 752, 688)
All Features	ESR	178	*
	PD	755	(720, 400, 272, 144, 1008)
	RHAPT	561	(304, 240)

* The features selected by Boruta and the complete feature set for ESR are identical, and a single model has been tuned for them.

TABLE VI
PERFORMANCE COMPARISON OF NEUROBORUTA, BORUTA, RFE AND ALL FEATURES ACROSS THREE DATASETS: ESR, PD, AND RHAPT

		F1 Score	ECE	OE
NeuroBoruta	ESR	% 97.5342 ± 0.4592	0.0084 ± 0.0015	0.0074 ± 0.0016
	PD	% 91.6095 ± 0.8640	0.0601 ± 0.0053	0.0579 ± 0.0057
	RHAPT	% 98.9528 ± 0.1321	0.0092 ± 0.0011	0.0088 ± 0.0010
Boruta	ESR	% 94.6043 ± 0.6410	0.0119 ± 0.0014	0.0085 ± 0.0013
	PD	% 91.4418 ± 0.9091	0.0621 ± 0.0067	0.0607 ± 0.0067
	RHAPT	% 98.8392 ± 0.2026	0.0097 ± 0.0014	0.0092 ± 0.0014
All Features	ESR*	% 94.6043 ± 0.6410	0.0119 ± 0.0014	0.0085 ± 0.0013
	PD	% 90.5719 ± 0.7385	0.0687 ± 0.0049	0.0675 ± 0.0048
	RHAPT	% 98.9097 ± 0.1925	0.0089 ± 0.0015	0.0083 ± 0.0015

* The features selected by Boruta and the complete feature set for ESR are identical, and a single model has been tuned for them. Thus results for All features are repeated.

TABLE VII
FURTHER STATISTICAL COMPARISON RESULTS OF DIFFERENT FEATURE SELECTION METHODS ACROSS THREE DATASETS: ESR, PD, AND RHAPT

		P-Value			Adjusted P-Value		
Model	Comparison	F1 Score	ECE	OE	F1 Score	ECE	OE
ESR	NeuroBoruta v. Boruta	0.0000	0.0000	0.0032	0.0000	0.0000	0.0097
PD	NeuroBoruta v. Boruta	**0.7922**	**0.3818**	**0.2129**	**1.0000**	**1.0000**	**0.6388**
	NeuroBoruta v. All Feat.	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	Boruta v. All Feat.	0.0000	0.0000	0.0000	0.0001	0.0001	0.0001
RHAPT	NeuroBoruta v. Boruta	0.0058	**0.1460**	**0.4280**	0.0173	**0.4380**	**1.0000**
	NeuroBoruta v. All Feat.	**0.4161**	**0.1706**	0.0262	**1.0000**	**0.5118**	**0.0787**
	Boruta v. All Feat.	**0.1048**	0.0099	0.0037	**0.3145**	0.0298	0.0112

Highlighted values indicates no significant difference (P-value $>$ 0.05)

and all features methods. For the RHAPT dataset, NeuroBoruta again shows superior F1 Scores but with ECE and OE values that are comparable to those achieved using Boruta and the all Features method. The subsequent statistical tests further validate these findings. In the ESR dataset, significant p-values (below 0.05) for the comparisons of F1 Score, ECE, and OE between NeuroBoruta and Boruta confirm the statistical significance of the performance improvements with NeuroBoruta. In contrast, in the PD dataset, the high p-values for the F1 Score and OE when comparing NeuroBoruta and Boruta ($p > 0.05$) indicate no significant difference. The PD dataset consists of 756 records with 755 features and performance constraints observed might be attributable to the limited dataset size, which poses challenges for neural network architectures that typically require a larger sample size to generalize effectively. Considering neural network is the backbone of the proposed NeuroBoruta method, this outcome may reflect a sensitivity of NeuroBoruta to dataset size, with smaller datasets potentially having similar performance with Boruta. In the RHAPT dataset, while the F1 Score shows a significant improvement when NeuroBoruta is compared to Boruta (p-values < 0.05), the non-significant results for ECE and OE (adjusted p-values > 0.05). In the RHAPT dataset, the number of features selected by NeuroBoruta closely aligns with those selected by Boruta, differing only slightly. This minimal difference suggests that both methods achieve a comparable optimization of the feature set. Despite the marginal increase in the number of features by NeuroBoruta, a significant improvement in the F1 score was noted, indicating that the additional features selected might be marginally more informative or relevant for the model's predictive accuracy. However, the calibration metrics (ECE and OE) exhibited similar performance between NeuroBoruta and Boruta. This observation implies that the slight increase in feature selection by NeuroBoruta does not substantially impact the model's calibration characteristics. It is therefore reasonable to conclude that when the number of features selected by both methods is approximately equivalent, their impact on calibration is expected to be similar.

