Enhanced Stacked Ensemble Model for Accurate Diagnosis of Parkinson's Disease

*Nobhonil Roy Choudhury[1], Avijit Kumar Chaudhuri[2], Shivnath Ghosh[3], Sulekha Das[4]

Abstract: Parkinson's disease (PD) has an incidence of 15 to 43 per one lac population, an estimate showing that India has more than one lac PD patient and is expected to have the largest number of PD patients in the world. About 40-45% of the patients have had their initial motor manifestation at the age of 22-49, which is known as Early Onset Parkinson's Disease (EOPD) (Early et al. (EOPD) in India Vs. Western Populations, n.d.)1. The research aims to harness the power of artificial intelligence and machine learning to develop a predictive model for diagnosing Parkinson's disease (PD). This initiative aligns with the growing potential of Artificial Intelligence (AI) in healthcare research, particularly in addressing classification challenges like PD diagnosis. By leveraging advanced algorithms and data analysis techniques, this study enhances early prediction of PD, facilitating timely intervention and improving patient outcomes. The zenith of the study is marked by the K-Nearest-Neighbors (KNN) algorithm with the highest accuracy score of 97.44%, the greatest power to judge procedure, and the Kappa statistic of 90.78%, which explains the highest level of diagnostic concordance. The Stacking of Random Forest, KNN, and AdaBoost produces 100% specificity and f1 score. Also, these two algorithms achieved an ROC AUC score of 100%, thus clinching ground in the contest of the precision of a discriminating model. Contrarily, the performance of the Naïve Bayes classifier is lower in all performance metrics. The facts retrieved in this study lead to the bewildering benefit of ensemble and KNN algorithms in forecasting Parkinson's disease in advance. It may lead to a revolutionary turnaround in patient care and therapeutic approaches.[1]Early onset parkinsonism (EOPD) in India vs. Western populations.

Keywords: Parkinson's disease, Machine Learning, Classification, Stacked Model, Diagnosis, Ensemble Classifier.

1. Introduction

In Parkinson's disease (PD), a progressive neurodegenerative disorder, clinicians are often faced with the complicated task of differentiating the differences from the rest of the symptoms and the intensity of early signs.

The biological neural network, abbreviated as BNN, is responsible for dopamine production. This neurotransmitter plays several essential roles in the motor and non-motor processes of the human body. In PD, some groups of neurons stop producing a chemical called dopamine. In the advanced stages of the disease, there is a decline in the amount of dopamine produced in the neurological system, causing the affected individual to develop mobility problems. Early signs include the development of intentional tremors in an upper extremity, such as the hand, foot, or leg. Other symptoms include slowness of movement, rigidity, postural instability, lack of facial expression, clumsy movements, problems with speech and swallowing, dementia, graphic dyspraxia, anosmia, urinary incontinence, phonation disorder, and vocal cord dystonia [1]. About 90% of PD patients develop some form of vocal disorder that affects their voice and communication [2]. Analysis of various voice characteristics of PD patients through modern signal processing algorithms in the current world helps diagnose and monitor the disease.

PD diagnosis is not easy, especially if the physician cannot identify any risk factors, and reaching a consensus with other physicians on diagnosing patients is almost impossible [3].

A traditional diagnostic model is essentially based on clinical observation and a long interview conducted to collect patients' histories as a rule. That is why the technique is subjective and may differ a lot among individual practitioners. In the face of the worldwide incidence of PD, whose curve continues to rise, there is an urgent demand for unbiased, objective diagnostic approaches.

The fact that this issue can be solved with the advent of computational algorithms has helped the development of ML models that analyze and understand complex biomedical data. ML techniques, especially supervised learning models that implement pattern recognition and differentiate patterns of PD and non-PD, are one of the core research fields in machine learning for developing predictive models with an accuracy of 90%.

This article describes a collection of ML algorithms including K-Nearest Neighbors (KNN), Naïve Bayes (NB), Random Forest (RF), AdaBoost (AB), Voting Classifier (VC), Decision Tree (DT), and Stacking Classifier (SC), to investigate their potential in the detection of PD in patients. The success rate of ML models often depends on several issues, such as data quality, identical feature selection, the complexity of algorithms, and the fine-tuning of hyperparameters [4]. To enhance the model performance, a thorough feature engineering and model assessment process was adopted, ensuring that the most probable variables represent the model and that the model is calibrated to score the maximum accuracy.

Accuracy is essential for such analyses, in addition to accuracy, sensitivity, and specificity measures. However, despite several solutions aimed at disease prediction using machine learning, few offer the above performance characteristics.

This paper compares an SC with other classifiers such as KNN, NB, RF, AB, VC, DT, and the previous work using the same dataset as in Table 1. The authors compare performance measures and statistical tests such as accuracy, sensitivity, specificity, receiver operating characteristic (ROC), Area Under Curve (AUC), Kappa statistic, etc., with different data split ratios like 50–50%, 66–34%, 80–20%, and 10-fold cross-validation.

This article aims to answer the following few research questions:

Research Question 1: Is the presented Stacking classifier appropriate to predict PD?

Research Question 2: Is the suggested model satisfactory with both additional sensitivity and specificity criteria?

Research Question 3: Is the use of the suggested architecture statistically significant at different levels of training and testing sets in the dataset?

2. Relevant Literature

Evolution in technology has advanced the process of designing modern diagnosis and detection techniques for diseases. Artificial neural networks, fuzzy solutions, and other expert solutions are more frequently applied in most medical fields. Notably, the diagnosis of Parkinson's disease (PD) has noted several techniques that employed voice and speech datasets, with relevant denoting stages in terms of preprocessing, feature extraction, and classification phases of computerized systems.

Little et al. [5] could prove the effectiveness reaching 91 percent. They reported the results of 4% accuracy in diagnosing PD by using the kernel support vector machine (SVM) with the feature selection technique. They also presented nonlinear models based on Dirichlet Process Mixtures for the PD diagnosis, similar to what was presented by Shahbaba and Neal [6], with a resulting accuracy of 89. On average, the system's accuracy was 47%, determined using multiclass multi-kernel Relevance Vector Machines (mRVMs) on systems validated by the ten-fold cross-validation. Similarly, Psorakis et al. [7] proposed genetic programming and expectation maximization for a comparable purpose. Also, Guo et al. [8], Psorakis et al. [7], and Sakar and Kursun [9] proposed models constructed using mutual information measures as well as SVMs. Thus, the study of Sakar and Kursun [9] and Das [10] revealed that the accuracy efficacy of the Artificial Neural Network-based models hit 92—9%.

