bioengineering

MDPI

Article

A Machine Learning Method with Filter-Based Feature Selection for Improved Prediction of Chronic Kidney Disease

Sarah A. Ebiaredoh-Mienye [1], Theo G. Swart [1,*], Ebenezer Esenogho [1] and Ibomoiye Domor Mienye [2]

[1] Center for Telecommunications, Department of Electrical and Electronic Engineering Science, University of Johannesburg, Johannesburg 2006, South Africa; snabofa@yahoo.com (S.A.E.-M.); ebenezere@uj.ac.za (E.E.)
[2] Department of Electrical and Electronic Engineering Science, University of Johannesburg, Johannesburg 2006, South Africa; ibomoiyem@uj.ac.za
* Correspondence: tgswart@uj.ac.za

Abstract: The high prevalence of chronic kidney disease (CKD) is a significant public health concern globally. The condition has a high mortality rate, especially in developing countries. CKD often go undetected since there are no obvious early-stage symptoms. Meanwhile, early detection and on-time clinical intervention are necessary to reduce the disease progression. Machine learning (ML) models can provide an efficient and cost-effective computer-aided diagnosis to assist clinicians in achieving early CKD detection. This research proposed an approach to effectively detect CKD by combining the information-gain-based feature selection technique and a cost-sensitive adaptive boosting (AdaBoost) classifier. An approach like this could save CKD screening time and cost since only a few clinical test attributes would be needed for the diagnosis. The proposed approach was benchmarked against recently proposed CKD prediction methods and well-known classifiers. Among these classifiers, the proposed cost-sensitive AdaBoost trained with the reduced feature set achieved the best classification performance with an accuracy, sensitivity, and specificity of 99.8%, 100%, and 99.8%, respectively. Additionally, the experimental results show that the feature selection positively impacted the performance of the various classifiers. The proposed approach has produced an effective predictive model for CKD diagnosis and could be applied to more imbalanced medical datasets for effective disease detection.

Keywords: AdaBoost; chronic kidney disease; cost-sensitive learning; machine learning; medical diagnosis

Citation: Ebiaredoh-Mienye, S.A.; Swart, T.G.; Esenogho, E.; Mienye, I.D. A Machine Learning Method with Filter-Based Feature Selection for Improved Prediction of Chronic Kidney Disease. *Bioengineering* **2022**, *9*, 350. https://doi.org/10.3390/bioengineering9080350

Academic Editor: Chengfei Zhang

Received: 31 May 2022
Accepted: 21 July 2022
Published: 28 July 2022

1. Introduction

Chronic kidney disease is among the leading causes of death globally. A recent medical report states that approximately 324 million people suffer from CKD globally [1]. The glomerular filtration rate (GFR) is a widely used CKD screening test [2]. Though CKD affects people worldwide, it is more prevalent in developing countries [3]. Meanwhile, early detection is vital in reducing the progression of CKD. However, people from developing countries have not benefitted from early-stage CKD screening due to the cost of diagnosing the disease and limited healthcare infrastructure. While the global prevalence of CKD is reported to be 13.4% [4], it is said to have a 13.9% prevalence in Sub-Saharan Africa [5,6]. Another study reported a 16% pooled prevalence of CKD in West Africa [7], the highest in Africa. Numerous research works have specified that CKD is more prevalent in developing countries [8]. Notably, it is reported that 1 out of every 10 persons suffers from CKD in South Asia, including Pakistan, India, Bhutan, Bangladesh, and Nepal [3].

Therefore, several researchers have proposed machine learning (ML)-based methods for the early detection of CKD. These ML methods could provide effective, convenient, and low-cost computer-aided CKD diagnosis systems to enable early detection and intervention, especially in developing countries. Researchers have proposed different methods to detect CKD effectively using the CKD dataset [9] available at the University of California, Irvine

(UCI) machine learning repository. For example, Qin et al. [9] proposed an ML approach for the early detection of CKD. The approach involved using the k-Nearest Neighbor (KNN) imputation to handle the missing values in the dataset. After filling the missing data, six ML classifiers were trained and tested with the preprocessed data. The classifiers include logistic regression, SVM, random forest, KNN, naïve Bayes, and a feed-forward neural network. Due to the misclassification of these classifiers, the authors developed an integrated classifier that uses a perceptron to combine the random forest and logistic regression classifiers, which produced an enhanced accuracy of 99.83%.

Meanwhile, Ebiaredoh-Mienye et al. [10] proposed a method for improved medical diagnosis using an improved sparse autoencoder (SAE) network with a softmax layer. The neural network achieved sparsity through weight penalty, unlike the traditional sparse autoencoders that penalize the hidden layer activations. When used for CKD prediction, the proposed SAE achieved an accuracy of 98%. Chittora et al. [11] studied how to effectively diagnose CKD using ML methods. The research employed seven algorithms, including the C5.0 decision tree, logistic regression, linear support vector machine (LSVM) with $L1$ and $L2$ norms, artificial neural network, etc. The authors studied the performance of the selected classifiers when they were trained under different experimental conditions. These conditions include instances where the complete and reduced feature sets were used to train the classifiers. The experimental results showed that the LSVM with $L2$ norm trained with the reduced feature set obtained an accuracy of 98.46%, which outperformed the other classifiers.

Furthermore, Silveira et al. [12] developed a CKD prediction approach using a variety of resampling techniques and ML algorithms. The resampling techniques include the synthetic minority oversampling technique (SMOTE) and Borderline-SMOTE, while the classifiers include random forest, decision tree, and AdaBoost. The experimental results showed that the decision tree with the SMOTE technique achieved the best performance with 98.99%. Generally, these ML research works utilize many attributes such as albumin, hemoglobin level, white blood cell count, red blood cell count, packed cell volume, blood pressure, specific gravity, etc., to flag patients at risk of CKD, thereby allowing clinicians to provide early and cost-efficient medical intervention. Despite the attention given to CKD prediction using machine learning, only a few research works have focused on identifying the most relevant features needed to improve CKD detection [13–15]. If identified correctly in suspected CKD patients, these features could be utilized for efficient computer-aided CKD diagnosis.

In machine learning tasks, the algorithms employ discriminative abilities of features in classifying the samples. The ML models' performance relies not only on the specific training algorithm but also on the input data characteristics, such as the number of features and the correlation between the features [16]. Moreover, in most ML applications, especially in medical diagnosis, all the input features may not have equal importance. The goal of feature selection is to remove redundant attributes from the input data, ensuring the training algorithm learns the data more effectively. By removing non-informative variables, the computational cost of building the model is reduced, leading to faster and more efficient learning with enhanced classification performance.

Filter- and wrapper-based methods are the two widely used feature selection mechanisms [17]. Wrapper-based feature selection techniques use a classifier to build ML models with different predictor variables and select the variable subset that leads to the best model. In contrast, filter-based methods are statistical techniques independent of a learning algorithm used to compute the correlation between the predictor and independent variables [18]. The predictor variables are scored according to their relevance to the target variable. The variables with higher scores are then used to build the ML model. Therefore, this research aims to use information gain (IG), a filter-based feature selection method, to identify the most relevant features for improved CKD detection. IG is a robust algorithm for evaluating the gain of the various features with respect to the target variable [19]. The

attributes with the least IG values are removed, and those whose IG values are above a particular threshold are used to train the classifiers.

Meanwhile, a significant challenge in applying machine learning algorithms for medical diagnosis is the imbalanced class problem [20,21]. Most ML classifiers underperform when trained with imbalanced datasets. Class imbalance implies there is an uneven distribution of samples in each class. The class with the most samples is the majority class, while the class with the lesser samples is the minority class. Imbalance learning can be divided into data and algorithm-level approaches [22]. Data level methods are based on resampling techniques. Several studies have employed resampling techniques such as undersampling and oversampling to solve the class imbalance problem [23–25]. In order to create a balanced dataset, undersampling methods remove samples from the majority class, while oversampling techniques artificially create and add more data in the minority class. However, there are limitations to using these resampling techniques. For example, the samples discarded from the majority class could be vital in efficiently training the classifiers [20]. Therefore, several studies have resorted to using algorithm-level methods such as ensemble learning and cost-sensitive learning to effectively handle the imbalanced data instead of data-level techniques [26–29].

