Engineering Innovations
ISSN: 2813-1002, Vol. 4, pp 47-66
doi:10.4028/p-h0cef4

Submitted: 2022-08-12
Revised: 2022-10-19
Accepted: 2022-10-23
Online: 2023-02-15

A Novel Ensemble of Support Vector Machines for Improving Medical Data Classification

Phuoc-Hai Huynh[1,2,a*] and Van Hoa Nguyen[1,2,b]

[1]Faculty of Information Technology, An Giang University, An Giang, Vietnam

[2]Vietnam National University Ho Chi Minh city, Vietnam

[a]hphai@agu.edu.vn, [b]nvhoa@agu.edu.vn

Keywords: Medical Data Classification, Ensemble Learning, Support Vector Machines

Abstract. In recent years, the increasing volume and availability of healthcare and biomedical data are opening up new opportunities for computational methods to enhance healthcare in many hospitals. Medical data classification is regarded as the challenging task to develop intelligent medical decision support systems in hospitals. In this paper, the ensemble approaches based on support vector machines are proposed for classifying medical data. This research's key contribution is that the ensemble multiple support vector machines use the function kernel in the style of gradient boosting and bagging to produce a more accurate fusion model than the mono-modality models. Extensive experiments have been conducted on forty benchmark medical datasets from the University of California at Irvine machine learning repository. The classification results show that there is a statistically significant difference (p-values < 0.05) between the proposed approaches and the best classification models. In addition, the empirical analysis of forty medical datasets indicated that our models can predict diseases with an accuracy rate of 82.82 and 81.76 percent without feature selection in the preprocessing data stage.

Introduction

In recent years, the use of machine learning in medical diagnostics to assist physicians in disease diagnosis has become widespread. In practice, a doctor often diagnoses a disease based on a patient's symptoms and signs. For this reason, the physician's experience highly affects the diagnostic accuracy. The clinical decision support systems (CDSS) has emerged as a realistic strategy to help doctors diagnose patients swiftly and correctly in an effort to improve healthcare quality [1]. Machine learning is used CDSS to predict and analyze diagnostic decisions to improve inpatient care more accurate and faster diagnoses as well as support doctors' decision-making [2]. In the field of medical informatics, the increase in the accuracy of classification algorithms is regarded as one of the most challenging tasks whose aim is to enhance the diagnosis, prediction, and treatment of disease [2]. Several classifying algorithms have been used in the analysis of medical diagnosis without feature selection. They include support vector machines (SVMs) used by [3], [4] random forests (RF) in [5], [6], *k* nearest neighbors (*K*NN) used by [7], [8], decision trees (DT) exploited in [7], [9], naive Bayes (NB) in [10], and logistics regression (LR) in [11]. Even though many studies have been done and reported in the literature, there is still room for improvement. In medical diagnosis, more and more studies have proposed novel classification algorithms to improve the accuracy of these algorithms. However, the No-Free-Lunch theorem demonstrated no perfect classifier [12]. It has motivated studies to build new classification models for medical data [13].

Recently, ensemble learning is an algorithm which uses multiple learners to boost a mono-modality model's prediction performance in order to maximize classification accuracy with a greater capacity for generalization. In fact, weak learners are used to build ensemble models in the majority of the study analyses of ensemble methods mentioned above. Therefore, it is essential to examine ensembles based on non-weak classifiers, such as SVM. Although several works presented insight for SVM ensemble theory and applications [14], [15]. However, SVM ensembles have not been investigated thoroughly like the decision tree ensemble [16]. In the medical field, SVM ensembles haven't been tested against a lot of data sets for disease diagnosis. This paper examines SVM

ensembles using two approaches to build models with various medical datasets in order to close this gap. This work explores the utility of ensemble SVMs for classifying medical data through two contributions.

Firstly, two ensemble frameworks (Bagging and Gradient boosting) are proposed for medical data classification to combine SVM models using the radial basis function (RBF) kernel to produce an ensemble of classifiers that is more accurate than a single SVM model. The key concept is to use RBF kernel-based constructs for SVM ensembles to strengthen SVM.

Secondly, empirical analysis for forty medical datasets is designed to select suitable classification models. We have been performed a comparative testing of several classifiers namely Linear SVM (LSVM) and SVM using RBF kernel, *KNN*, DT, NB, LR and ensemble of decision trees include random forest (RFs), bagging (BA-DTs) and gradient boosting (GB-DTs). The accuracy (ACC) as well as Area Under the ROC Curve (AUC) of the 12 algorithms are used to compare in experiments.

The remaining sections of the paper are structured as follows. In the section titled "Related works," the classification of medical data is briefly discussed. In the “Materials and Methods” Section, we present our models. We analyze the experiment and present the numerical test results in Section Evaluation before completing in Section Conclusion.

Related Works

After analyzing the literature, we discovered numerous approaches to classifying diseases. These applications involve individual learning techniques as well as combining techniques to improve the performance of models. In the session, we will discuss these techniques and their applications.