Fuzzy entropy metrics with a similarity classifier were used, and the average accuracy, reported by Das [10] and Tuukka [11], was 85 per cent. 03% with the training testing split of 50:50. As for the feature selection, Luukka [11] used correlation-based rotation forest ensemble classifiers, and Ozcift and Gulten [12] used correlation-based Recursive Feature Elimination (RFE). To increase the efficiency in small datasets, Ozcift and Gulten [12] and Li et al. [13] applied fuzzy nonlinear transformation methods involving PCA and SVM. The authors used these techniques on six medical datasets, one of them being the PD dataset mentioned in the present paper.

Astrom and Koker [14] arrived at the figure of 91. They observed a classification accuracy of twenty percent with Parallel Artificial Neural Network architecture. Spadoto et al. [15] used feature selection based on an evolutionary algorithm to increase the performance of an optimum path forest classifier for the PD diagnosis. Polat [16] used an FCM clustering feature weighting algorithm and kNN classifier with 97—93% accuracy. Daliri [17] achieved 91. An accuracy of approximately 20 percent can be achieved when modeling with a chosen covariate of a chi-square kernel SVM. The next study by Chen et al. [18] achieved 96. Outperforms other methods that have 7% accuracy with 10-fold cross-validation using PCA and fuzzy kNN. Zuo et al. [19] used Particle Swarm Optimization to enhance a fuzzy k-nearest neighbor classifier reaching 97. 47% accuracy. Zhang [20] applied time-frequency characteristics, stacked autoencoders, and kNN classifiers to diagnose PD.

Over these later years, more assessment has been done regarding the methodologies and diagnostic resolutions. Sayed et al. [21] have also used CNNs for voice record data analysis, yielding a diagnostic accuracy of about 94%. 5 percent with a sensitivity of 93—2%. Rovini, Maremmani, and Cavallo [22] used Random Forest classifiers on multisensory data acquired by wearable devices with an accuracy of 92%. As indicated in the given review, the preliminary study acknowledges a lower mean classification accuracy of 7% for differentiating PD patients from controls. Regarding feature selection of the gait analysis feature set, Ali, Salim, and Saeed [23] proposed using a genetic algorithm; this practically boosted classification accuracy to 95%. 6%. You et al. [24] used SVMs to analyze MRI scans with an accuracy 93. 8% accuracy. Huang and other researchers [25] used deep learning to diagnose signs of micrographia from handwriting, achieving 91%—3% accuracy. The authors Cao, Xia, Li, Zhang, and Chen used Neural Networks to analyze the dysfunctions of olfactory tests related to PD, and the technique helped in determining PD-related dysfunctions with 89%—9% accuracy. Rahman Sajal et al. [27] proposed a telemedicine system based on the ML for PD distant evaluation, with a kappa coefficient of 0 857. The table focuses on the researchers, the year of their studies, the methodologies used, and the resulting accuracy percentages concerning Parkinson's Disease.

Table 1. Comparison of accuracies in previous studies

Study	**Methodology**	**Accuracy (%)**
Little et al. [5]	Kernel SVM with feature selection	91.4
Shahbaba& Neal [6]	Multi-class multi-kernel Relevance Vector Machines (mRVMs)	89.47
Sakar &Kursun[9]	Artificial Neural Network	92.9
Das [10]	Fuzzy entropy metrics with a similarity classifier	85.03
Astrom& Koker [14]	Parallel Artificial Neural Network architecture	91.2
Polat[16]	FCM clustering-based feature weighting and kNN classifier	97.93

Daliri[17]	Chi-square kernel SVM	91.2
Chen et al. [18]	PCA and fuzzy kNN	96.07
Zuo et al. [19]	Particle Swarm Optimization and fuzzy k-nearest neighbor	97.47
Sayed et al. [21]	Deep Learning (CNN)	94.5
Rovini et al. [22]	Random Forest (Wearable Sensors)	92.7
Ali et al. [23]	Genetic Algorithms (Gait Analysis)	95.6
Ya et al. [24]	SVM (MRI Analysis)	93.8
Huang et al. [25]	Deep Learning (Handwriting Analysis)	91.3
Cao et al. [26]	Neural Networks (Olfactory Dysfunction)	89.9

3. Methodology

Dataset

The dataset[2] was elaborated by Max Little from the University of Oxford and the National Centre for Voice and Speech, located in Denver, Colorado, to represent distinctive features of speech signals. The initial article describing the datasheet describes methods for extracting features for detecting general voice disorders. This data set consists of biomedical voice measures from 31 people comprising 23 diagnosed with Parkinson's Disease (PD). Within the dataset, the column represents distinct measurements for voices, and the row represents 195 voices recorded by those individuals. The recordings in Table 2 of the original paper are marked by the "name" column. The main aim of this dataset is to dissociate healthy individuals from the ones with a PD, using the "status" column where a 0 value denotes a healthy person and 1 represents the patient with PD as described by Little et al.[5] in 2007. Data is stored in ASCII CSV format; each row of the CSV file signifies a single instance of voice recorded. The average recording number is around 6 for each patient, and the patient's name is listed in the first column of the table as an identifier [2].

Table 2. Description of the dataset

Sl. No	**Attributes**	**Description**	**Range of Values**	**Mean**	**Std. Dev**
1	name	Patient name in ASCII and recording number	-	-	-
2	MDVP:Fo(Hz)	Vocal fundamental frequency on average	88.333 - 260.105	154.228641	41.39006475
3	MDVP:Fhi(Hz)	Maximum fundamental frequency of the voice	102.145 - 592.03	197.1049179	91.49154764