Ensemble learning is a breakthrough in ML research and application, which is used to obtain a very accurate classifier by combining two or more classifiers. Boosting [30] and Bagging [31] are widely used ensemble learning techniques. Adaptive boosting [30] is a type of boosting technique that creates many classifiers by assigning weights to the training data and adjusting these weights after every boosting cycle. The wrongly classified training instances are given higher weights in the next iteration, whereas the weight of correctly predicted examples is decreased. However, the AdaBoost algorithm does not treat the minority class and majority class weight updates differently when faced with imbalanced data. Therefore, in this study, we develop an AdaBoost classifier that gives higher weight to examples in the minority class, thereby enhancing the prediction of the minority class samples and the overall classification performance. A cost-sensitive classifier is obtained by biasing the weighting technique to focus more on the minority class. Recent findings have demonstrated that cost-sensitive learning is an efficient technique suitable for imbalanced classification problems [32–34]. The contribution of this study is to obtain the most important CKD attributes needed to improve the performance of CKD detection and develop a cost-sensitive AdaBoost classifier that gives more attention to samples in the minority class.

The rest of this paper is structured as follows: Section 2 presents the materials and methods, including an overview of the CKD dataset, the information gain technique, the traditional AdaBoost method, the proposed cost-sensitive AdaBoost, and the performance evaluation metrics. Section 3 presents the experimental results and discussion, while Section 4 concludes the paper.

2. Materials and Methods

This section provides an overview of the CKD dataset and the various methods used in the research. In particular, a detailed overview of the traditional AdaBoost algorithm and the proposed cost-sensitive AdaBoost is presented, thereby showing the difference between both methods.

2.1. Dataset

This study utilizes the CKD dataset prepared in 2015 by Apollo Hospitals, Tamil Nadu, India. The dataset is publicly available at the University of California, Irvine (UCI) machine learning repository [35]. It contains medical test results and records from 400 patients; 250 correspond to patients with CKD, and 150 correspond to patients without CKD, so the dataset is imbalanced. There are 24 independent variables (11 numerical and 13 nominal) and a class variable (ckd or notckd). The attributes and their corresponding descriptions are shown in Table 1.

Table 1. CKD dataset description.

No.	Attribute	Description	Category	Scale
f1	age	Age of the patient	Numerical	age in years
f2	bp	Blood pressure	Numerical	mm/Hg
f3	sg	Specific gravity	Nominal	1.005, 1.010, 1.015, 1.020, 1.025
f4	al	Albumin	Nominal	0, 1, 2, 3, 4, 5
f5	su	Sugar	Nominal	0, 1, 2, 3, 4, 5
f6	rbc	Red blood cells	Nominal	normal, abnormal
f7	pc	Pus cell	Nominal	normal, abnormal
f8	pcc	Pus cell clumps	Nominal	present, not present
f9	ba	Bacteria	Nominal	present, not present
f10	bgr	Blood glucose random	Numerical	mgs/dl
f11	bu	Blood urea	Numerical	mgs/dl
f12	sc	Serum creatinine	Numerical	mgs/dl
f13	sod	Sodium	Numerical	mEq/L
f14	pot	Potassium	Numerical	mEq/L
f15	hemo	Hemoglobin	Numerical	gms
f16	pcv	Packed cell volume	Numerical	-
f17	wc	White blood cell count	Numerical	cells/cumm
f18	rc	Red blood cell count	Numerical	millions/cmm
f19	htn	Hypertension	Nominal	yes, no
f20	dm	Diabetes mellitus	Nominal	yes, no
f21	cad	Coronary artery disease	Nominal	yes, no
f22	appet	Appetite	Nominal	good, poor
f23	pe	Pedal edema	Nominal	yes, no
f24	ane	Anemia	Nominal	yes, no
f25	class	Class	Nominal	ckd, notckd

Some of the features in Table 1 are briefly described as follows: Specific gravity estimates the concentration of particles in the urine and the urine's density relative to the density of water. It indicates the hydration status of a patient together with the functional ability of the patient's kidney. Albumin is a protein found in the blood [36]. When the kidney is damaged, it allows albumin into the urine. Higher albumin levels in the urine could indicate the presence of CKD. Meanwhile, blood urea indicates vital information about the functionality of the kidney. A blood urea nitrogen test measures the quantity of urea nitrogen in a patient's blood, and a high amount implies the kidneys are not functioning normally. While a random blood glucose test measures the amount of sugar circulating in a patient's blood, and a level of 200 mg/dL or above implies the patient has diabetes. Serum creatinine is a waste product produced by a person's muscles [37]. A creatinine test measures the creatinine levels in the blood or urine, and high levels of the substance imply the kidney is not functioning well enough to filter the waste from the blood.

Furthermore, sodium is an electrolyte in the blood that helps the muscles and nerves work effectively. A sodium blood test measures the amount of sodium in the patient's blood, and a very high or low amount may indicate a kidney problem, dehydration, or other medical condition. Potassium is another electrolyte in the blood, and a very high or low amount could signal the presence of an underlying condition. White blood cells (WBC) protect the human body from invading pathogens. They are part of the body's immune system, protecting it from infections [38]. The normal range is between 4000 and 11,000 per microliter of blood. Elevated WBC count is a popular indicator of the progression of CKD. Red blood cells (RBC) in humans deliver oxygen to the body tissues. The average RBC count is 4.7 to 6.1 million cells per microliter for men and 4.2 to 5.4 million cells per microliter for women. A low RBC, also called anemia, is a common complication of CKD.

Meanwhile, the data needs to be preprocessed to make it suitable for machine learning. Therefore, all the nominal or categorical data were coded. Specifically, the attributes whose scales are 'normal' and 'abnormal' were transformed to 1 and 0, respectively. The attributes whose scales are 'present' and 'not present' were transformed to 1 and 0, respectively.

Additionally, the 'yes' and 'no' scales were coded to 1 and 0, respectively. Lastly, the attribute with 'good' and 'poor' scales was transformed to 1 and 0, respectively.

Furthermore, the dataset contains a few missing values. It is vital to appropriately deal with missing values before building ML models because ignoring or deleting the missing data could degrade or bias the performance of the models [39,40]. Imputation is a method used to estimate and fill missing values in a dataset. Since the number of missing values in our dataset is not large, the mean imputation technique is used to handle the missing values. The mean imputation technique computes the mean of the observed values for each variable, and the missing values are filled with the corresponding computed mean value [41]. Meanwhile, except for the 'age' and binary attributes, the remaining attributes were scaled to have values between 0 and 1 using the Min–Max Scaling technique [42].

Additionally, the clinicians at Apollo Hospitals, Tamil Nadu, India, categorized attributes as normal or abnormal, present or not present, yes or no. The clinicians selected the 24 attributes representing the patient's medical tests and records associated with chronic kidney disease. However, the attributes do not carry equal weights in diagnosing a patient with CKD, and some attributes are more indicative of the presence of CKD than others. Additionally, certain attributes might be redundant and could increase the complexity of the ML model [43]. Hence, this research employs the information gain technique to rank the attributes according to their relevance in detecting the disease, and only the most relevant attributes are used to build the ML model.