VI. Conclusion

The innovation of NeuroBoruta lies in selecting a feature subset that not only improves model accuracy but also enhances prediction uncertainty. The proposed method extends its focus from solely error-based performance to simultaneously optimizing both error-based metrics (e.g., F1 score) and confidence measures (e.g., ECE and Brier score). NeuroBoruta was evaluated on the FracAtlas medical imaging dataset using transfer learning with pre-trained models for feature extraction and three UCI datasets. Comparative analyses with Boruta, RFE, and all-features models demonstrated that NeuroBoruta excelled in enhancing both predictive accuracy and uncertainty, particularly in the FracAtlas dataset. However, in smaller datasets like PD, NeuroBoruta's advantages were less pronounced, underscoring its sensitivity to data size.

To the best of the authors' knowledge, the consideration of prediction uncertainty in feature selection has been largely overlooked in the existing literature. Traditional wrapper-based methods predominantly focus on optimizing prediction accuracy by minimizing error rates, without explicitly addressing the impact of feature selection on model uncertainty. This presents an opportunity to extend current methodologies by adapting wrapper methods to prioritize the selection of features that not only improve predictive accuracy but also enhance the model's ability to quantify and manage uncertainty.

Moreover, the possible extend of this work could be systematically studying the effects of varying levels of noise on shadow features and evaluating the robustness of different pre-trained models under these conditions. Additionally, investigating the potential of integrating filter-based methods that maximize information gain with the proposed method that aim to minimize prediction uncertainty could provide deeper insights into the development of more robust and uncertainty-aware feature selection strategies.

References

[1] J. Tang, S. Alelyani, and H. Liu, "Feature selection for classification: A review" in *Data classification: Algorithms and applications*. Boca Raton, FL, USA: CRC Press, 2014, pp. 37–64.

[2] J. Xu, K. Qu, Y. Sun, and J. Yang, "Feature selection using self-information uncertainty measures in neighborhood information systems," *Appl. Intell.*, vol. 53, no. 4, pp. 4524–4540, 2023.

[3] G. Sosa-Cabrera, M. García-Torres, S. Gómez-Guerrero, C. E. Schaerer, and F. Divina, "A multivariate approach to the symmetrical uncertainty measure: Application to feature selection problem," *Inf. Sci.*, vol. 494, pp. 1–20, 2019.

[4] I. M. Johnstone and D. M. Titterington, "Statistical challenges of high-dimensional data," *Roy. Soc. Philos. Trans. A*, vol. 367, pp. 4237–4253, 2009.

[5] Y. Ding, J. Liu, J. Xiong, and Y. Shi, "Revisiting the evaluation of uncertainty estimation and its application to explore model complexity-uncertainty trade-off," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops*, 2020, pp. 4–5.

[6] M. Abdar et al., "A review of uncertainty quantification in deep learning: Techniques, applications and challenges," *Inf. Fusion*, vol. 76, pp. 243–297, 2021.

[7] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On calibration of modern neural networks," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 1321–1330.

[8] L. Kaplan, F. Cerutti, M. Sensoy, A. Preece, and P. Sullivan, "Uncertainty aware AI ML: Why and how," 2018, *arXiv:1809.07882*.

[9] L. Song, A. Smola, A. Gretton, J. Bedo, and K. Borgwardt, "Feature selection via dependence maximization," *J. Mach. Learn. Res.*, vol. 13, no. 5, pp. 1393–1434, 2012.

[10] P. Zhang and W. Gao, "Feature selection considering uncertainty change ratio of the class label," *Appl. Soft Comput.*, vol. 95, 2020, Art. no. 106537.

[11] R. Bellman and R. Kalaba, "On adaptive control processes," *IRE Trans. Autom. Control*, vol. 4, no. 2, pp. 1–9, 1959.