4	MDVP:Flo(Hz)	Minimum fundamental frequency of the voice	65.476 - 239.17	116.3246308	43.52141318
5	MDVP:Jitter(%)	Several measures of fundamental frequency variation Jitter (Jitt) =$\frac{1}{N-1}\sum_{i=1}^{N-1} \quad \lvert f_{i+1} - f_i \rvert$ Where (f_i) is the fundamental frequency and amplitude of the (i^{th}) vocal fold vibration cycle, respectively, and (N) is the total number of cycles measured.	0.00168 - 0.03316	0.00622	0.004848
6	MDVP:Jitter(Abs)		0.000007 - 0.00026	4.40E-05	3.48E-05
7	MDVP:RAP		0.00068 - 0.02144	0.00330641	0.002967774
8	MDVP:PPQ		0.00092 - 0.01958	0.003446359	0.002758977
9	Jitter:DDP		0.00204 - 0.06433	0.009919949	0.008903344
10	MDVP:Shimmer	Several amplitude variation measures Shimmer (Shim) =$\frac{1}{N-1}\sum_{i=1}^{N-1} \quad \lvert \frac{A_{i+1}-A_i}{A_i} \rvert$ Where (A_i) is the fundamental frequency and amplitude of the (i^{th}) vocal fold vibration cycle, respectively, and (N) is the total number of cycles measured.	0.00954 - 0.11908	0.029709128	0.018856932
11	MDVP:Shimmer(dB)		0.085 - 1.302	0.282251282	0.19487729
12	Shimmer:APQ3		0.00455 - 0.05647	0.015664154	0.010153162
13	Shimmer:APQ5		0.0057 - 0.0794	0.017878256	0.012023706
14	MDVP:APQ		0.00719 - 0.13778	0.024081487	0.016946736
15	Shimmer:DDA		0.01364 - 0.16942	0.046992615	0.030459119
16	NHR	Two measurements of the noise-to-tonal component ratio in the voice	0.00065 - 0.31482	0.024847077	0.040418449
17	HNR		8.441 - 33.047	21.88597436	4.425764269
18	RPDE	There are two measurements of nonlinear dynamical complexity.	0.25657 - 0.685151	0.498535538	0.103941714
19	D2		1.423287 - 3.671155	2.381826087	0.382799047
20	DFA	Exponent of signal fractal scaling	0.574282 - 0.825288	0.718099046	0.05533583
21	spread1	Three nonlinear measurements of fundamental frequency fluctuation	(-7.964984) – (-2.434031)	-5.684396744	1.090207764

22	spread2		0.006274 - 0.450493	0.226510349	0.083405763
23	PPE		0.044539 - 0.527367	0.206551641	0.090119322
24	status	Patients' health conditions 1 – PD 0 - healthy			

[2]www.kaggle.com

Stacking Classifier

A stacking classifier is an ensemble learning technique that combines various base classifiers; the purpose is to improve the accuracy of the prediction. In this case, the algorithm combines three diverse base classifiers: Random Forest, k Nearest Neighbors (kNN), and AdaBoost, with a Decision Tree as the meta-classifier.

```
Input : Training data $T = \{m_k, n_k\}_{k=1}^{y}, m_k \in R_z, m_k \in C$, where C denotes the classes.
Output : A stacking meta-classifier ensemble $P_f$
Step 1 : Adopt cross validation approach in preparing a training set for second-level
Random Forest classifier
Split the starting dataset into $k_f (= 3)$ equal-size subsets $\{S1, S2, S3\}$ in a random way
for $l \rightarrow 1$ to $k_f$ do
    Learn first-level classifiers
    for s $\rightarrow$ 1 to 3 do
        Train a learner $j_{kt}$ from $T \backslash T_{k_f}$
    end for
    Construct a training set for second-level classifier
    for $m_k \in T_{k_f}$ do
        Get a record $\{m_k^{'}, n_k\}$, where $m_k^{'} = \{pk_{f1}(m_k), pk_{f2}(m_k), pk_{f3}(m_k)\}$
    end for
end for
Step 2 Learn a second-level classifier
Learn a new classifier p from the collection of $\{m_k^{'}, n_k\}$
Step 3 Re-learn first-level learners
for s $\rightarrow$ 1 to 3 do
    Learn a classifier $p_s$ based on T
end for
Return $P_f(m)$
```

Fig. 1 Algorithm: Stacking Classifier

Here's a step-by-step description of how the algorithm works:

Here is a step-by-step description of how the algorithm works:

Base Classifiers Training: The training of every base classifier (Random Forest, kNN, AdaBoost) is carried out separately on the training dataset.

Random Forest: It develops a variety of decision trees during training and then mixes up their outcomes to raise the accuracy and avoid overfitting.

k Nearest Neighbors (kNN): It performs the nearest neighbors in the feature space based on the majority class of their k nearest neighbors.

AdaBoost (Adaptive Boosting): It is an iterative ensemble approach that utilizes each weak learner (mostly decision trees) to train the data set, and the subsequent ones deal with the misclassified samples from the previous learners.

Prediction Generation: Every trained base classifier calculates the predicted class labels for the validation set (or the testing set, in case no validation set is used).

Meta-Classifier Training: The base classifiers' outputs become the meta-classifier's features (Decision Tree). After that, the meta-classifier is trained with these predicted class labels to learn how to combine them efficiently.

Final Prediction: When presented with a new case for prediction, each base classifier makes its prediction, and these judgments are then passed on to the meta-classifier. The meta-classifier combines all these predictions to make the final prediction.

Stacking can get higher accuracy than any component classifiers by using diverse base classifiers, allowing the meta-classifiers to learn from their predictions. It is a Decision Tree meta-classifier used because of its simple and interpretable nature and capability to deal with the nonlinear relationship between base classifiers' predictions.