2.2. Information Gain

Effective feature selection could remove attributes that are less useful in obtaining an excellent predictive model. Additionally, it is necessary to remove attributes unrelated to the target variable because these attributes could increase the computational cost and prevent the model from obtaining optimal performance [44]. This study utilizes the information gain (IG) technique to extract the optimal features. IG is a type of filter-based feature selection that calculates the predictor variable's ability to classify the dependent variable [45]. The IG method has its roots in information theory, and it calculates the statistical dependence between two variables. Mathematically, the IG between two variables X and Y is formulated as:

$$\mathrm{IG}(X|Y) = H(X) - H(X|Y), \tag{1}$$

where $H(X)$ is the entropy for variable X and $H(X|Y)$ represents the conditional entropy for X given Y. Computing the IG value for an attribute involves calculating the entropy of the target variable for the whole dataset and subtracting the conditional entropies for every potential value of that attribute [46]. Furthermore, the entropy $H(X)$ and conditional entropy $H(X|Y)$ are computed as:

$$H(X) = -\sum_{x \in X} P(x) log_2(x), \tag{2}$$

$$H(X|Y) = -\sum_{x \in X} P(x) \sum_{y \in Y} P(x|y) log_2(P(x|y)), \tag{3}$$

Hence, given two variables X and Z, a given variable Y is said to have a more significant correlation to X than Z if $\mathrm{IG}(X|Y) > \mathrm{IG}(Z|Y)$. Furthermore, IG considers every attribute in isolation, calculates its information gain, and computes its relevance to the target variable.

2.3. AdaBoost Algorithm

The AdaBoost algorithm is an ML technique derived from the concept of boosting. The boosting technique primarily entails converting weak learners into strong learners [47]. Freund and Schapire [48] proposed the AdaBoost algorithm to iteratively train multiple learning classifiers using the same training dataset. After training the weak learners, they are combined to obtain a strong classifier. The AdaBoost procedure involves selecting an appropriate weak learner and employing the same training dataset to train the weak

learner iteratively to enhance its performance, as shown in Algorithm 1. Two weights are utilized in implementing the AdaBoost algorithm; the first is the sample weight, and the second is the weight of every weak learner [49]. The algorithm adjusts the sample weight depending on the weak classifier's result, thereby giving more attention to wrongly classified samples. Subsequent base learners are trained with the adjusted samples [50]. The final strong classifier is obtained by combining the output of the weak learners using a weighted sum [51]. The AdaBoost is adaptive because subsequent weak classifiers are trained to pay more attention to the samples that were wrongly classified by preceding classifiers.

Algorithm 1: Conventional AdaBoost technique

Input: training dataset $S = \{(x_1, y_1), \ldots, (x_2, y_2), \ldots, (x_n, y_n)\}$, base learner h, the number of training rounds T.
Output: the final strong classifier H.
Procedure:

1. ***for*** $i = 1 : 1 : n$
2. compute the weight of the sample x_i: $D_1(i) = \frac{1}{n}$
3. ***end for***
4. ***for*** $t = 1 : 1 : T$
5. select a training data subset X from S, fit h using X to get a weak classifier h_t, compute the classification error ε_t: $\varepsilon_t = P[h_t(x_i) \neq y_i] = \sum_{i=1}^{n} D_t(i) \mathrm{I}[h_t(x_i) \neq y_i]$ where $h_t(x_i)$ denotes the predicted label of x_i using the weak classifier h_t, and y_i denotes the actual label of x_i.
6. compute the weight of h_t : $\alpha_t = \frac{1}{2} \ln\left(\frac{1-\varepsilon_t}{\varepsilon_t}\right)$
7. update the weight of all the instances in S: ***for*** $i = 1 : 1 : n$ $D_{t+1}(i) = \frac{D_t(i)}{Z_t} \exp(-\alpha_t y_i h_t(x_i))$ where Z_t is a normalization factor and is calculated as: $Z_t = \sum_{i=1}^{n} D_t(i) \exp(-\alpha_t y_i h_t(x_i))$
8. ***end for***
9. ***end for***
10. assuming $H(x)$ is the class label for an instance x; after the iterations, the final classifier H is obtained as: $H(x) = sign\left(\sum_{t=1}^{T} \alpha_t h_t(x)\right)$

2.4. Proposed Cost-Sensitive AdaBoost

At every iteration, the AdaBoost algorithm increases the weights of the misclassified training samples and decreases that of the samples that were predicted correctly. This weighting method distinguishes instances as correctly or wrongly classified, and the examples from both classes are treated equally. For example, the weights of incorrectly classified instances from both classes are increased by a similar ratio. The weights of the samples that were predicted correctly from both classes are decreased by a similar ratio [52]. However, in imbalance classification, the goal is to improve the classifier's prediction performance on the minority class. Hence, a suitable weighting approach will identify the different types of instances and give more weight to the samples with greater detection importance, i.e., the minority class.

From the AdaBoost learning method (Algorithm 1), cost items (β_i) are added to the weight update equation to bias the weighting technique. The rate of correct and incorrect predictions of the various classes are included as part of β_i. The cost of false positive is denoted as c_{10} and the cost of false negative is denoted as c_{01}, while the cost of true positive and true negative are denoted as c_{11} and c_{00}, respectively [53]. The weight update in the traditional AdaBoost takes into account the overall error rate only [54]. However, in the cost-sensitive AdaBoost, the error rate of each class is considered. In this new weighting strategy, the weights of the minority class examples are higher compared to the majority class examples. The new cost-sensitive AdaBoost is presented in Algorithm 2.

Algorithm 2: Cost-Sensitive AdaBoost

Input: training dataset $S = \{(x_1, y_1), \ldots, (x_2, y_2), \ldots, (x_n, y_n)\}$, base learner h, the number of iterations T.
Output: the final strong classifier H.
Procedure:

1. ***for*** $i = 1 : 1 : n$
2. compute the weight of the sample x_i: $D_1(i) = \frac{1}{n}$
3. ***end for***
4. ***for*** $t = 1 : 1 : T$
5. select a training data subset X from S, fit h using X to get a weak classifier h_t
6. let $n+$ and $n-$ indicate the positive and negative classes, respectively. Compute the error rate ε_t of the base learner for both the positive class ε_t^p and negative class ε_t^n: $\varepsilon_t = \left(\frac{\varepsilon_t^p + \varepsilon_t^n}{2}\right)$, where $\varepsilon_t^p = P[h_t(x_i) \neq y_i] = \sum_{i=1}^{n+} D_t(i)\mathrm{I}[h_t(x_i) \neq y_i]$ and $\varepsilon_t^n = P[h_t(x_i) \neq y_i] = \sum_{i=1}^{n-} D_t(i)\mathrm{I}[h_t(x_i) \neq y_i]$
7. compute the weight of h_t: $\alpha_t = \frac{1}{2}\ln\left(\frac{1-\varepsilon_t}{\varepsilon_t}\right)$
8. update the weight of all the instances in S: ***for*** $i = 1 : 1 : n$
   $D_{t+1}(i) = \frac{D_t(i)}{Z_t}\exp(-\alpha_t\beta_i y_i h_t(x_i))$ where Z_t is a normalization factor and is calculated as: $Z_t = \sum_{i=1}^{n} D_t(i)\exp(-\alpha_t\beta_i y_i h_t(x_i))$ and β_i is calculated as:
   $$\beta_i = \begin{cases} \frac{TP_t}{FP_t+TP_t}c_{10}, \; if \; y_i = 1, \; h_t(x_i) = -1 \\ \frac{TN_t}{FN_t+TN_t}c_{01}, \; if \; y_i = -1, \; h_t(x_i) = 1 \\ \frac{TP_t}{FP_t+TP_t}c_{11}, \; if \; y_i = 1, \; h_t(x_i) = 1 \\ \frac{TN_t}{FN_t+TN_t}c_{00}, \; if \; y_i = -1, \; h_t(x_i) = -1 \end{cases}$$ where TP_t, TN_t, FP_t, FN_t are true positive, true negative, false positive, and false negative values for iteration t. Meanwhile, c_{10}, c_{01}, c_{11}, c_{00} are the cost-sensitive factors, where $c_{10} > c_{00}$ and $c_{01} > c_{11}$.
9. ***end for***
10. ***end for***
11. the final classifier is obtained as follows: $H(x) = sign\left(\sum_{t=1}^{T}\alpha_t h_t(x)\right)$

By giving higher weights to samples in the minority class, the weak classifiers tend to focus more on the misclassification of examples in that class, thereby accurately classifying more instances at each iteration. Therefore, the final strong classifier will obtain more correct predictions. The architecture of the proposed approach is shown in Figure 1.