On the one hand, there are many individual classifying techniques to predict disease. As shown by [17], [18], the KNN algorithm is used to identify heart disease. According to [10], NB and *KNN* are used in the diagnosis of breast cancer. In another paper [19], behavior determinant-based cervical cancer early detection using NB and LR. [20] says that DT is often used to find the best way to classify a medical diagnosis .

In addition, the [21] is to classify breast cancer survival patterns using SVM, LR, and DT. More specifically, research comparing some of the methods mentioned above has revealed that SVM outperforms a large number of similar methods [22], [23]. In some studies, the expert system based on fuzzy logic was implemented to predict the responses to the treatment method in paper [24].

On the other hand, it is known that combining multiple classifiers, also called "ensemble classifier” often gives better results than a single classifier [25]. In the experiment in the paper [15], improvement of results from the NB algorithms was investigated using the combined ensemble method. This paper showed the improved accuracy of the Bagging of NB algorithm on Breast Cancer Wisconsin (97.51%) and Hepatitis (86.25%) datasets. In [26], single, boosted, and decision tree forest (RF) are used to detect breast cancer [26]. BDTs (97.07%) outperformed DT (95.75%). RF scored 97.51%, beating DT (95.75%) and BDTs (97.07%).

The performance of ensemble models in medical diagnosis has not been extensively studied, with the exception of [27] and [15], which show that bagging and boosting learning perform better than a single model. The fact that the medical dataset is typically class-imbalanced is yet another factor that can make things more complicated. Therefore, judging the effectiveness of the prediction models based solely on their classification accuracy is insufficient [28].

Therefore, in this study, we aim to investigate the usefulness of ensemble SVM for medical diagnosis. In order to evaluate SVM ensemble model, we use classification accuracy, AUC, and classifier training time. So, the results of this study should make it easy for researchers in the future to choose the best algorithms for medical diagnosis.

Materials and Methods

In medical data classification, the popular method for classification models is to build only a single strong classifier to predict labels from data. In recent decades, the ensemble method has been one of the main research directions in machine learning. It combines multiple learners in order to improve the performance of a single classifier [29]. Bagging and boosting are two different approaches to ensembles. While the boosting method will reduce both bias and difference, the bagging method will only reduce the difference between the base models [30].

Support Vector Machines.

The support vector machine (SVM) is a method for classifying data that was made by Vapnik in the 1990s. It is a joint effort between researchers in statistics and machine learning [31]. The main idea behind the SVM algorithm is that it finds the best way to divide datasets into classes by finding the best hyperplane. Figure 1 shows the best hyperplane, which is the one that is the farthest away from the two categories of labeled points on either side. It can be found by maximizing the distance or margin between the supporting planes for each class.

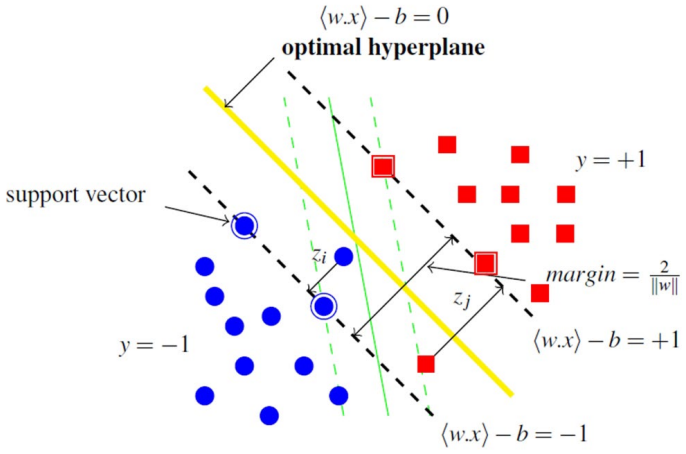

Figure 1: Binary classification using a support vector machine

The margin between these supporting planes is $2/\|w\|$ (where $\|w\|$ is the 2 - norm of the vector w). The quadratic programming of (1) and (2) are used by the standard SVM to endeavor to achieve these goals (2).

$$min_{\propto}(1/2) = \sum_{i=1}^{m}\sum_{j=1}^{m} y_i y_j \alpha_i \alpha_j K\langle x, x_i\rangle - \sum_{i=1}^{m} \alpha_i \tag{1}$$

$$s.t \begin{cases} \sum_{i=1}^{m} x_i \alpha_i = 0 \\ 0 <= \alpha_i <= C \; \forall i = 1,2 \ldots m \end{cases} \tag{2}$$

where C is a positive constant used to tune the margin and the error and a linear kernel function $K\langle x_i, x_j\rangle = K\langle x_i . x_j\rangle$. Using the SVM model, a new data point x is classified as follows (3):

$$predict(x, SVM\ model) = sign(\sum_{i=1}^{SV} y_i \alpha_i K\langle x_i, x_j\rangle - b) \tag{3}$$

The SVM algorithm employs various kernel functions for non-linear classification task [4]. For nonlinear data, the radial basis function (RBF) is a popular kernel function. Kernel function are method for approximating multivariable functions using linear combinations of terms derived from a

single univariable function. The RBF function is given (4), where γ is specified by parameter gamma, must be greater than 0.