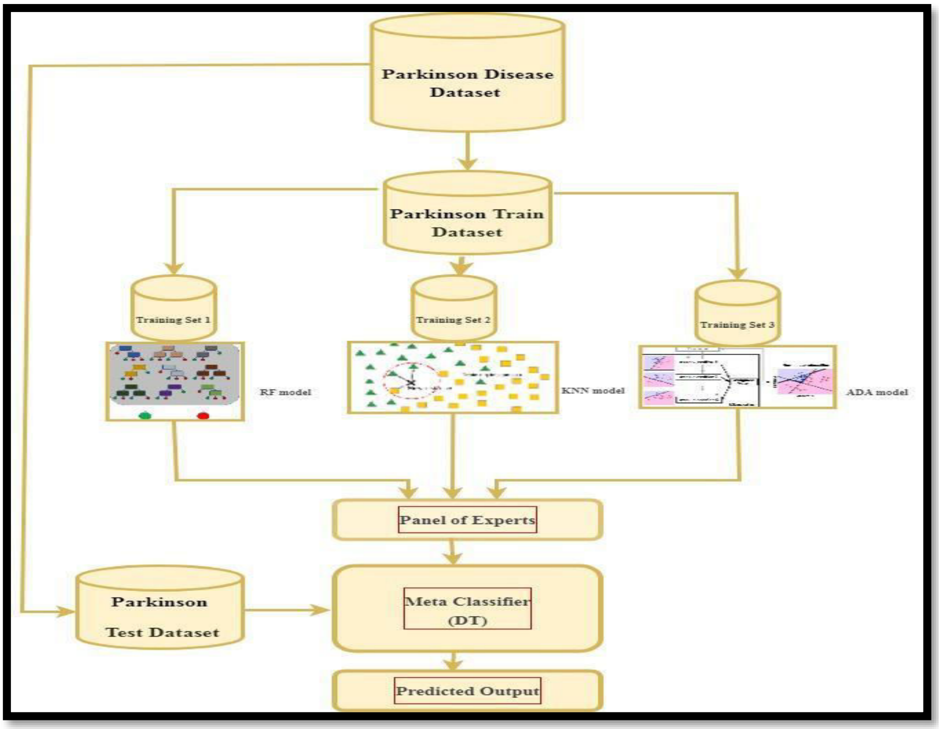


Fig. 2.Working of the Proposed Stacked Model

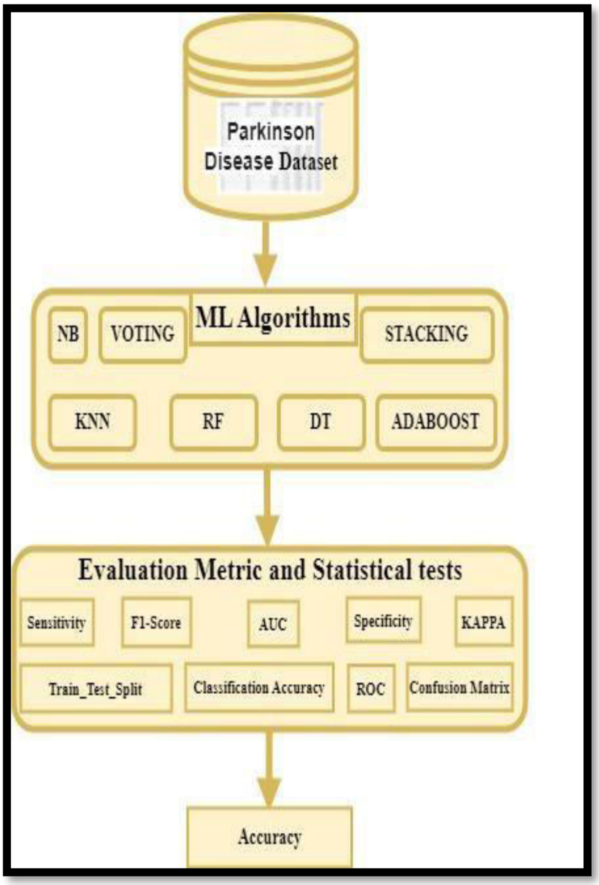


Fig. 3. Parkinson's Diseases Prediction Framework

The Decision Tree (DT)

DT is one of the supervised machine learning algorithms used frequently because of its flexibility, as it can be used for classification and regression [28]. The approach utilizes a tree-like structure, which embodies an algorithmic flow of conditional control statements. Within this framework, possible results include resource codes, stochastic events, and objectives [29]. The main idea in deploying Decision Tree is to use the training data to build the model where the selection criteria are specified to predict class or approximate target values.

Naïve Bayes

NB is a very basic and commonly used ML classifier that examines the dataset's features without accounting for any dependency among the features. This model is characterized by generalizing the output class of a particular example by a high probability class. NB works as a probabilistic classifier, whereby the algorithm retrieves earlier training data for predicting limited independence. Bayes' theorem, on the other hand, depicts the possibility of a specific event, given that another event has already occurred [4]. Naive Bayes classifiers can be used successfully for different problems, from traditional spam detection to medical screening. In equation (i), the mathematical interpretation of the Bayesian theorem is summarized derivatively.

$$P\left(\frac{X}{Y}\right)=\frac{P\left(\frac{X}{Y}\right)*P(X)}{P(Y)} \qquad \text{(i)}$$

where X and Y are events and P(Y) ≠ 0

Random Forest

The Random Forest (RF) is a tree-based classifier, a powerful tool for handling both classification and regression analysis. The RF framework employs many trees, and for the classification task, the output is the mean of tree predictions [2]. RF uses the ensemble learning methodology for regression, classification, and many other tasks, generating multiple decision trees (DTs) during training and providing the class using the commonly occurring classification method or the average prediction for regression from different trees. This measure takes DTs away from overfitting their training datasets [30]. The central, indispensable feature between the aggregated DTs and their endings lies in the forest of trees. A third layer of unpredictability is employed within random forests, making the system more reliable and correct, yielding more accurate predictions.

K-nearest neighbour

The classic KNN (K-nearest neighbor) algorithm is a supervised machine-learning approach for classification tasks. It uses the parameter 'k' as the key, where k represents the number of nearest neighbors considered while making a prediction. KNN relies on detecting the nearest data points or neighbors from the query's training dataset. These neighbors of the nearest points are the nearest neighbors of the query point. Subsequently, KNN employs a majority voting scheme to determine the most common class label from the k nearest data points. Such a dominant class is finally allocated as the output by the system [30].

Ada Boost

It allows aggregating a set of "weak classifiers" into a unified "strong classifier." Usually, it is favored to implement decision trees with the smallest complexity among the classifiers, which we call decision stumps, and they contain only a single level or split. This methodology develops a model that starts by placing the same weights on all the data points, and then it reweights the misclassified points for every model iteration. While the method takes place, the weights of the points get higher and are given more importance each time, which, as a result, helps build the successive models until the error rate is minimal [31, 32].

Voting classifier

This meta-classifier combines similar or entirely different machine learning algorithms through a voting system to sharpen the predictions. The classifier employs two voting strategies: hard and soft voting. In hard voting, the final prediction is most of the base models. On the other hand, soft voting requires base models to use the predict_proba method. Here, the forecasted values are assigned according to probability, providing more details when deciding. The voting classifier usually performs better than single models using different model predictions. Here, RF, KNN and Adaboost classifiers make up the ensemble in this implementation. The training data and points are shuffled before feeding to the RF, KNN, and Adaboost models. They are used as standalone models which make their predictions. These predictions are then collected through a voting process, using soft voting for probability decision-making. Finally, most votes are arrived at, providing the final prediction.

4. Results and Discussion

A comprehensive breakdown of various performance metrics commonly used in binary classification problems is described below.