2.5. Performance Evaluation Metrics

The dataset used in this research comprises two classes, ckd and notckd classes. The ckd-labeled data are the positive patients, while the notckd-labeled data are the negative patients. Meanwhile, accuracy (ACC), sensitivity (SEN), and specificity (SPE) are used to assess the performance of the classifiers. Accuracy is the total number of correct predictions divided by the total number of predictions made by the classifier. Sensitivity or true positive rate is the number of correct positive predictions divided by the number of positive cases in the dataset; it measures the ability of the classifier to correctly detect those with the disease [55]. Specificity measures the classifier's ability to correctly identify people without the disease, i.e., negative instances. These metrics are computed as follows:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}, \tag{4}$$

$$\text{Sensitivity} = \frac{TP}{TP + FN}, \tag{5}$$

$$\text{Specificity} = \frac{TN}{TN + FP}, \tag{6}$$

where true positive (TP) represents the ckd instances that were correctly classified, and false-negative (FN) represents the ckd instances that were wrongly classified. True negative (TN) denotes the notckd samples that were correctly classified, and false-positive (FP) indicates the notckd instances that were wrongly classified [56]. Additionally, this research utilizes the receiver operating characteristic (ROC) curve and the area under the ROC curve (AUC) to further evaluate the classifiers' performance. The ROC curve is a plot of the true positive rate (TPR) against the false positive rate (FPR) at different threshold values. It shows the ability of the classifier to distinguish between the ckd and notckd classes. Meanwhile, the AUC is mainly utilized to summarize the ROC curve, and it has a value between 0 and 1 that shows the classifiers' ability to differentiate between both classes [57]. The higher the AUC value, the better the classifiers can distinguish between the ckd and notckd classes.

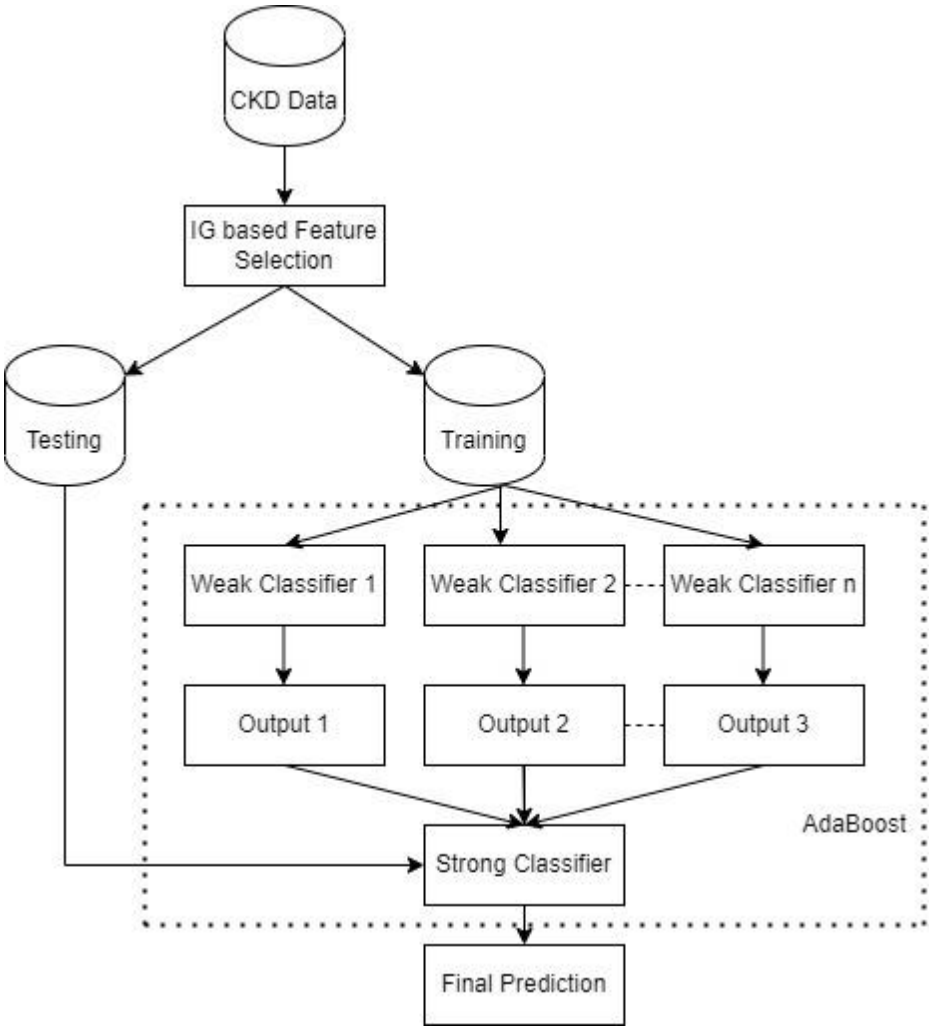

Figure 1. The architecture of the proposed approach.

3. Results

In this section, the experimental results are presented and discussed. All experiments used the preprocessed data, as discussed in Section 2.1, and the ML models were developed using scikit-learn [58], a machine learning library for Python programming. The experiments were conducted using a computer with the following specifications: Intel(R) Core(TM) i7-113H @ 3.30 GHz, 4 Core(s), and 16 GB RAM. Furthermore, to have a baseline for comparing the proposed cost-sensitive AdaBoost (CS AdaBoost), this research implements the traditional AdaBoost [30] presented in Algorithm 1 and other well-known classifiers, including logistic regression [59], decision tree [60], XGBoost [61], random

forest [62], and SVM [63]. The classifiers are trained with the complete feature set and the reduced feature set to demonstrate the impact of the feature selection. Meanwhile, the 10-fold cross-validation method is employed to evaluate the performance of the various models. The decision tree algorithm is the base learner for the AdaBoost and CS AdaBoost implementations.

3.1. Performance of the Classifiers without Feature Selection

This subsection presents the experimental results obtained when the complete feature set was used to train the various classifiers. These results are tabulated in Table 2. Additionally, Figure 2 shows the AUC values and the ROC curves of the different classifiers.

Table 2. Performance of the classifiers trained with the complete feature set.

Classifier	ACC	SEN	SPE	AUC
Logistic regression	0.940	0.948	0.933	0.940
Decision tree	0.902	0.932	0.890	0.910
XGBoost	0.958	0.964	0.942	0.970
Random forest	0.952	0.955	0.940	0.960
SVM	0.937	0.943	0.930	0.940
AdaBoost	0.930	0.941	0.935	0.950
Proposed CS AdaBoost	0.967	0.975	0.960	0.980

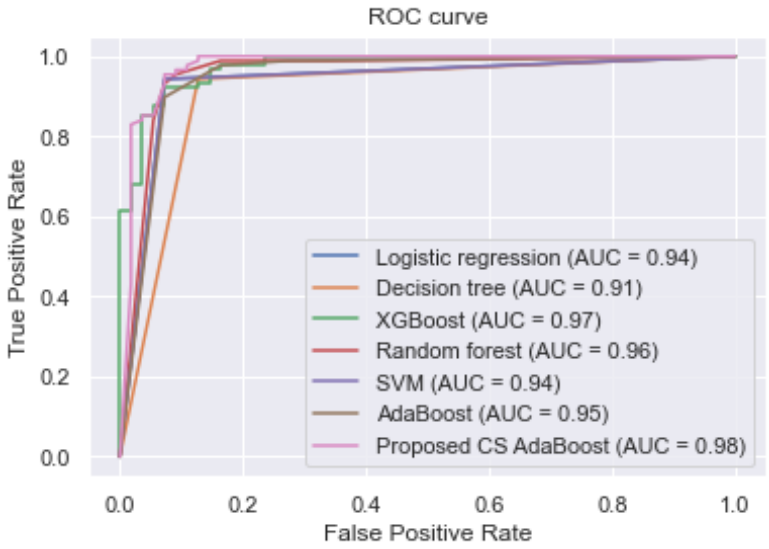

Figure 2. ROC curve of the classifiers trained using the complete feature set.