[12] R. Zebari, A. Abdulazeez, D. Zeebaree, D. Zebari, and J. Saeed, "A comprehensive review of dimensionality reduction techniques for feature selection and feature extraction," *J. Appl. Sci. Technol. Trends*, vol. 1, no. 2, pp. 56–70, 2020.

[13] A. Jović, K. Brkić, and N. Bogunović, "A review of feature selection methods with applications," in *Proc. 38th Int. Conv. Inf. Commun. Technol., Electron. Microelectronics*, 2015, pp. 1200–1205.

[14] M. Habibpour et al., "An uncertainty-aware deep learning framework for defect detection in casting products, 2021, *arXiv:2107.11643*.

[15] M. B. Kursa, A. Jankowski, and W. R. Rudnicki, "Boruta–A system for feature selection," *Fundamenta Informaticae*, vol. 101, no. 4, pp. 271–285, 2010.

[16] N. S. Maurya, S. Kushwah, S. Kushwaha, A. Chawade, and A. Mani, "Prognostic model development for classification of colorectal adenocarcinoma by using machine learning model based on feature selection technique Boruta," *Sci. Rep.*, vol. 13, no. 1, 2023, Art. no. 6413.

[17] K. Debbi et al., "Radiomics model to classify mammary masses using breast DCE-MRI compared to the BI-RADS classification performance," *Insights Imag.*, vol. 14, no. 1, 2023, Art. no. 64.

[18] E. Santos Febles, M. Ontivero, M. Ortega Valdés Sosa, and H. Sahli, "Machine learning techniques for the diagnosis of schizophrenia based on event-related potentials," *Front. Neuroinform.*, vol. 16, 2022, Art. no. 893788.

[19] S. Subbiah, K. S. M. Anbananthen, S. Thangaraj, S. Kannan, and D. Chelliah, "Intrusion detection technique in wireless sensor network using grid search random forest with Boruta feature selection algorithm," *J. Commun. Netw.*, vol. 24, no. 2, pp. 264–273, 2022.

[20] A. Rashidi Nasab and H. Elzarka, "Optimizing machine learning algorithms for improving prediction of bridge deck deterioration: A case study of Ohio bridges," *Buildings*, vol. 13, no. 6, 2023, Art. no. 1517.

[21] A. S. Maliuk, Z. Ahmad, and J.-M. Kim, "Hybrid feature selection framework for bearing fault diagnosis based on wrapper-WPT," *Machines*, vol. 10, no. 12, 2022, Art. no. 1204.

[22] H. Ahmadpour, O. Bazrafshan, E. Rafiei-Sardooi, H. Zamani, and T. Panagopoulos, "Gully erosion susceptibility assessment in the kondoran watershed using machine learning algorithms and the Boruta feature selection," *Sustainability*, vol. 13, no. 18, 2021, Art. no. 10110.

[23] M. Jamei et al., "A high dimensional features-based cascaded forward neural network coupled with MVMD and Boruta-GBDT for multi-step ahead forecasting of surface soil moisture," *Eng. Appl. Artif. Intell.*, vol. 120, 2023, Art. no. 105895.

[24] S. S. Subbiah, S. K. Paramasivan, K. Arockiasamy, S. Senthivel, and M. Thangavel, "Deep learning for wind speed forecasting using bi-LSTM with selected features," *Intell. Automat. Soft Comput.*, vol. 35, no. 3, pp. 3829–3844, 2023.

[25] P. Borugadda, R. Lakshmi, and S. Sahoo, "Transfer learning VGG16 model for classification of tomato plant leaf diseases: A novel approach for multi-level dimensional reduction," *Pertanika J. Sci. Technol.*, vol. 31, no. 2, pp. 813–841, 2023.

[26] B. Venkatesh and J. Anuradha, "A review of feature selection and its methods," *Cybern. Inf. Technol.*, vol. 19, no. 1, pp. 3–26, 2019.

[27] S. P. Potharaju and M. Sreedevi, "A novel m-cluster of feature selection approach based on symmetrical uncertainty for increasing classification accuracy of medical datasets," *J. Eng. Sci. Technol. Rev.*, vol. 10, no. 6, pp. 154–162, 2017.

[28] L. Zhang and X. Chen, "Feature selection methods based on symmetric uncertainty coefficients and independent classification information," *IEEE Access*, vol. 9, pp. 13845–13856, 2021.