$$exp(-\gamma\|x - x'\|^2) \quad (4)$$

Bagging of Support Vector Machines

Bootstrap Aggregation (Bagging) is an ensemble method that combines base classifiers to form a final prediction [32]. Bagging employs bootstrapping estimates to estimate the sampling distribution of almost any statistic using random sampling techniques. To improve the performance of SVM using the RBF kernel for medical data, we propose using Bagging of SVM for medical data. Bagging of Support Vector Machines (denoted by BA-SVMs) constructs a collection of SVM classifiers (SVMs). The main goal of BA-SVMs is to create an ensemble by adding randomization to the SVM construction process and thereby reducing its variance. The Figure 2 shows a schematic diagram of the complete bagging process.

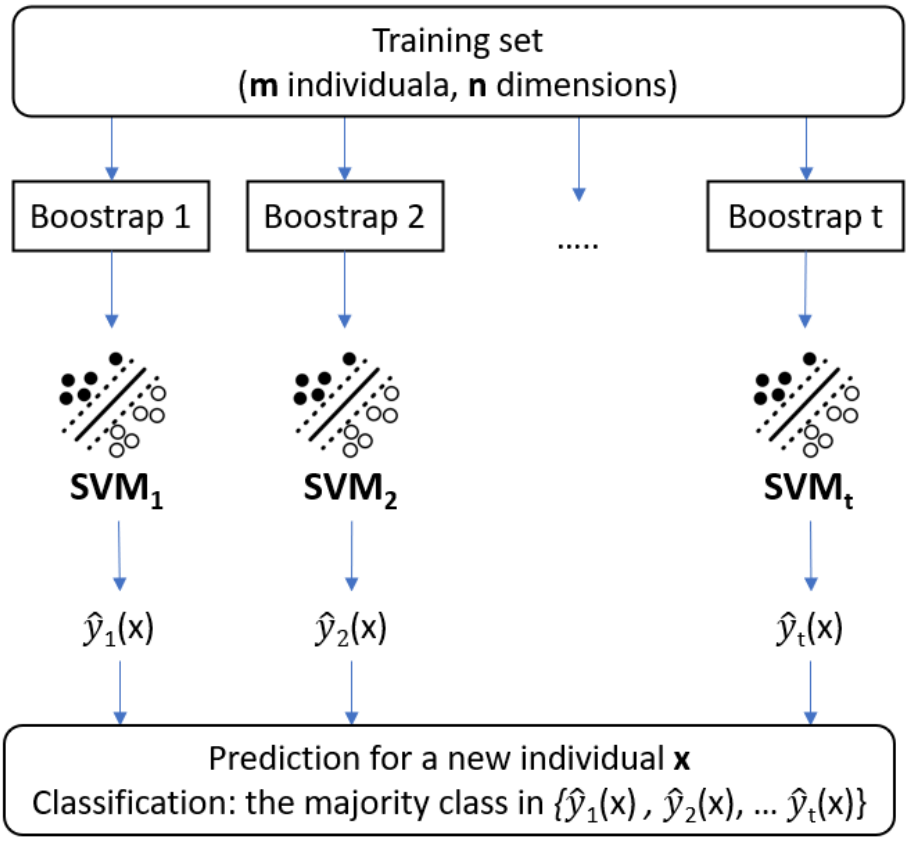

Figure 2. Bagging of Support Vector Machines (BA-SVMs)

The idea of algorithm is simple and appealing:

(1) Given a training set $D = \{(x_1, y_1), \ldots, (x_n, y_n)\}$. Random sampling is used to select n replacement individuals from the initial training set D to create the learning set Bootstrap T.

(2) train a SVM using RBF kernel on each $D_i, i = 1, \ldots, T$ and obtain a sequence of T outputs $f_1(x), \ldots, f_T(x)$.

(3) SVM classifiers that classify new individuals by using a majority vote (as follow (5)).

$$\bar{f}(x) = sign(\sum_{i=1}^{T} sign(f_i(x))) \quad (5)$$

In medical data classification, learning classifying models face numerous challenges, including errors caused primarily by bias, noise, and variance [33]. In Breiman's paper [32], bagging is shown to be able to provide significant accuracy gains. He pointed out that the stability of the base learner is the key factor for the performance of this algorithm. Because bagging is primarily a variance-

reduction method, and since the overall error is the sum of bias and variance. SVM has a built-in mechanism for reducing variance: they look for the classifier with the highest margin among all potential linear separators. In our study, we propose tuning the BA-SVMs to introduce significant changes in the various learning sets while minimizing bias. In order to give the large margin model noise robustness for medical datasets, it is important to set the cost constant C, which trades off margin size and errors, to a high value (more than 100) in SVM learning tasks. Another key idea is reducing the number of samples to be drawn from the training set to train each base estimator. The random addition of discounts ranging from 10% to 30% (depending on the data set). This idea is a little bit bigger when C is high. This is mostly because the unbiased variance is smaller when C is high. The paper [34] says that the relative reduction of the unbiased variance is about 90%, while the bias is mostly the same.