Table 2 and Figure 2 show that the proposed cost-sensitive AdaBoost obtained excellent performance by outperforming the traditional AdaBoost and the other classifiers, having obtained an AUC, accuracy, sensitivity, and specificity of 0.980, 0.967, 0.975, and 0.960, respectively.

3.2. Performance of the Classifiers after Feature Selection

The information-gain-based feature selection ranked the chronic kidney disease attributes. This step aims to select the features with the highest information gain with respect to the target variable. The ranked features and their IG values are shown in Table 3. After obtaining the IG values of the various features, the standard deviation [19] of the values is computed, which serves as the threshold value for the feature selection. The standard deviation measure has been used in recent research to obtain a reasonable threshold for feature selection [19,64,65]. The threshold value obtained is 0.156. Therefore, the IG values equal to or greater than 0.156 are selected as the informative features and used for building the models. In contrast, the attributes with IG values lower than the threshold are discarded.

Hence, from Table 3, the top 18 features are selected as the optimal feature set since their IG values (rounded to three decimal values) are greater than 0.156, and the following features are discarded: f22, f5, f24, f8, f9, and f21.

Table 3. Feature ranking.

No.	Feature Name	IG Value
f4	al	0.598
f15	hemo	0.581
f16	pcv	0.526
f18	rc	0.482
f12	sc	0.474
f10	bgr	0.422
f3	sg	0.392
f11	bu	0.389
f13	sod	0.344
f17	wc	0.325
f19	htn	0.270
f14	pot	0.266
f1	age	0.253
f7	pc	0.251
f20	dm	0.215
f2	bp	0.209
f6	rbc	0.206
f23	pe	0.184
f22	appet	0.155
f5	su	0.135
f24	ane	0.128
f8	pcc	0.097
f9	ba	0.069
f21	cad	0.065

To demonstrate the effectiveness of the feature selection, the reduced feature set is used to train the proposed CS AdaBoost and the other classifiers. The experimental results are shown in Table 4. Additionally, the ROC curve and the various AUC values are shown in Figure 3. The experimental results in Table 4 and Figure 3 show that the proposed CS AdaBoost obtained an AUC, accuracy, sensitivity, and specificity of 0.990, 0.998, 1.000, and 0.998, respectively, which outperformed the logistic regression, decision tree, XGBoost, random forest, SVM, and conventional AdaBoost. Secondly, it is observed that the performance of the various classifiers in Table 4 is better than their corresponding performance in Table 2. This improvement demonstrates the effectiveness of the feature selection step. Therefore, the combination of feature selection and cost-sensitive AdaBoost is an effective method for predicting CKD.

Table 4. Performance of the classifiers trained with the reduced feature set.

Classifier	ACC	SEN	SPE	AUC
Logistic regression	0.961	0.959	0.961	0.970
Decision tree	0.940	0.935	0.948	0.940
XGBoost	0.989	0.990	0.986	0.990
Random forest	0.977	0.981	0.973	0.980
SVM	0.954	0.957	0.961	0.960
AdaBoost	0.964	0.960	0.968	0.970
Proposed CS AdaBoost	0.998	1.000	0.998	0.990

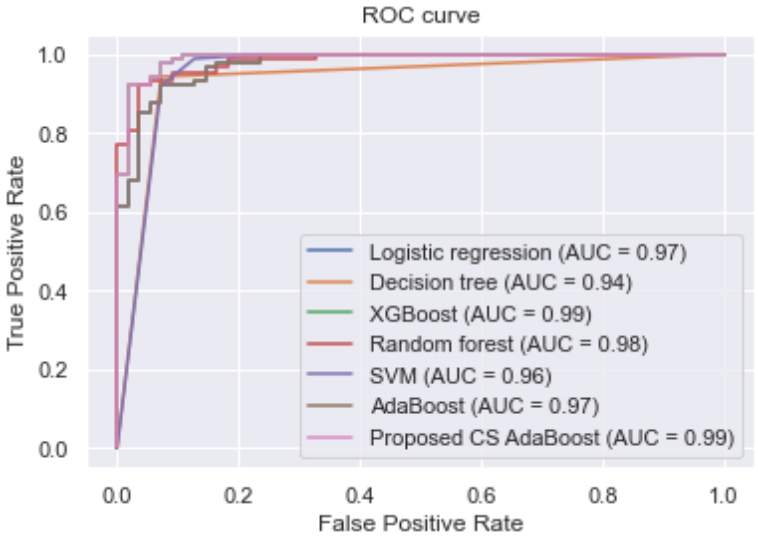

Figure 3. ROC curve of the classifiers trained with the reduced feature set.

3.3. Comparison with Other CKD Prediction Studies

Even though the proposed approach showed superior performance to the other algorithms, it is not enough to conclude its robustness. It is, however, necessary to compare it with other state-of-the-art methods in the literature. Hence, the proposed approach is compared with the following methods: a probabilistic neural network (PNN) [66], an enhanced sparse autoencoder (SAE) neural network [10], a naïve Bayes (NB) classifier with feature selection [67], a feature selection method based on cost-sensitive ensemble and random forest [3], a linear support vector machine (LSVM) and synthetic minority oversampling technique (SMOTE) [11], a cost-sensitive random forest [68], a feature selection method based on recursive feature elimination (RFE) and artificial neural network (ANN) [69], a correlation-based feature selection (CFS) and ANN [69]. The other methods include optimal subset regression (OSR) and random forest [9], an approach to identify the essential CKD features using improved linear discriminant analysis (LDA) [13], a deep belief network (DBN) with Softmax classifier [70], a random forest (RF) classifier with feature selection (FS) [71], a model based on decision tree and the SMOTE technique [12], a logistic regression (LR) classifier with recursive feature elimination (RFE) technique [14], and an XGBoost model with a feature selection approach combining the extra tree classifier (ETC), univariate selection (US), and RFE [15].

The proposed approach based on cost-sensitive AdaBoost and feature selection achieved excellent performance compared to several state-of-the-art methods in the literature, as shown in Table 5.

3.4. Discussion

This study aimed to solve two problems: first, to select the most informative features to enable the effective detection of CKD. The second aim was to develop an effective cost-sensitive AdaBoost classifier that accurately classifies samples in the minority class. The use of more features than necessary sometimes affects ML classifiers' performance and increases the computational cost of training the models. Hence, this research employed the IG-based feature selection method to obtain the optimal feature set. Furthermore, seven classifiers were used in this study, trained using the complete and the reduced feature sets. From the experimental results, the proposed framework showed improved classification performance with the reduced feature set, i.e., 18 out of 24 input variables. Additionally, the models trained with the reduced feature set performed better than those trained with the complete feature set. Remarkably, the proposed method obtained higher performance than the other classifiers.

Table 5. Comparison with other studies.