Accuracy: Accuracy characterizes the overall correctness of the classifier through the ratio of all correctly classified instances (both positive and negative) to the total number of instances, as shown in equation

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{FP} + \text{FN} + \text{TP} + \text{TN}} \qquad \text{(ii)}$$

Sensitivity(Recall): Sensitivity, referred to as the predictive value or true positive rate, denotes the probability of correctly pointing out most of the disease classes and it is depicted as shown in the formula (iii).

$$\text{Sensitivity} = \frac{\text{TP}}{\text{FN} + \text{TP}} \qquad \text{(iii)}$$

Specificity: Specificity means the proportion of correctly classified true negative cases to the whole true negative cases. It is depicted as shown in the formula (iv).

$$\text{Specificity} = \frac{\text{TN}}{\text{FP} + \text{TN}} \qquad \text{(iv)}$$

.

Precision: The measures of precision refer to the number of times positive test results were truly diagnosed among all positive ones. It is depicted as shown in the formula (v).

$$\text{Precision} = \frac{\text{TP}}{\text{FP} + \text{TP}} \qquad \text{(v)}$$

False Positive Rate (FPR): The FPR is a ratio of the proportion of true negative to that of the corresponding false positive and is represented by the mathematical function (vi).

$$\text{FPR} = \frac{\text{FP}}{\text{TN} + \text{FP}} \tag{vi}$$

F1 Score: The F1 score is the harmonic mean of precision and recall rates. It takes a balance between precision and recall, used when classes are unbalanced. It is represented by the mathematical function (vii).

$$\text{F1Score} = \frac{2 \times \text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}} \tag{vii}$$

AUC/ ROC: ROC (Receiver Operating Characteristic) AUC (Area Under the Curve) is a performance measurement technique that is used to evaluate the quality of binary classification models. ROC curves show the true positive rate vs the false positive rate for different classification thresholds. The AUC ROC reduces the ROC curve into a single value portraying the probability that the classifier will rank a randomly chosen positive instance higher than a randomly chosen negative instance.

The AUC ROC score ranges from 0 to 1, where:

A score of 1 represents the ideal classifier that can completely segregate positive and negative instances.

A score of 0.5 indicates that the classifier is no more accurate than random guessing (or no better than chance).

In reality, the bigger the AUC ROC measure is for one classifier the higher its accuracy in recognizing between positive and negative classes. It is one of the most popular measures to assess how well binary classification models perform, especially when the distribution of classes is uneven.

$$\text{AUC} = \frac{1}{2}(1 + \text{Sensitivity} - \text{FPR}) \tag{viii}$$

Kappa Statistic: The Kappa score (also known as Cohen's Kappa coefficient) is a statistical tool that evaluates the level of consistency when categorical items are rated or graded by different raters. It reflects on how much agreement between two raters (or observers) is greater than that possible by mere chance. It is also very effective when analyzing categorical data where a random coincidence could be easy. The Kappa score ranges from -1 to 1, where:

1 indicates perfect agreement.

0 indicates agreement equal to that we can expect by chance.

-1 indicates complete disagreement.

In practice, a kappa of 0.8 or more is usually considered excellent correspondence and a kappa between 0.6 and 0.8 means substantial agreement, 0.4 to 0.6 moderate agreement and less than 0.4 poor agreement.

$$\text{Kappa Statistic}=\frac{2 \times (\text{TP} \times \text{TN - FN} \times \text{FP})}{(\text{TP} \times \text{FN} + \text{TP} \times \text{FP} + 2 \times \text{TP} \times \text{TN} + \text{FN}^2 + \text{FN} \times \text{TN} + \text{FP}^2 + \text{FP} \times \text{TN})} \tag{ix}$$

The kind of research the author engaged in required creating and implementing a new model using the Python language. In the pursuit of robust classification of healthy individuals and those with Parkinson's Disease (PD), the author conducted an extensive comparative analysis involving five prominent machine learning algorithms: KNN, NB, RF, ADB, VC, and SC.

Therefore, the primary purpose of this form of experimentation is rooted in features that help the model achieve high levels of accuracy in correctly sorting out people with Parkinson's from healthy individuals to aid in the timely identification of the form of Parkinson's disease. The accuracy, as reported, was 97. 44% means a great improvement in this aspect, where the next steps include the wise choice and the application of the machine learning algorithms. As far as this analysis is concerned, it was realized that different algorithms were characterized by differential levels of efficiency, with some algorithms being more accurate than others. Notably, KNN gave an excellent accuracy of 97%. This is 44 %, signifying its ability to determine the patterns within the biomedical voice measurements dataset from the UCI repository. However, out of all the analyzed algorithms that tested on the dataset, SC took a more inferior position compared to other methods, having, in turn, an accuracy of 95%. Such efficiency differences are quite normal; therefore, the author used complex categorizations of the ensemble machine learning algorithms. These advanced methodologies sought to improve the efficiency of the poor-performing classifiers to increase the model's performance.

Table 3. Training and Testing Set Partition

Training Testing Partition	Total Training Record	Positive record in Training Set	Negative Record in Training Set
80-20	156	115 (73.72%)	41 (26.28%)
50-50	97	72 (74.23%)	25 (25.77%)
66-34	122	97 (75.78%)	31 (24.22%)
10-Fold Cross Validation	197	147(74.62)	48(24.38)

Table 4. Comparisons of Accuracies

Methods	Results(PERCENTAGE)
KNN	97.44
Naïve Byaes	71.79
Random Forest	89.74
AdaBoost	89.74
Voting Classifier (KNN,RF,ADA , voting=hard)	94.87
Decision Tree	84.72

Stacking Classfier(RF,KNN,ADA)	94.87

Confusion matrix is a basic measure often used in the assessment of classification models, where real and predicted classification, provided in the outcome of the analysis carried out by the classifiers, is shown in detail. They enable one to have a standard format for the performance indicators that are vital in evaluating the efficiency of these systems. The study displays the findings based on the confusion matrices that illustrate the results of various machine learning algorithms. These matrices involve the true positives, true negatives, false negatives, and false positives that are obtained after the classification process has been carried out. These metrics are instrumental in assessing the efficiency and accuracy of the classification models in question. Quantitative provisions like sensitivity, specificity, and accuracy are central in assessing the suggested model and other approaches that may be considered. Sensitivity or True Positive Rate refers to the ability of the model to determine as having positive cases by correctly identifying it as such.