Gradient boosting of Support Vector Machines

Boosting [35] has been one of the most important developments in classification problems in the last 20 years. The boosting technique is used to make a model with fewer mistakes because it focuses on improving the positive aspects and fixing the bad elements of a single model. Unlike with the bagging approach, the target of boosting is to make the prediction model more powerful by reducing its bias. The boosting models are designed from weak classifiers such as DT, LR, NB, and *KNN*. In this paper, we propose using the Gradient Boosting to improve accuracy of SVM with RBF kernel for medical data classification.

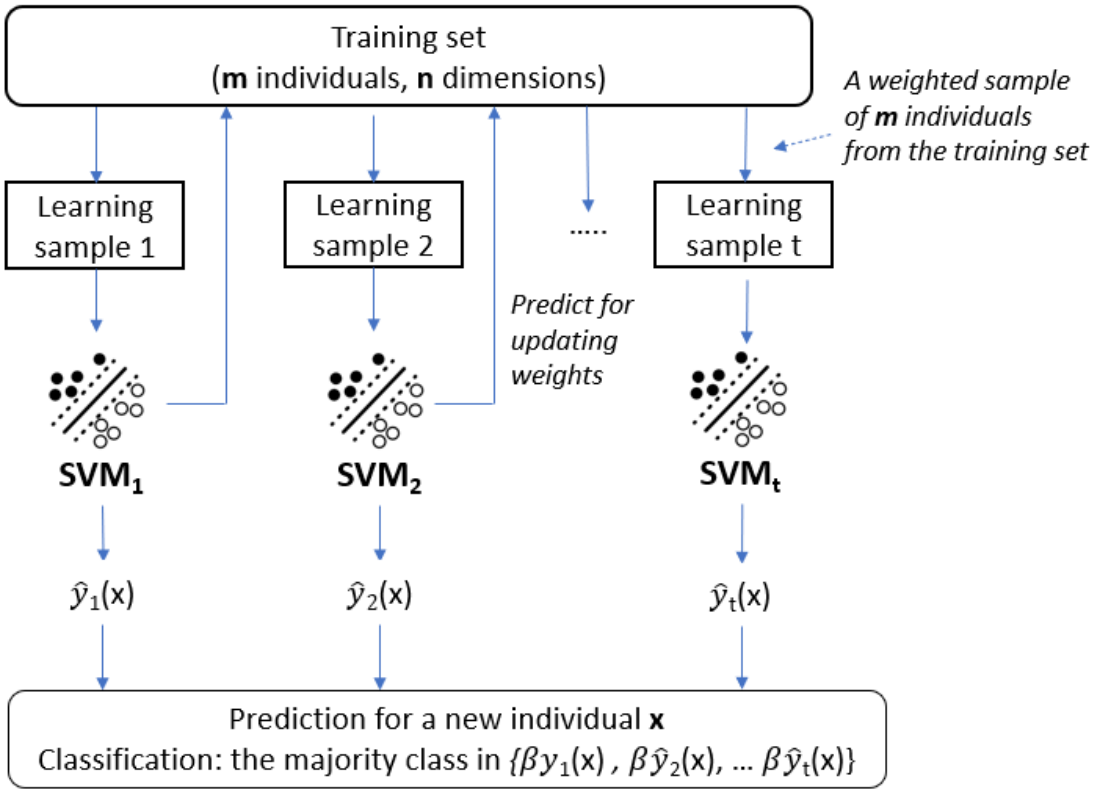

Figure 3: Gradient Boosting of Support Vector Machines (GB-SVMs)

Figure 3 shows a diagram of the whole process of boosting. The main idea behind boosting is to keep adding new models to the ensemble sequentially. By starting with an SVM model, boosting attacks the trade-off between bias and variance. The algorithm improves performance by building more and more models, with each new SVM trying to fix where the last one went wrong the most. This means that each new SVM in the sequence will focus on the training rows where the previous SVM made the most wrong predictions. SVM classifiers use a weighted majority vote to decide how to put a new person, x, into a category.

In our algorithm, the learning procedure consecutively fits new classifiers to compute a more accurate estimate of the response error of the previous step. The main idea behind GB-SVMs is to build SVM models that are most similar to the loss function of the negative gradient of the whole ensemble. The loss function is to be optimized for classification with probabilistic outputs. In GB-SVMs, the *friedman-mse* loss function is a modified kind of *mse* loss function, especially for boosting algorithms. It defines as (4), where N is the number of samples.

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \qquad (4)$$

Boosting algorithm has been demonstrated that boosting can improve the predictive performance of weak learners like decision trees, but it does not improve the performance of strong learners like SVM. Using SVM as the base learner in boosting does not seem to offer significant advantages in terms of generalization error. Therefore, the main idea of GB-SVM is to tune the C and γ parameters are adjusted to the large margin solution of SVMs and can improve the generalization capacity of GB-SVMs against over-fitting.