Reference	Method	ACC	SEN	SPE	AUC
Rady and Anwar [66]	PNN	0.969	0.987	0.964	-
Ebiaredoh-Mienye et al. [10]	SAE	0.980	0.970	-	-
Almustafa [67]	NB and FS	0.976	0.988	-	0.989
Ali et al. [3]	Cost-sensitive ensemble with RF	0.967	0.986	0.935	0.982
Chittora et al. [11]	LSVM and SMOTE	0.988	1.000	-	-
Mienye and Sun [68]	Cost-sensitive RF	0.986	1.000	-	-
Akter et al. [69]	RFE and ANN	0.970	0.980	-	0.980
Akter et al. [69]	CFS and ANN	0.960	0.970	-	0.970
Qin et al. [9]	OSR and RF	0.995	0.993	-	-
Nishanth and Thiruvaran [13]	Enhanced LDA	0.980	0.960	-	-
Elkholy et al. [70]	DBN with Softmax classifier	0.985	0.875	-	-
Rashed-Al-Mahfuz et al. [71]	RF and FS	0.990	0.979	0.996	0.987
Silveira et al. [12]	Decision tree and SMOTE	0.989	0.990	-	-
Motwani et al. [14]	LR and RFE	0.983	0.990	-	-
Ogunleye and Wang [15]	XGBoost and ETC-US-RFE	0.976	1.000	0.917	0.990
This paper	Proposed CS AdaBoost	0.998	1.000	0.998	0.990

Furthermore, the features selected by the IG technique were similar to current medical practices. For example, the IG technique ranked albumin, hemoglobin, packed cell volume, red blood cell count, and serum creatinine as the most informative features, and numerous studies have identified a strong correlation between these variables and chronic kidney disease [71–74].

Meanwhile, the class imbalance problem is common in most real-world classification tasks. Another objective of this study was to develop a robust classifier to prevent the misclassification of the minority class that occurs when classifiers are trained using imbalanced data. Hence, this study developed a cost-sensitive AdaBoost classifier, giving more attention to the minority class. The experimental results indicate that the proposed method achieved a higher classification performance than the baseline classifiers and techniques in recent literature. Secondly, the results demonstrate that the combination of the information-gain-based feature selection and the cost-sensitive AdaBoost classifier significantly improved the detection of chronic kidney disease.

4. Conclusions

This paper proposed an approach that combines information-gain-based feature selection and a cost-sensitive AdaBoost classifier to improve the detection of chronic kidney disease. Six other machine learning classifiers were implemented as the baseline for performance comparison. The classifiers include logistic regression, decision tree, random forest, SVM, XGBoost, and the traditional AdaBoost. Firstly, the IG technique was used to compute and rank the importance of the various attributes. Secondly, the classifiers were trained with the reduced and complete feature sets. The experimental results show that selected features enhanced the performance of the classifiers.

Furthermore, the proposed cost-sensitive AdaBoost achieved superior performance to the other classifiers and methods in recent literature. Therefore, combining the IG-based feature selection technique and cost-sensitive AdaBoost is an effective approach for CKD detection and can be potentially applied for early detection of CKD through computer-aided diagnosis. Future research will focus on collecting large amounts of data to train ML models, including datasets that allow for the prediction of the disease severity, duration of the disease, and the age of onset.

Author Contributions: Conceptualization, S.A.E.-M., T.G.S. and E.E.; methodology, S.A.E.-M.; software, S.A.E.-M. and I.D.M.; validation, S.A.E.-M. and I.D.M.; formal analysis, S.A.E.-M.; resources, T.G.S.; data curation, S.A.E.-M.; writing—original draft preparation, S.A.E.-M.; writing—review and editing, T.G.S. and E.E.; visualization, S.A.E.-M.; supervision, T.G.S. and E.E.; funding acquisition, T.G.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Bhaskar, N.; Suchetha, M.; Philip, N.Y. Time Series Classification-Based Correlational Neural Network With Bidirectional LSTM for Automated Detection of Kidney Disease. *IEEE Sens. J.* **2021**, *21*, 4811–4818. [CrossRef]
2. Sobrinho, A.; Queiroz, A.C.M.D.S.; Dias Da Silva, L.; De Barros Costa, E.; Eliete Pinheiro, M.; Perkusich, A. Computer-Aided Diagnosis of Chronic Kidney Disease in Developing Countries: A Comparative Analysis of Machine Learning Techniques. *IEEE Access* **2020**, *8*, 25407–25419. [CrossRef]
3. Ali, S.I.; Bilal, H.S.M.; Hussain, M.; Hussain, J.; Satti, F.A.; Hussain, M.; Park, G.H.; Chung, T.; Lee, S. Ensemble Feature Ranking for Cost-Based Non-Overlapping Groups: A Case Study of Chronic Kidney Disease Diagnosis in Developing Countries. *IEEE Access* **2020**, *8*, 215623–215648. [CrossRef]
4. Lv, J.-C.; Zhang, L.-X. Prevalence and Disease Burden of Chronic Kidney Disease. In *Renal Fibrosis: Mechanisms and Therapies*; Liu, B.-C., Lan, H.-Y., Lv, L.-L., Eds.; Advances in Experimental Medicine and Biology; Springer: Singapore, 2019; pp. 3–15. ISBN 9789811388712.
5. Chothia, M.Y.; Davids, M.R. Chronic kidney disease for the primary care clinician. *South Afr. Fam. Pract.* **2019**, *61*, 19–23.
6. Stanifer, J.W.; Jing, B.; Tolan, S.; Helmke, N.; Mukerjee, R.; Naicker, S.; Patel, U. The epidemiology of chronic kidney disease in sub-Saharan Africa: A systematic review and meta-analysis. *Lancet Glob. Health* **2014**, *2*, e174–e181. [CrossRef]
7. Olanrewaju, T.O.; Aderibigbe, A.; Popoola, A.A.; Braimoh, K.T.; Buhari, M.O.; Adedoyin, O.T.; Kuranga, S.A.; Biliaminu, S.A.; Chijioke, A.; Ajape, A.A.; et al. Prevalence of chronic kidney disease and risk factors in North-Central Nigeria: A population-based survey. *BMC Nephrol.* **2020**, *21*, 467. [CrossRef]
8. Varughese, S.; Abraham, G. Chronic Kidney Disease in India: A Clarion Call for Change. *Clin. J. Am. Soc. Nephrol.* **2018**, *13*, 802–804. [CrossRef]
9. Qin, J.; Chen, L.; Liu, Y.; Liu, C.; Feng, C.; Chen, B. A Machine Learning Methodology for Diagnosing Chronic Kidney Disease. *IEEE Access* **2020**, *8*, 20991–21002. [CrossRef]
10. Ebiaredoh-Mienye, S.A.; Esenogho, E.; Swart, T.G. Integrating Enhanced Sparse Autoencoder-Based Artificial Neural Network Technique and Softmax Regression for Medical Diagnosis. *Electronics* **2020**, *9*, 1963. [CrossRef]
11. Chittora, P.; Chaurasia, S.; Chakrabarti, P.; Kumawat, G.; Chakrabarti, T.; Leonowicz, Z.; Jasiński, M.; Jasiński, Ł.; Gono, R.; Jasińska, E.; et al. Prediction of Chronic Kidney Disease—A Machine Learning Perspective. *IEEE Access* **2021**, *9*, 17312–17334. [CrossRef]
12. Silveira, A.C.M.D.; Sobrinho, Á.; Silva, L.D.D.; Costa, E.D.B.; Pinheiro, M.E.; Perkusich, A. Exploring Early Prediction of Chronic Kidney Disease Using Machine Learning Algorithms for Small and Imbalanced Datasets. *Appl. Sci.* **2022**, *12*, 3673. [CrossRef]
13. Nishanth, A.; Thiruvaran, T. Identifying Important Attributes for Early Detection of Chronic Kidney Disease. *IEEE Rev. Biomed. Eng.* **2018**, *11*, 208–216. [CrossRef]
14. Motwani, A.; Shukla, P.K.; Pawar, M. Novel Machine Learning Model with Wrapper-Based Dimensionality Reduction for Predicting Chronic Kidney Disease Risk. In Proceedings of the Soft Computing and Signal Processing, Hyderabad, India, 18–19 June 2021; Reddy, V.S., Prasad, V.K., Wang, J., Reddy, K.T.V., Eds.; Springer: Singapore, 2021; pp. 29–37.
15. Ogunleye, A.; Wang, Q.-G. Enhanced XGBoost-Based Automatic Diagnosis System for Chronic Kidney Disease. In Proceedings of the 2018 IEEE 14th International Conference on Control and Automation (ICCA), Anchorage, AK, USA, 12–15 June 2018; pp. 805–810.
16. Haq, A.U.; Zhang, D.; Peng, H.; Rahman, S.U. Combining Multiple Feature-Ranking Techniques and Clustering of Variables for Feature Selection. *IEEE Access* **2019**, *7*, 151482–151492. [CrossRef]
17. Tadist, K.; Najah, S.; Nikolov, N.S.; Mrabti, F.; Zahi, A. Feature selection methods and genomic big data: A systematic review. *J. Big Data* **2019**, *6*, 79. [CrossRef]
18. Pirgazi, J.; Alimoradi, M.; Esmaeili Abharian, T.; Olyaee, M.H. An Efficient hybrid filter-wrapper metaheuristic-based gene selection method for high dimensional datasets. *Sci. Rep.* **2019**, *9*, 18580. [CrossRef]
19. Prasetiyowati, M.I.; Maulidevi, N.U.; Surendro, K. Determining threshold value on information gain feature selection to increase speed and prediction accuracy of random forest. *J. Big Data* **2021**, *8*, 84. [CrossRef]
20. Shaw, S.S.; Ahmed, S.; Malakar, S.; Sarkar, R. An Ensemble Approach for Handling Class Imbalanced Disease Datasets. In Proceedings of the Proceedings of International Conference on Machine Intelligence and Data Science Applications, Dehradun, India, 4–5 September 2020; Prateek, M., Singh, T.P., Choudhury, T., Pandey, H.M., Gia Nhu, N., Eds.; Springer: Singapore, 2021; pp. 345–355.
21. Aruleba, K.; Obaido, G.; Ogbuokiri, B.; Fadaka, A.O.; Klein, A.; Adekiya, T.A.; Aruleba, R.T. Applications of Computational Methods in Biomedical Breast Cancer Imaging Diagnostics: A Review. *J. Imaging* **2020**, *6*, 105. [CrossRef]