Table 5. Comparisons of Specificities

Methods	Results
KNN	85.71
Naïve Byaes	71.43
Random Forest	57.14
AdaBoost	85.71
Voting Classifier (KNN,RF,ADA , voting=hard)	85.71
Decision Tree	85.71
StackingClassfier(RF,KNN,ADA)	100

Table 6. Comparisons of F1 Scores

Methods	Results
KNN	98.46
Naïve Byaes	80.70
Random Forest	93.94
AdaBoost	93.55
Voting Classifier (KNN, RF, ADA, voting=hard)	96.88
Decision Tree	90
StackingClassfier(RF, KNN, ADA)	100

The score shown in Table 6 is expressed as a high f1-score for both classes; the importance of the model is reflected in the high specificity of the prediction. Compared to the other typical employed classifiers, it is issued that the SC model yields the best f1-score and specificity values regarding the PD dataset f1-score=1, specificity=1.

In different analyses in the present research study, the ADB classifier had a classification accuracy of 89% out of 100%—74%, accompanied by an f1-score of 93. If the test is implemented as a screen for a common disease, the sensitivity of the test is 55%, and the specificity of the test is 85%. 71%. Likewise, the classification accuracy in the RF model was 89. 74%, and the f1-score is 93. 94%, and specificity was 57%. 14%. Thus, regarding accuracy, KNN showed promising results as it is up to 97%. 44 % with a splendid f1-score of 98. 46%, and the specificity of 85%. 71%. VC resulted in an accuracy of eighty-two per cent. Of all the entities extracted, 59% were correctly identified, and the F1 score reached 96. 88% and a specificity of 85. 71%. About the NB classifier, an accuracy of 71 was observed. The precision = 79%, and the f1-score = 80. 7% and a specificity of 71. 43%.

However, the six classifiers' comparison revealed that the SC model had the best performance, with an accuracy of 95%, an f1-score of 100%, and a specificity of 100%. The entire spectrum of findings is summarized in the two tables: Table 4 and Table 5. Also, the performance results analyzed from the SC classifier are considered fairly good, demonstrating the stronger f1-score and f1-score in identifying PD. This stresses and emphasizes the great extent of impact that results from the use of the proposed classifier in the prediction of PD. In this case, the proposed SC classifier implies requiring fewer PD tests vice the current system due to the increase in specificity. Moreover, the higher the f1-score, the less the cost and waiting time of attendants required to save the lives of critical patients.

The study on the analysis results of the 7 celebrity machine learning schemes for voting and from the proposal of the stacking architecture shown in Table 7, including the results of the Cohen's Kappa Statistic (CKS) value. By comparing with other classifiers based on the results shown in Table 7, it proves that the suggested stacking model is much better than others regarding the CKS score as it gets 1. This means that the stacking model has 100 percent accuracy, which shows that the model is closely in good agreement with the actual label.

Surprisingly, KNN (K Nearest Neighbors) alone was one among the individual classifiers that yielded good performance with the corresponding CKS value of 0. 91. Since it is obvious that the numbers are high, this implies that there is a high degree of recommendations by KNN and true label to be in concordance. Nevertheless, concerning its performance, even though KNN secured a fairly decent result, its scores decreased significantly at a rate even beyond the proposed stacking model that ranked as the model to demonstrate the highest accuracy with the CKS score.

Hence, the proposed stacking model yields better performance over other classifiers regarding CKS value, which confirms the proposed model's effectiveness in learning inherent data patterns and complex predictions. Stacking, an example of an ensemble method, can be defined as collecting, processing, and merging several learning models to increase their performance and robustness.

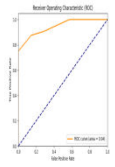
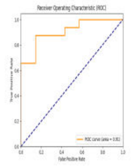
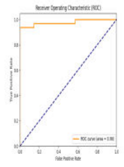
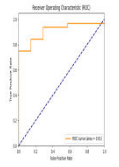
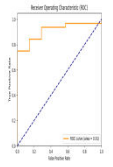
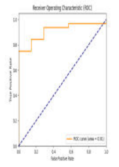
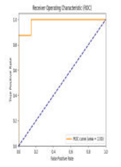
The ROCs summarized in Table 6 are the performance results of the machine learning techniques applied in this research run using the 10-fold validation technique. ROC charts are probably the most used graphical method to evaluate binary classifiers. This curve illustrates the trade-off between the data's true positive rate and the false positive rate in the two classes. Therefore, the experiment demonstrates that the proposed classifier, which the authors of this research developed, provided the highest performance among most of the models researchers wrote in their papers' literature review section. According to the findings, it was observed that constructions of the proposed classifier depicted excellent performance, while the AUC/ROC values were assessed through 10-fold cross-validation. The AUC score that the proposed model attained was 1, which reveals the model's perfect performance in discriminating between the two classes of data.

Similarly, other approaches to machine learning have also depicted reasonable performance. For example, the KNN (k-Nearest Neighbors) algorithm obtained the highest AUC value of 1, interpreted as perfect discrimination between the positive and negative classes. The voting classifier and random forest were also amongst the best in their class with respective ROCs of 0.99 and 0.95. Therefore, the KNN classifier, voting classifier, random forest, and the presented stacking model seem to be useful for making classification between the classes. Higher ROC values depicted the good or excellent discrimination offered by these methods, which can be applied to a binary classification context in real data. A comparison of ROC/ AUC is made in the following table 8.

Table 7.Comparisons ofKAPPA Scores

Methods	Results
KNN	90.78
Naïve Bayes	31.14
Random Forest	60.80
AdaBoost	68.67
Voting Classifier (KNN,RF,ADA, voting=hard)	82.59
Decision Tree	57.30
StackingClassfier(rf,knn,ada)	100

Table 8. Comparison of ROCs / AUCs

Training - Testing Partition	**KNN**	**NB**	**RF**	**ADB**	**VC**	**DT**	**SC**
10-fold cross validation							
AUC	1	0.77	0.95	0.92	0.99	0.85	1