[29] S. Bakhshandeh, R. Azmi, and M. Teshnehlab, "Symmetric uncertainty class-feature association map for feature selection in microarray dataset," *Int. J. Mach. Learn. Cybern.*, vol. 11, pp. 15–32, 2020.

[30] W. Xu, K. Yuan, W. Li, and W. Ding, "An emerging fuzzy feature selection method using composite entropy-based uncertainty measure and data distribution," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 7, no. 1, pp. 76–88, Feb. 2023.

[31] H. Peng and Y. Fan, "Feature selection by optimizing a lower bound of conditional mutual information," *Inf. Sci.*, vol. 418, pp. 652–667, 2017.

[32] K. Pandey, A. Mishra, P. Rani, J. Ali, and R. Chakrabortty, "Selecting features by utilizing intuitionistic fuzzy entropy method," *Decis. Mak.: Appl. Manage. Eng.*, vol. 6, no. 1, pp. 111–133, 2023.

[33] L. Sun and J. Xu, "Feature selection using mutual information based uncertainty measures for tumor classification," *Bio- Med. Mater. Eng.*, vol. 24, no. 1, pp. 763–770, 2014.

[34] L. Sun, X. Zhang, Y. Qian, J. Xu, and S. Zhang, "Feature selection using neighborhood entropy-based uncertainty measures for gene expression data classification," *Inf. Sci.*, vol. 502, pp. 18–41, 2019.

[35] M. Rostami, K. Berahmand, and S. Forouzandeh, "A novel method of constrained feature selection by the measurement of pairwise constraints uncertainty," *J. Big Data*, vol. 7, no. 1, 2020, Art. no. 83.

[36] A. K. Sinha, P. Shende, and N. Namdev, "Uncertainty optimization based feature subset selection model using rough set and uncertainty theory," *Int. J. Inf. Technol.*, vol. 14, no. 5, pp. 2723–2739, 2022.

[37] G. Zhang, Y. Song, S. Liao, L. Qu, and Z. Li, "Uncertainty measurement for a three heterogeneous information system and its application in feature selection," *Soft Comput.*, vol. 26, no. 4, pp. 1711–1725, 2022.

[38] A. K. Sinha and P. Shende, "Uncertainty optimization based feature selection model for stock marketing," *Comput. Econ.*, vol. 63, no. 1, pp. 357–389, 2024.

[39] Z. Wang, S. Gao, Y. Zhang, and L. Guo, "Symmetric uncertainty-incorporated probabilistic sequence-based ant colony optimization for feature selection in classification," *Knowl.-Based Syst.*, vol. 256, 2022, Art. no. 109874.

[40] S. P. Potharaju, M. Sreedevi, and S. S. Amiripalli, "An ensemble feature selection framework of sonar targets using symmetrical uncertainty and multi-layer perceptron (SU-MLP)," in *Proc. Cognitive Informat. Soft Comput.*, Springer, 2019, pp. 247–256.

[41] O. Goldstein, M. Kachuee, K. Karkkainen, and M. Sarrafzadeh, "Target-focused feature selection using uncertainty measurements in health-care data," *ACM Trans. Comput. Healthcare*, vol. 1, no. 3, pp. 1–17, 2020.

[42] J.-X. Zhong and H. Zhang, "Uncertainty-aware invase: Enhanced breast cancer diagnosis feature selection," 2021, *arXiv:2105.02693*.

[43] H. Gharoun, A. Keramati, M. M. Nasiri, and A. Azadeh, "An integrated approach for aircraft turbofan engine fault detection based on data mining techniques," *Expert Syst.*, vol. 36, no. 2, 2019, Art. no. e12370.

[44] L. Famiglini et al., "Towards a rigorous calibration assessment framework: Advancements in metrics, methods, and use," *Front. Artif. Intell. Appl.*, vol. 372, pp. 645–652, 2023.

[45] M. Habibpour et al., "Uncertainty-aware credit card fraud detection using deep learning," *Eng. Appl. Artif. Intell.*, vol. 123, 2023, Art. no. 106248.

[46] T. Dawood et al., "Uncertainty aware training to improve deep learning model calibration for classification of cardiac MR images," *Med. Image Anal.*, vol. 88, 2023, Art. no. 102861.