22. Zhang, C.; Tan, K.C.; Li, H.; Hong, G.S. A Cost-Sensitive Deep Belief Network for Imbalanced Classification. *IEEE Trans. Neural Netw. Learn. Syst.* **2019**, *30*, 109–122. [CrossRef]
23. Asniar; Maulidevi, N.U.; Surendro, K. SMOTE-LOF for noise identification in imbalanced data classification. *J. King Saud. Univ. Comput. Inf. Sci.* **2022**, *34*, 3413–3423. [CrossRef]
24. Xu, Z.; Shen, D.; Nie, T.; Kou, Y. A hybrid sampling algorithm combining M-SMOTE and ENN based on Random forest for medical imbalanced data. *J. Biomed. Inform.* **2020**, *107*, 103465. [CrossRef]
25. Hasanin, T.; Khoshgoftaar, T.M.; Leevy, J.L.; Bauder, R.A. Severely imbalanced Big Data challenges: Investigating data sampling approaches. *J. Big Data* **2019**, *6*, 107. [CrossRef]
26. Khan, S.H.; Hayat, M.; Bennamoun, M.; Sohel, F.A.; Togneri, R. Cost-Sensitive Learning of Deep Feature Representations From Imbalanced Data. *IEEE Trans. Neural Netw. Learn. Syst.* **2018**, *29*, 3573–3587. [PubMed]
27. Ma, Y.; Zhao, K.; Wang, Q.; Tian, Y. Incremental Cost-Sensitive Support Vector Machine With Linear-Exponential Loss. *IEEE Access* **2020**, *8*, 149899–149914. [CrossRef]
28. Wang, H.; Cui, Z.; Chen, Y.; Avidan, M.; Abdallah, A.B.; Kronzer, A. Predicting Hospital Readmission via Cost-Sensitive Deep Learning. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **2018**, *15*, 1968–1978. [CrossRef]
29. Esenogho, E.; Mienye, I.D.; Swart, T.G.; Aruleba, K.; Obaido, G. A Neural Network Ensemble with Feature Engineering for Improved Credit Card Fraud Detection. *IEEE Access* **2022**, *10*, 16400–16407. [CrossRef]
30. Schapire, R.E. A brief introduction to boosting. In Proceedings of the 16th Internation Joint Conference on Artificial Intelligence, IJCAI, Stockholm, Sweden, 31 July–6 August 1999; Volume 2, pp. 1401–1406.
31. Breiman, L. Bagging predictors. *Mach. Learn.* **1996**, *24*, 123–140. [CrossRef]
32. Ali, S.; Majid, A.; Javed, S.G.; Sattar, M. Can-CSC-GBE: Developing Cost-sensitive Classifier with Gentleboost Ensemble for breast cancer classification using protein amino acids and imbalanced data. *Comput. Biol. Med.* **2016**, *73*, 38–46. [CrossRef]
33. Feng, F.; Li, K.-C.; Shen, J.; Zhou, Q.; Yang, X. Using Cost-Sensitive Learning and Feature Selection Algorithms to Improve the Performance of Imbalanced Classification. *IEEE Access* **2020**, *8*, 69979–69996. [CrossRef]
34. Phankokkruad, M. Cost-Sensitive Extreme Gradient Boosting for Imbalanced Classification of Breast Cancer Diagnosis. In Proceedings of the 2020 10th IEEE International Conference on Control System, Computing and Engineering (ICCSCE), Penang, Malaysia, 21–22 August 2020; pp. 46–51.
35. UCI Machine Learning Repository: Chronic_Kidney_Disease Data Set. Available online: https://archive.ics.uci.edu/ml/datasets/Chronic_Kidney_Disease (accessed on 20 July 2021).
36. Nikravesh, F.Y.; Shirkhani, S.; Bayat, E.; Talebkhan, Y.; Mirabzadeh, E.; Sabzalinejad, M.; Aliabadi, H.A.M.; Nematollahi, L.; Ardakani, Y.H.; Sardari, S. Extension of human GCSF serum half-life by the fusion of albumin binding domain. *Sci. Rep.* **2022**, *12*, 667. [CrossRef]
37. Kumar, V.; Hebbar, S.; Kalam, R.; Panwar, S.; Prasad, S.; Srikanta, S.S.; Krishnaswamy, P.R.; Bhat, N. Creatinine-Iron Complex and Its Use in Electrochemical Measurement of Urine Creatinine. *IEEE Sens. J.* **2018**, *18*, 830–836. [CrossRef]
38. Muthumanjula, M.; Bhoopalan, R. Detection of White Blood Cell Cancer using Deep Learning using Cmyk-Moment Localisation for Information Retrieval. *J. IoT Soc. Mob. Anal. Cloud* **2022**, *4*, 54–72. [CrossRef]
39. Khan, S.I.; Hoque, A.S.M.L. SICE: An improved missing data imputation technique. *J. Big Data* **2020**, *7*, 37. [CrossRef]
40. Emmanuel, T.; Maupong, T.; Mpoeleng, D.; Semong, T.; Mphago, B.; Tabona, O. A survey on missing data in machine learning. *J. Big Data* **2021**, *8*, 140. [CrossRef]
41. Jamshidian, M.; Mata, M. Advances in Analysis of Mean and Covariance Structure when Data are Incomplete. In *Handbook of Latent Variable and Related Models*; Handbook of Computing and Statistics with Applications; Lee, S.-Y., Ed.; Elsevier: Amsterdam, The Netherlands, 2007; pp. 21–44.
42. Han, J.; Kamber, M.; Pei, J. 3-Data Preprocessing. In *Data Mining*, 3rd ed.; Han, J., Kamber, M., Pei, J., Eds.; The Morgan Kaufmann Series in Data Management Systems; Morgan Kaufmann: Boston, MA, USA, 2012; pp. 83–124. ISBN 978-0-12-381479-1.
43. Shakya, S. Modified Gray Wolf Feature Selection and Machine Learning Classification for Wireless Sensor Network Intrusion Detection. *IRO J. Sustain. Wirel. Syst.* **2021**, *3*, 118–127. [CrossRef]
44. Khaire, U.M.; Dhanalakshmi, R. Stability of feature selection algorithm: A review. *J. King Saud Univ. Comput. Inf. Sci.* **2019**. [CrossRef]
45. Gao, Z.; Xu, Y.; Meng, F.; Qi, F.; Lin, Z. Improved information gain-based feature selection for text categorization. In Proceedings of the 2014 4th International Conference on Wireless Communications, Vehicular Technology, Information Theory and Aerospace Electronic Systems (VITAE), IEEE, Aalborg, Denmark, 11–14 May 2014; pp. 1–5.
46. Alhaj, T.A.; Siraj, M.M.; Zainal, A.; Elshoush, H.T.; Elhaj, F. Feature Selection Using Information Gain for Improved Structural-Based Alert Correlation. *PLOS ONE* **2016**, *11*, e0166017. [CrossRef]
47. Shahraki, A.; Abbasi, M.; Haugen, Ø. Boosting algorithms for network intrusion detection: A comparative evaluation of Real AdaBoost, Gentle AdaBoost and Modest AdaBoost. *Eng. Appl. Artif. Intell.* **2020**, *94*, 103770. [CrossRef]
48. Freund, Y.; Schapire, R.E. A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. *J. Comput. Syst. Sci.* **1997**, *55*, 119–139. [CrossRef]
49. Zhao, D.; Wang, Y.; Wang, Q.; Wang, X. Comparative analysis of different characteristics of automatic sleep stages. *Comput. Methods Programs Biomed.* **2019**, *175*, 53–72. [CrossRef]