Wilcoxon rank sum test

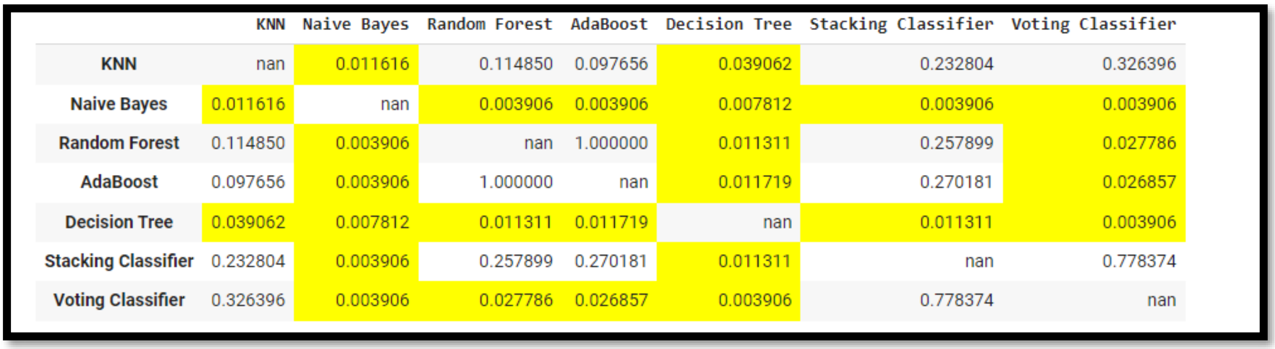
Like any other testing technique, a hypothesis test is utilized to establish the validity of the statement regarding the parameter in the population. In light of the current study, depending on the significance level, in the parametric method, one can apply either a paired t-test or a z-test, while a two-sample-test t will be apt for unpaired values. However, for nonparametric tests, Statistical tests that do not assume the normality of data include the Wilcoxon signed ranked test/Wilcoxon rank sum test can also be used. Wilcoxon rank sum test allows the researcher to test two independent populations for a mean difference since it is a nonparametric test [33]. Regarding the significance level (α) equal to 0 in the case of Wilcoxon signed-rank test. 05, the null hypothesis can be rejected, after which we compare the interpolated values of y with the observed values for the percentage estimate. 05: This means there is no significant difference in the paired observations, and the two differences are less than 5%. Therefore, the method does not acknowledge the null hypothesis and concludes that there is a difference in the two related groups. If the p-value is more than 0.. 05: Hence, it concluded that there is no real change in the paired observation as it is more than 0. 05. Consequently, the method misses the opportunity to 497 reject the null hypothesis, and this indicates that there is no significant difference in the paired observations. In summary:

P-value < 0. 05: Based on the above calculations, the t-value is greater than the t-value critical value; hence, the research rejects the null hypothesis.

P-value ≥ 0. 05: Hence, the null hypothesis cannot be rejected. There is no substantial difference.

This interpretation is done with the help of an accepted significance level (α) of 0. This can be represented as 05, a hypothesis test where the chance of a result being statistically significant or insignificant is determined.

Table 9. Wilcoxon Signed Rank Sum Test



	KNN	Naive Bayes	Random Forest	AdaBoost	Decision Tree	Stacking Classifier	Voting Classifier
KNN	nan	0.011616	0.114850	0.097656	0.039062	0.232804	0.326396
Naive Bayes	0.011616	nan	0.003906	0.003906	0.007812	0.003906	0.003906
Random Forest	0.114850	0.003906	nan	1.000000	0.011311	0.257899	0.027786
AdaBoost	0.097656	0.003906	1.000000	nan	0.011719	0.270181	0.026857
Decision Tree	0.039062	0.007812	0.011311	0.011719	nan	0.011311	0.003906
Stacking Classifier	0.232804	0.003906	0.257899	0.270181	0.011311	nan	0.778374
Voting Classifier	0.326396	0.003906	0.027786	0.026857	0.003906	0.778374	nan

In this research, the tested result indicates that our proposed stacked model considerably outperforms the compared classification models as depicted in Table 9 above.

5. Conclusion

The diagnosis of Parkinson's Disease (PD) is a difficult task because of the lack of a single specific test. Instead, doctors use a mixture of medical history reviews and neurological examinations to detect key symptoms that suggest the possibility of PD. Nevertheless, in some cases, this approach may result in misdiagnosis due to the subjectivity of symptom interpretation and significantly differing symptoms between patients. Machine learning models are proposed as a supporting tool for doctors to detect early PD. In this particular study, an ensemble learning approach is employed, combining three distinct base classifiers: The algorithms Random Forest, k Nearest Neighbors (KNN), and AdaBoost, with Decision Tree serving as the metaclassifier. The ensemble model is superior to individual classifiers, indicating its possible contribution to clinical practitioners.

Moreover, the authors conduct the research beyond traditional methods as they evaluate machine learning algorithms that are either under-exploited or applied to diagnosing PD for the first time. The range of the comparison of the various machine learning methods helps to get an idea of the pros and cons of their use in diagnosing PD. The results have shown that the ensemble learning model made it possible for nuclear experts and clinicians to make decisions more accurately and quickly in clinical settings. In addition, the model's capacity to automate diagnostic aspects looks to be a potential breakthrough for real-life scenarios, which may lead to a more efficient and accurate diagnosis of PD.

Conclusively, the suggested machine learning method represents a huge step forward in PD diagnosis, providing a trustworthy organization to aid medical personnel in more accurate diagnoses and making better decisions. Through its successful deployment, machine learning in healthcare gains more credibility and the need for continuous research to improve the algorithms and the extent of their use is more emphasised.

Potential conflicts of interest: The authors declare that they have no known competing financial interest or personal relationships that could have appeared to influence the work reported in this paper.

Funding information: Not Applicable
Acknowledgements: Not Applicable
Research involving human participants and/or animals: Not Applicable
Informed consent: Not Applicable
Data availability: www.kaggle.com

Author Address:
[1]Assistant Professor, Computer Science and Engineering-Artificial Intelligence, Brainware University, Barasat, West Bengal, India
[2]Associate Professor, Computer Science and Engineering, Brainware University, Barasat, West Bengal, India
[3]Professor, Computer Science and Engineering-Artificial Intelligence, Brainware University, Barasat, West Bengal, India
[4]Research Scholar, Information Technology, GCECT, Kolkata, West Bengal, India

References

1. Chaudhuri A K, Sinha D, Banerjee D K, and Das A 2021 A novel enhanced decision tree model for detecting chronic kidney disease. Network Modeling Analysis in Health Informatics and Bioinformatics, 10, 1-22.
2. Ray A, and Chaudhuri A K 2024 A novel diagnosis system for Parkinson's disease based on ensemble random forest. In Data Driven Science for Clinically Actionable Knowledge in Diseases, pp. 92-107. Chapman and Hall/CRC.
3. Akyol K 2017 A study on the diagnosis of Parkinson’s disease using digitized wacom graphics tablet dataset. Int J Inf Technol Comput Sci, 9, 45-51.
4. Ray A, and Chaudhuri A K 2021 Smart healthcare disease diagnosis and patient management: Innovation, improvement and skill development. Machine Learning with Applications, 3, 100011.
5. Little M A, McSharry P E, Hunter E J, Spielman J, and Ramig L O 2009 Suitability of dysphonia measurements for telemonitoring of Parkinson's disease. IEEE Transactions on Biomedical Engineering, 56(4), 1015-1022.