[47] I. Abedeen, M. A. Rahman, F. Z. Prottyasha, T. Ahmed, T. M. Chowdhury, and S. Shatabda, "Fracatlas: A dataset for fracture classification, localization and segmentation of musculoskeletal radiographs," *Sci. Data*, vol. 10, no. 1, 2023, Art. no. 521.

[48] H. E. Kim, A. Cosa-Linan, N. Santhanam, M. Jannesari, M. E. Maros, and T. Ganslandt, "Transfer learning for medical image classification: A literature review," *BMC Med. Imag.*, vol. 22, no. 1, 2022, Art. no. 69.

[49] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," in *Proc. AAAI Conf. Artif. Intell.*, vol. 31, no. 1, 2017, pp. 4278–4284.

[50] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 4700–4708.

[51] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 6105–6114.

[52] A. Kolesnikov et al., "Big transfer (BiT): General visual representation learning," in *Proc. 16th Eur. Conf. Comput. Vis.*, Glasgow, U.K., Springer, 2020, pp. 491–507.

[53] A. Dosovitskiy et al., "An image is worth 16x16 words: Transformers for image recognition at scale," in *Proc. Int. Conf. Learn. Representations*, 2021. [Online]. Available: https://openreview.net/forum?id=YicbFdNTTy

[54] C. Aldrich and L. Auret, *Unsupervised Process Monitoring and Fault Diagnosis With Machine Learning Methods*. vol. 16, no. 3. London, U.K.: Springer, 2013.

[55] J. Wang et al., "Generalizing to unseen domains: A survey on domain generalization," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 8, pp. 8052–8072, Aug. 2023.

[56] D. Wang, E. Shelhamer, S. Liu, B. Olshausen, and T. Darrell, "Tent: Fully test-time adaptation by entropy minimization," in *Proc. Int. Conf. Learn. Representations*, 2021. [Online]. Available: https://openreview.net/forum?id=uXl3bZLkr3c

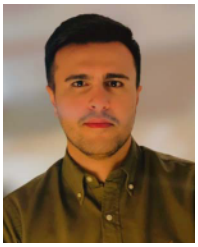
[57] H. Gharoun, F. Momenifar, F. Chen, and A. Gandomi, "Meta-learning approaches for few-shot learning: A survey of recent advances," *ACM Comput. Surv.*, vol. 56, pp. 1–41, 2023.

[58] S. Thulasidasan, G. Chennupati, J. A. Bilmes, T. Bhattacharya, and S. Michalak, "On mixup training: Improved calibration and predictive uncertainty for deep neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 13911–13922.

[59] A. D. O. L. Reyes-Ortiz, Jorge, and X. Parra, "Smartphone-based recognition of human activities and postural transitions data set," UCI Mach. Learn. Repository, School Inf. Comput. Sci., Univ. California, Irvine, CA, USA, 2015. Accessed: Jun. 20, 2024. [Online]. Available: https://archive.ics.uci.edu/dataset/341/smartphone+based+recognition+of+human+activities+and+postural+transitions

[60] Q. Wu and E. Fokoue, "Epileptic seizure recognition data set," UCI Mach. Learn. Repository, School Inf. Comput. Sci., Univ. California, Irvine, CA, USA, 2017. Accessed: Jun. 20, 2024. [Online]. Available: https://archive.ics.uci.edu/ml/datasets/Epileptic+Seizure+Recognition

[61] C. Sakar, G. Serbes, A. Gunduz, H. Nizam, and B. Sakar, "Parkinson's disease classification data set," UCI Mach. Learn. Repository, School Inf. Comput. Sci., Univ. California, Irvine, CA, USA, 2018. Accessed: Jun. 20, 2024. [Online]. Available: https://archive.ics.uci.edu/dataset/470/parkinson+s+disease+classification

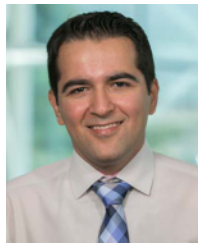


Hassan Gharoun received the master's degree in industrial engineering from the University of Tehran, Tehran, Iran. He is currently working toward the Ph.D. degree in the field of analytics with the University of Technology Sydney, NSW, Australia, where his doctoral research focuses on uncertainty-aware machine learning models. He applies data-driven approaches to address industrial challenges, particularly in healthcare. He was a Reviewer for esteemed journals such as IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS and IEEE INTERNET OF THINGS JOURNAL. His research interests include probabilistic ML and meta-learning.