Evaluation

Figure 4 displays a flowchart of our model development process. The following steps make up the experimental process.

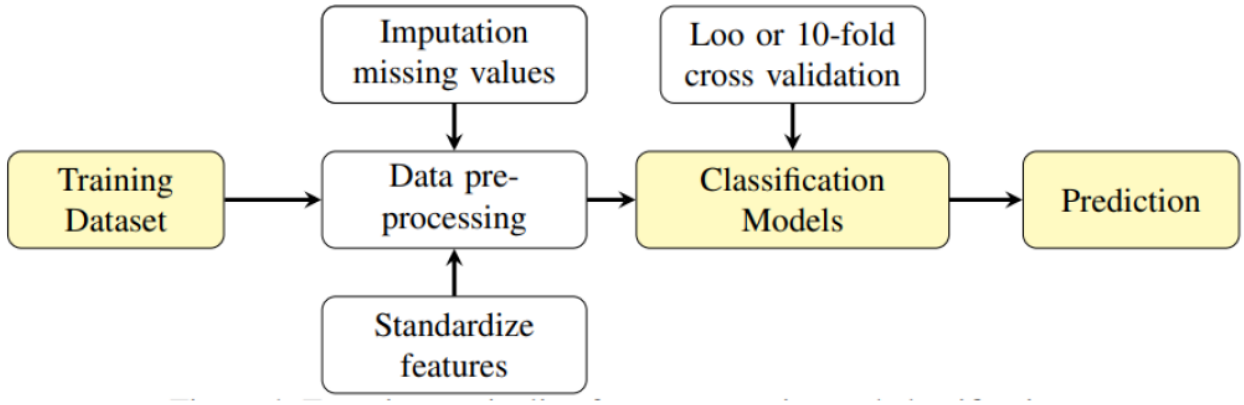

Figure 4. Experiment pipeline for preprocessing and classification

First of all, *KNN* is used for completing missing values for some missing datasets [36]. The training set's n nearest neighbors' mean values are used to impute each sample's missing values. The next step is to transform the processed data. The processed data may contain attributes with a mixture of scales for various quantities. By bringing the scale to unit variance and lowering the mean, we have standardized the features. It is important to standardize a dataset because individual features that don't follow the rules for normally distributed data may not work well.

For each medical dataset, we construct twelve classification models. Firstly, we implement a baseline model, such as LSVM, SVM with an RBF kernel, KNN, DT, NB, and LR. RFs, BA-DTs, and GB-DTs are all types of ensemble models that use decision trees. We are interested in the classification performance of our proposal for medical data. Therefore, the bagging and gradient boosting of SVM are constructed to compare with the classification models (named BA-SVMs and GB-SVMs). Summary, the experiment is implemented with 12 classifiers on forty medical datasets to compare the predictive of the models.

In order to evaluate models, the accuracy measure (ACC) is utilized, which quantifies a classification model's performance as the number of correct predictions divided by the total number of predictions. In addition, the classification models also are evaluated by AUC (Area under the ROC Curve) because AUC is a more discriminating and statistically consistent measure than accuracy.

The paired Student's t-test might then be used to determine whether the difference in mean accuracy and AUC between the two models is statistically significant.

Experiments Setup

Using the Scikit library, we built BA-SVMs and GB-SVMs in Python [37]. As starting points, we use algorithms from the Scikit library like KNN, LR, NB, RF, XGB, BA-DTs, and GB-DTs, as well as the very effective standard linear SVM [38]. The server used was an Intel Xeon(R) CPU PC running at 2.7GHz and having 44 cores. The RAM was 32GB.

Dataset description

The following experiments were conducted using forty benchmark datasets as described in Table 1. were obtained from the Machine Learning Repository at the University of California, Irvine (UCI) [39]. In this table, the protocol evaluation is shown in Column 6. Stratified 10-fold and leave-one-out cross-validation (LOO) are used to evaluate the robustness of the estimates made with the classification models.

Table 1. Brief description of 40 medical datasets are used in this research

ID	Dataset	Number samples	Number features	Number clasess	Protocal evaluation
1	Lung Cancer	32	56	3	Loo
2	Cervical Cancer Behavior Risk	72	19	2	Loo
3	Cryotherapy	90	6	2	Loo
4	Immunotherapy	90	7	2	Loo
5	Breast_tissue	106	9	6	Loo
6	Breast cancer	116	9	2	Loo
7	Lymphography	148	18	4	Loo
8	Hepatitis	155	19	2	Loo
9	Hepatocellular carcinoma	168	49	2	Loo
10	EEG signals (Planning Relax)	182	12	2	Loo
11	Parkinsons	195	22	2	Loo
12	Prognostic Wisconsin Breast Cancer	198	33	2	Loo
13	Thyroid	215	5	3	Loo
14	Parkinson (replicated acoustic features)	240	45	2	Loo
15	Spect heart	267	22	2	Loo
16	Spectf heart	267	44	2	Loo
17	Statlog heart	270	13	2	Loo
18	Breast cancer	286	9	2	Loo
19	Heart failure	299	12	2	Loo
20	Heart cleveland	303	13	5	10-Folds
21	Haberman's Survival	306	3	2	10-Folds
22	Primary tumor	339	16	3	10-Folds
23	Bupa Liver Disorders	345	6	2	10-Folds