50. Wang, F.; Li, Z.; He, F.; Wang, R.; Yu, W.; Nie, F. Feature Learning Viewpoint of Adaboost and a New Algorithm. *IEEE Access* **2019**, *7*, 149890–149899. [CrossRef]
51. Wang, Y.; Feng, L. Improved Adaboost Algorithm for Classification Based on Noise Confidence Degree and Weighted Feature Selection. *IEEE Access* **2020**, *8*, 153011–153026. [CrossRef]
52. Sun, Y.; Kamel, M.S.; Wong, A.K.C.; Wang, Y. Cost-sensitive boosting for classification of imbalanced data. *Pattern Recognit.* **2007**, *40*, 3358–3378. [CrossRef]
53. Elkan, C. The foundations of cost-sensitive learning. In Proceedings of the International Joint Conference on Artificial Intelligence, Seattle, WA, USA, 4–10 August 2001; Lawrence Erlbaum Associates Ltd.: San Francisco, CA, USA, 2001; Volume 17, pp. 973–978.
54. Zhang, Y.; Jian, X. Unbalanced data classification based on oversampling and integrated learning. In Proceedings of the 2021 Asia-Pacific Conference on Communications Technology and Computer Science (ACCTCS), Shenyang, China, 22–24 January 2021; pp. 332–337.
55. Mienye, I.D.; Sun, Y. Effective Feature Selection for Improved Prediction of Heart Disease. In Proceedings of the Pan-African Artificial Intelligence and Smart Systems, Windhoek, Namibia, 6–8 September 2021; Ngatched, T.M.N., Woungang, I., Eds.; Springer International Publishing: Cham, Switzerland, 2022; pp. 94–107.
56. Aruleba, R.T.; Adekiya, T.A.; Ayawei, N.; Obaido, G.; Aruleba, K.; Mienye, I.D.; Aruleba, I.; Ogbuokiri, B. COVID-19 Diagnosis: A Review of Rapid Antigen, RT-PCR and Artificial Intelligence Methods. *Bioengineering* **2022**, *9*, 153. [CrossRef]
57. Mienye, I.D.; Sun, Y. Improved Heart Disease Prediction Using Particle Swarm Optimization Based Stacked Sparse Autoencoder. *Electronics* **2021**, *10*, 2347. [CrossRef]
58. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
59. Cramer, J.S. *The Origins of Logistic Regression*; Social Science Research Network: Rochester, NY, USA, 2002.
60. Krzywinski, M.; Altman, N. Classification and regression trees. *Nat. Methods* **2017**, *14*, 757–758. [CrossRef]
61. Chen, T.; Guestrin, C. XGBoost: A Scalable Tree Boosting System. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; Association for Computing Machinery: New York, NY, USA, 2016; pp. 785–794.
62. Breiman, L. Random Forests. *Mach. Learn.* **2001**, *45*, 5–32. [CrossRef]
63. Cortes, C.; Vapnik, V. Support-vector networks. *Mach. Learn.* **1995**, *20*, 273–297. [CrossRef]
64. Xie, J.; Wang, M.; Xu, S.; Huang, Z.; Grant, P.W. The Unsupervised Feature Selection Algorithms Based on Standard Deviation and Cosine Similarity for Genomic Data Analysis. *Front. Genet.* **2021**, *12*. [CrossRef]
65. Van Hulse, J.; Khoshgoftaar, T.M.; Napolitano, A.; Wald, R. Threshold-based feature selection techniques for high-dimensional bioinformatics data. *Netw. Model. Anal. Health Inform. Bioinform.* **2012**, *1*, 47–61. [CrossRef]
66. Rady, E.-H.A.; Anwar, A.S. Prediction of kidney disease stages using data mining algorithms. *Inform. Med. Unlocked* **2019**, *15*, 100178. [CrossRef]
67. Almustafa, K.M. Prediction of chronic kidney disease using different classification algorithms. *Inform. Med. Unlocked* **2021**, *24*, 100631. [CrossRef]
68. Mienye, I.D.; Sun, Y. Performance analysis of cost-sensitive learning methods with application to imbalanced medical data. *Inform. Med. Unlocked* **2021**, *25*, 100690. [CrossRef]
69. Akter, S.; Habib, A.; Islam, M.A.; Hossen, M.S.; Fahim, W.A.; Sarkar, P.R.; Ahmed, M. Comprehensive Performance Assessment of Deep Learning Models in Early Prediction and Risk Identification of Chronic Kidney Disease. *IEEE Access* **2021**, *9*, 165184–165206. [CrossRef]
70. Elkholy, S.M.M.; Rezk, A.; Saleh, A.A.E.F. Early Prediction of Chronic Kidney Disease Using Deep Belief Network. *IEEE Access* **2021**, *9*, 135542–135549. [CrossRef]
71. Rashed-Al-Mahfuz, M.; Haque, A.; Azad, A.; Alyami, S.A.; Quinn, J.M.W.; Moni, M.A. Clinically applicable machine learning approaches to identify attributes of Chronic Kidney Disease (CKD) for use in low-cost diagnostic screening. *IEEE J. Transl. Eng. Health Med.* **2021**, *9*, 4900511. [CrossRef]
72. Mienye, I.D.; Obaido, G.; Aruleba, K.; Dada, O.A. Enhanced Prediction of Chronic Kidney Disease Using Feature Selection and Boosted Classifiers. In Proceedings of the Intelligent Systems Design and Applications, ISDA, Online, 13–15 December 2021; Abraham, A., Gandhi, N., Hanne, T., Hong, T.P., Nogueira Rios, T., Ding, W., Eds.; Lecture Notes in Networks and Systems. Springer: Cham, Switzerland, 2022; Volume 418, pp. 527–537.
73. Kikuchi, H.; Kanda, E.; Mandai, S.; Akazawa, M.; Iimori, S.; Oi, K.; Naito, S.; Noda, Y.; Toda, T.; Tamura, T.; et al. Combination of low body mass index and serum albumin level is associated with chronic kidney disease progression: The chronic kidney disease-research of outcomes in treatment and epidemiology (CKD-ROUTE) study. *Clin. Exp. Nephrol.* **2017**, *21*, 55–62. [CrossRef]
74. Sun, J.; Su, H.; Lou, Y.; Wang, M. Association Between Serum Albumin Level and All-Cause Mortality in Patients With Chronic Kidney Disease: A Retrospective Cohort Study. *Am. J. Med. Sci.* **2021**, *361*, 451–460. [CrossRef]