Navid Yazdanjue received the Master of Science degree in information technology from the Iran University of Science and Technology (IUST), Tehran, Iran, in 2018. From 2019 to 2022, he was a Research and Teaching Assistant with IUST. He is currently working toward the Ph.D. degree with Digital Finance Cooperative Research Centre (DFCRC) Industry, Data Science Institute, University of Technology Sydney (UTS), Ultimo, NSW, Australia. He also collaborates with Cyber Intelligence House (CIH) Company, as an Industry Partner within the aforementioned DFCRC Industry Ph.D. Program. His research interests include meta-heuristic optimization techniques, evolutionary computation, machine learning models, social network analysis (SNA), and cybersecurity.

Mohammad Sadegh Khorshidi received the B.Sc. and M.Sc. degrees in civil engineering from Shiraz University, Shiraz, Iran, in 2014 and 2017, respectively. He is currently working toward the Ph.D. degree in information systems with the University of Technology Sydney (UTS), Ultimo, NSW, Australia, where his Doctoral research focuses on applying advanced data analytics, machine learning, and genetic programming. He has also been recognised for his contributions to the field with the DECRA Ph.D. Scholarship from Australian Research Council. He has authored or coauthored in various reputable journals. He is a Reviewer for journals such as *Environmental Research Letters*, *Journal of Hydrology*, and *Water Resources Management*.

Fang Chen (Member, IEEE) is currently a Distinguished Professor of data science with the Faculty of Engineering & Information Technology, University of Technology Sydney (UTS), Ultimo, NSW, Australia, where she is also the Executive Director with Data Science Institute. Prior to joining UTS, she was Dean of the Faculty with Beijing Jiaotong University, Beijing, China. She held Senior Leadership Positions with Intel, Motorola, and the Commonwealth Scientific and Industrial Research Organisation (CSIRO). She has authored or coauthored more than 400 peer-reviewed papers in science and engineering, along with several influential books. She has filed more than 30 patents across eight countries, including Australia, the US, Canada, Europe, Japan, Korea, Mexico, and China, showcasing her significant contributions to the field. Her research interests include developing innovative, data-driven solutions to complex challenges across large-scale networks in various sectors, such as transportation, water, energy, agriculture, telecommunications, education, health, financial services, real estate, and retail. She was the recipient of the multiple Prestigious awards for her research excellence and impact, including the Australian Museum Eureka Prize for Excellence in Data Science in 2018 (often referred to as the "Oscar" of Australian Science), NSW Premier's Prize for Science and Engineering in 2021, Australia and New Zealand "Women in AI" Award, and the "Brian Shackle Award" in 2017 from the International Federation for Information Processing (IFIP) for her outstanding contributions to human-computer interaction. She actively contributes to several boards and expert panels, including the Australian Federal Industry Science and Innovation Australia Board, NSW Government AI Review Board, and Singapore National Research Foundation's expert panel. She is also the Steering Committee Chair for ACM Intelligent User Interfaces and is also on the ITS Australia Board and various startup boards as a venture partner.



Amir H. Gandomi (Senior Member, IEEE) is currently a Professor of data science with the Faculty of Engineering & Information Technology, University of Technology Sydney, Ultimo NSW, Australia. He is also with Obuda University, Budapest, Hungary, as a Distinguished Professor. Prior to joining UTS, he was an Assistant Professor with the Stevens Institute of Technology and a Distinguished Research Fellow with BEACON Center, Michigan State University, East Lansing, MI, USA. He has authored or coauthored more than 300 journal papers and 12 books, which have collectively been cited more than 64,000 times (H-index=111). His research focuses on applied AI. He has been named one of the most influential scientific minds. In a recent study at Stanford University, released by Elsevier, Prof Amir H Gandomi is ranked 24th most impactful researcher in the AI and Image Processing subfield in 2023! He was the recipient of the awards for his research excellence and impact, such as the 2024 IEEE TCSC Award for Excellence in Scalable Computing (MCR), 2023 Achenbach Medal, 2022 Walter L. Huber Prize, highest-level mid-career research award in all areas of civil engineering, and the Highly Cited Researcher Award from Web of Science for six years. He was an Associate Editor, Editor, and the Guest Editor of several prestigious journals, such as AE of IEEE NETWORKS and IEEE INTERNET OF THINGS JOURNAL. He is active in delivering keynotes and invited talks.