24	Dermatology	366	34	4	10-Folds
25	Chronic	400	24	2	10-Folds
26	Arrhythmia	452	279	13	10-Folds
27	SA heart	462	9	2	10-Folds
28	Thoraric surgery	470	16	2	10-Folds
29	Diabetes sylhet	520	16	2	10-Folds
30	Breast Cancer (wdbc)	569	30	2	10-Folds
31	ILPD	583	10	2	10-Folds
32	HCV	615	12	5	10-Folds
33	Soybean	683	35	19	10-Folds
34	Breast cancer	699	9	2	10-Folds
35	Pima	768	8	2	10-Folds
36	Cervical cancer	858	35	2	10-Folds
37	Mammographic	961	5	2	10-Folds
38	Messidor	1151	19	2	10-Folds
39	Parkinson	1208	27	170	10-Folds
40	Myocardial infarction complications	1700	122	8	10-Folds

Tuning parameters

The ensemble models of decision trees are trained from 50 to 200 trees. The *KNN* algorithm uses k among 1, 3, 5, 7, 9. For DT, the Gini index is used to split a decision tree. LR needs to choose C parameters to find out the best C value for it. The C of LR and is used among 1, 10^1, 10^2, 10^3, 10^4. For a linear SVM, the cost constant C is tuned among 1, 10^1, 10^2, 10^3, 10^4. SVM using the RBF kernel has two hyper-parameters (C or γ values). For C, we consider many cases in which all coefficients are constant and have the values 1, 10^1, 10^2, 10^3, 10^4., respectively. The γ is selected in among 1, 10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}. For the NB algorithm, we use Gaussian Naive Bayes to classify for all datasets.

The performance of BA-SVMs and GB-SVMs is determined by the number of SVMs and the RBF kernel parameters, which aids SVM in finding the optimal separating hyperplane in the feature space. In order to find the best hyper-parameters of BA-SVMs and GB-SVMs models, we tune the number of SVM of ensemble model (No. of SVM) and C, γ. We tried 25 to 200 SVMs to get the best results. In addition, we also tuned the RBF kernel's γ and C parameters for accuracy. For γ, its value is 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1, 10, 10^2 and *auto*}. If γ is '*auto*', uses 1 / number of features. For C, The cost constant C is tuned among 10^{-2}, 10^{-1}, 1, 10, 10^2}.

Table 2 displays the top parameters. The finding best parameters of BA-SVMs and GB-SVMs in this paper are very benefit for the related studies. Table 2 show that the C constant to high value (more than 100) in BA-SVMs algorithms make improve the effective classification for medical dataset. Additionally, depending on the data set, the bootstrap process of BA-SVMs has decreased the number with samples of reductions ranging from 10 to 30%. For GB-SVMs, the tuning C and γ show that this algorithm has the best accuracy with C contain is low value (less than 100) in algorithm.

Table 2: The best tuning parameters of BA-SVMs and GB-SVMs

ID	BA-SVMs			GB-SVMs			ID	BA-SVMs			GB-SVMs		
		C	γ		C	γ			C	γ		C	γ
1	150	1	10^{-2}	150	1	1	21	200	10	10^{-3}	200	0,1	*auto*
2	150	10^{2}	10^{-3}	50	10^{2}	*Auto*	22	50	1	scale	50	0,01	0,05
3	200	10^{2}	10^{-2}	25	10^{2}	0,01	23	150	10	0,01	150	10^{2}	*auto*
4	200	0,01	*Auto*	50	1	10^{-3}	24	50	1	10	50	1	1
5	200	10	*Auto*	50	10	*auto*	25	50	1	10^{3}	50	1	*auto*
6	200	0,1	10^{-3}	50	0,01	0,1	26	200	10	10^{-3}	200	1	10^{-3}
7	200	10^{2}	10^{-3}	50	10^{2}	*auto*	27	50	10^{3}	10^{-3}	50	10	0,01
8	25	10	10^{3}	200	10^{2}	10	28	150	0,1	10	150	10^{2}	10^{-3}
9	200	10	0,01	50	0,01	*auto*	29	200	10^{2}	0.1	200	1	0,1
10	50	1	1	50	1	1	30	102	10^{2}	*auto*	10^{2}	1	*auto*
11	50	10^{2}	*auto*	50	10	*auto*	31	200	1	0,1	200	10^{2}	0,01
12	50	10^{2}	scale	50	10	*auto*	32	150	10	0,01	150	0,1	0,1
13	50	10^{3}	10^{-3}	50	1	0,2	33	50	10	*auto*	50	1	*auto*
14	50	10	10^{-3}	50	0,1	0,01	34	50	10	0,01	50	1	10^{-3}
15	50	10	10^{-3}	50	0,01	0,1	35	50	1	0,01	50	10	0,1
16	150	10	*auto*	150	0,1	*auto*	36	200	10	10^{-3}	200	0,1	10^{-3}
17	200	1	10^{-3}	50	0,1	10	37	150	10^{3}	10^{2}	150	10^{2}	10^{3}
18	50	10	scale	50	10	10	38	150	10^{3}	0,01	150	10^{2}	10^{-3}
19	200	1	1	50	10^{2}	*auto*	39	102	10^{3}	10^{-3}	10^{2}	10^{2}	10^{-3}
20	200	10	0,01	50	1	10^{-3}	40	102	10^{2}	10^{-3}	10^{2}	10^{-2}	10^{-3}