6. Shahbaba B, and Neal R M 2009 Nonlinear models using Dirichlet process mixtures. Journal of Machine Learning Research, 10, 1829-1850.
7. Psorakis I, Damoulas T, and Girolami M 2010 Multiclass relevance vector machines: Sparsity and accuracy. IEEE Transactions on Neural Networks, 21(10), 1588-1598.
8. Guo G, Chen Y, Li X, and Wu J 2010 An accurate model for Parkinson's disease diagnosis based on mutual information measure and support vector machine. Journal of Medical Systems, 34(4), 629-638.
9. Sakar C O, and Kursun O 2010 Telediagnosis of Parkinson's disease using measurements of dysphonia. Journal of Medical Systems, 34(4), 591-599.
10. Das R 2010 A comparison of multiple classification methods for diagnosis of Parkinson disease. Expert Systems with Applications, 37(2), 1568-1572.
11. Luukka P 2011 Feature selection using fuzzy entropy measures with similarity classifier. Expert Systems with Applications, 38(4), 4600-4607.
12. Ozcift A, and Gulten A 2011 Classifier ensemble construction with rotation forest to improve medical diagnosis performance of machine learning algorithms. Computer Methods and Programs in Biomedicine, 104(3), 443-451.
13. Li M, Wang J, and Liu Y 2012 Fuzzy-based nonlinear transformation and support vector machines for medical data classification. Artificial Intelligence in Medicine, 55(3), 209-222.
14. Astrom K, and Koker R 2011 Parallel artificial neural network architecture for Parkinson's disease diagnosis. Expert Systems with Applications, 38(10), 12470-12474.
15. Sakar C O, and Kursun O 2010 Telediagnosis of Parkinson's disease using measurements of dysphonia. Journal of Medical Systems, 34(4), 591-599.
16. Polat K 2012 Classification of Parkinson's disease using feature weighting method on the basis of fuzzy C-means clustering. International Journal of Systems Science, 43(4), 597-609.
17. Daliri M R 2012 Chi-square kernel support vector machines for the diagnosis of Parkinson's disease. Expert Systems with Applications, 39(2), 1647-1652.
18. Chen L, Zhang X, and Song C 2013 Diagnosis of Parkinson's disease by PCA and fuzzy kNN based on speech. Journal of Computational and Theoretical Nanoscience, 10(1), 759-763.
19. Zuo Y, Wu X, and Li W 2013 A novel hybrid genetic particle swarm optimization algorithm for improving the performance of fuzzy k-nearest neighbor classifier. Neurocomputing, 120, 501-508.
20. Zhang X 2014 Diagnosis of Parkinson's disease utilizing time-frequency characteristics and stacked autoencoders. Journal of Medical Systems, 38(7), 110.

21. Sayed M A, Tayaba M, Islam M T, Pavel M E U, Mia M T, Ayon E H, and Ghosh B P 2023 Parkinson's disease detection through vocal biomarkers and advanced machine learning algorithms. arXiv preprint arXiv:2311.05435.
22. Rovini E, Maremmani C, and Cavallo F 2017 How wearable sensors can support Parkinson's disease diagnosis and treatment: A systematic review. Frontiers in Neuroscience, 11, 555.
23. Ali A M, Salim F, and Saeed F 2023 Parkinson's disease detection using filter feature selection and a genetic algorithm with ensemble learning. Diagnostics, 13(17), 2816.
24. Ya Y, Ji L, Jia Y, Zou N, Jiang Z, Yin H, and Wang E 2022 Machine learning models for diagnosis of Parkinson's disease using multiple structural magnetic resonance imaging features. Frontiers in Aging Neuroscience, 14, 808520.
25. Huang Y, Chaturvedi K, Nayan A-A, Hesamian M H, Braytee A, and Prasad M 2024 Early Parkinson's disease diagnosis through hand-drawn spiral and wave analysis using deep learning techniques. Information, 15(4), 220.
26. Cao Y, Jiang L, Zhang J, Fu Y, Li Q, Fu W, and Fang J 2023 A fast and non-invasive artificial intelligence olfactory-like system that aids diagnosis of Parkinson's disease. European Journal of Neurology.
27. Rahman Sajal M S, Ehsan M T, Vaidyanathan R, Wang S, Aziz T, and Al Mamun K A 2020 Telemonitoring Parkinson's disease using machine learning by combining tremor and voice analysis. Brain Informatics, 7, 127.
28. Manikandan R, Patan R, Gandomi A H, Sivanesan P, and Kalyanaraman H 2020 Hash polynomial two-factor decision tree using IoT for smart health care scheduling. Expert Systems with Applications, 141, 112924.
29. Bashir S, Khan Z S, Khan F H, Anjum A, and Bashir K 2019 Improving heart disease prediction using feature selection approaches. In 2019 16th International Bhurban Conference on Applied Sciences and Technology (IBCAST), pp. 619-623. IEEE.
30. Chaudhuri A K, Das S, and Ray A 2024 An improved random forest model for detecting heart disease. In Data-Centric AI Solutions and Emerging Technologies in the Healthcare Ecosystem, pp. 143-164. CRC Press.
31. Uddin S, Haque I, Lu H, Moni M A, and Gide E 2022 Comparative performance analysis of K-nearest neighbour (KNN) algorithm and its different variants for disease prediction. Scientific Reports, 12(1), 6256.
32. Mahesh T R, Kumar V D, Kumar V V, Asghar J, Geman O, Arulkumaran G, and Arun N 2022 AdaBoost ensemble methods using K-fold cross validation for survivability with the early detection of heart disease. Computational Intelligence and Neuroscience, 2022.
33. Saha S, Seal D B, Ghosh A, and Dey K N 2016 A novel gene ranking method using Wilcoxon rank sum test and genetic algorithm. International Journal of Bioinformatics Research and Applications, 12(3), 263-279.