Classification Results

First of all, Table 3 shows the mean ACC and AUC of classifying models on 40 medical datasets in the experiment. The models we proposed (BA-SVMs and GB-SVMs) did the best job of classifying when compared to other models. The results demonstrate that the GB-SVMs model, which had an ACC and AUC of 82.94% and 81.40%, respectively, attained the best classification performance. In addition, the BA-SVMs model also has better classification results than other classifiers, with an ACC of 81,41% and an AUC of 79,44%. As the classification results show, it is clear that BA-SVMs provide more accuracy than the individual SVM classifiers and base classifiers. In the classification models, the GB-SVMs algorithm has the highest accuracy compared to the others, while logistic regression and KNN have the lowest accuracy for close movement. In addition, the ensemble decision tree models include RFs, BA-DTs, and GB-DTs that improve the accuracy of simple decision trees. This shows that the ensemble approaches could enhance the performance of base classifiers. However, the contribution of this work shows that using the strong base classifier (SVM) in the ensemble method has better performance compared to the weak base classifiers (decision trees, decision stump) for medical data. The explanation of this result is the following:

A low C value (less than 100) smooths the decision surface for GB-SVMs, iteratively learning SVM classifiers, and after a SVM learner is added, the data weights are readjusted, a process known

as "re-weighting." Thus, future SVM learners focus more on the examples that previous SVM learners misclassified.

On the other hand, for BA-SVMs, a high value C (more than 100) aims at classifying all training examples correctly. This is reason lead to overfiting issue of the base classifer. However, the combining of predictions from multiple separate models handles this problem.

Table 3: The mean accuracy and AUC of twelve algorithms are compared

Models	LR	NB	*K*NN	DT	BA DTs	GB DTs	RF	XGB	LSVM	SVM	BA SVMs	GB SVMs
ACC	74.61	74.63	78.39	79.89	80.34	80.31	80.21	80.07	78.78	79.78	81.76	**82.82**
AUC	75.04	71.22	75.17	73.02	75.35	77.21	76.75	75.29	74.72	77.16	80.06	**80.52**

Furthermore, the mean ACC and AUC scores of models are illustrated in the graph in Figure 5. The bar chart shows the mean percentage of the classification results on 40 medical datasets. The chart shows the percentages in 12 algorithms, the BA-SVMs, the GB-SVMs, and other state-of-the-art algorithms. As the diagram suggests, it shows that our proposed models are better than the ones that are already out there. As is observed, GB-SVMs have performance still better BA-SVMs in ensemble approaches. Ensemble decision tree models (RF, XGB, Bag-DTs, and GB-DTs) are better at classifying than algorithms like DT, NB, LSVM, and SVM, which are used alone.

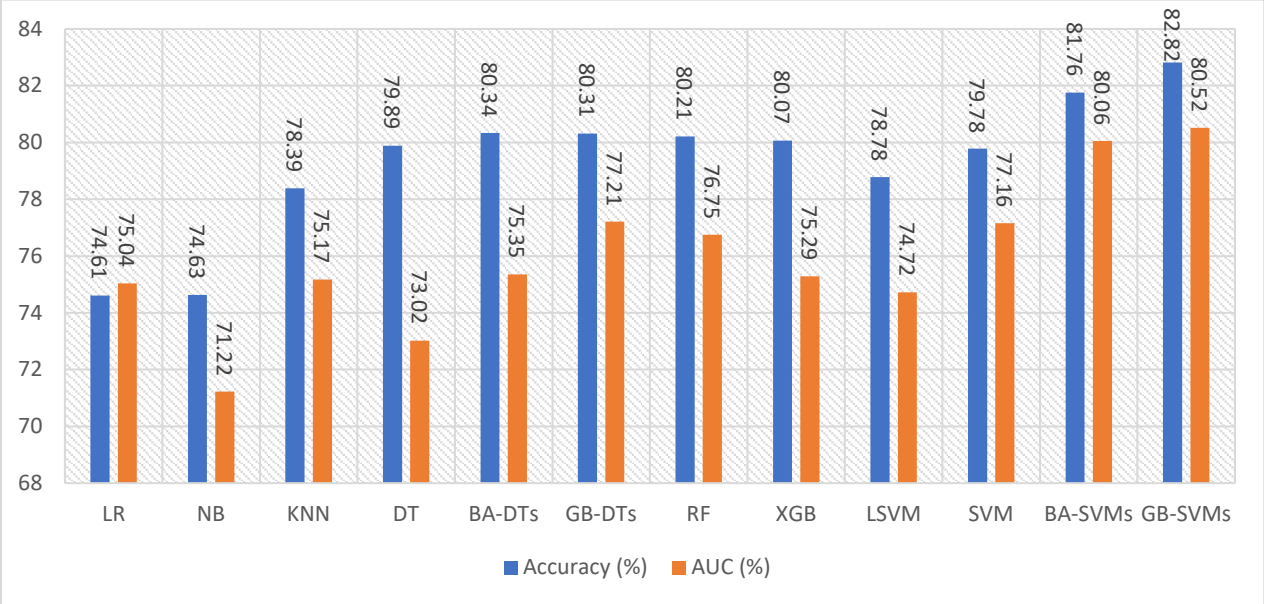

Figure 5: Comparison of the mean Accuracy and AUC among twelve algorithms

Secondly, we investigated the performance of BA-SVMs and GB-SVMs in classification result details on 40 datasets. To evaluate classification models, most studies use an accuracy metric. However, in this work we are not only using accuracy but also AUC. The Area Under the Curve (AUC) is the measure of the ability of a classifier to distinguish between classes and is used as a summary of the ROC curve. Table 4 presents accuracy, and Table 5 shows AUC. Two tables clearly show the results of the 40 datasets. In two tables, the value of each row is the results of classifying 40 medical datasets. The best value is bolded, and this is the best model out of 12 models. Moreover, Table 6 summarizes the results of these statistical tests with paired student ratio tests. The significant results indicate excess p-values just below 0.05 and are reported in bold. The p-values higher than 0.05 (p-value > 0.05) are not statistically significant.

For GB-SVMs, as we can see in Table 6, the performance of GB-SVMs and BA-SVMs allows them to produce significantly better performance (all p-values less than 0.05). In particular, the best

performances are obtained by GB-SVMs for classification (ACC as well as AUC). The classification results of GB-SVMs and BA-SVMs are compared with base classifiers, including KNN, NB, LR, DT, LSVM, and SVM. In detail, the GB-SVMs model outperforms the competitive models in terms of ACC and AUC performance measures. Table 4 also shows that it improves the accuracy mean of 3.37, 7.13, 7.15, 4.44, 2.98, and 1.97 percent points obtained by KNN, NB, LR, DT, LSVM, and SVM, respectively. In addition, it also enhances the AUC mean of KNN, NB, LR, DT, LSVM, and SVM, which is 4.9, 8.84, 5.02, 7.04, 5.25, and 2.9 percent points in Table 5.

For BA-SVMs, in terms of ACC performance metrics, the BA-SVMs algorithm obviously increases the mean accuracy of 3.37, 7.13, 7.15, 4.44, 2.98, and 1.97 percent points compared to KNN, NB, LR, DT, LSVM, and SVM, respectively. In terms of AUC scores, BA-SVMs clearly outperform KNN, NB, LR, DT, LSVM, and SVM by 4.9, 8.84, 5.02, 7.04, 5.25, and 2.9 percent points, respectively. Remarkably, it becomes apparent that GB-SVMs and BA-SVMs outperform performance compared with single SVM models (see Tables 5 and 6). All p-values are less than 0.05, which results in statistically meaningful results. It shows that our proposed models have improved ACC and AUC over the single SVM model.

In addition, we compare GB-SVMs and BA-SVMs with the ensemble of decision tree models. It aims to explore the effectiveness of SVM in the ensemble learning method. In Figure 5, we see that GB-SVMs (ACC: 82,92%, AUC: 80,52%) and BA-SVMs (ACC: 81.76%, AUC: 80.06%) outperform RF (ACC: 80.21%, AUC: 76.75%), BA-DTs (ACC: 79.89%, AUC: 75.8%), GB-DTs (ACC: 80.31%, AUC: 77.21%), and XGB (ACC: 80.07%, AUC: 75.29%). In Table 6, the results of proper statistical analyses reveal that GB-SVMs and BA-SVMs revealed statistically significant comparisons with ensemble models using the decision tree (all p-values are less than 0.05).

Table 4: Accuracy classification results of twelve algorithms on 40 datasets

ID	*K*NN	NB	LR	DT	BA DTs	GB DTs	RF	XGB	LSVM	SVM	BA SVMs	GB SVMs
1	46.88	56.25	46.88	53.12	56.25	62.5	40.62	50	53.12	50	54.06	**65.31**
2	87.5	91.67	94.44	83.47	88.89	88.89	90.28	86.11	**95.83**	93.06	94.16	91.53
3	92.22	85.56	87.78	92.22	91.11	94.44	94.44	93.33	90	92.22	92.11	**95.78**
4	80	78.89	68.89	85.56	86.67	87.78	87.78	81.11	78.89	80	**94.67**	89
5	75.47	64.15	72.64	76.42	72.64	69.81	71.7	71.7	70.75	**78.3**	77.17	77.74
6	71.55	69.83	71.55	75.86	75.86	79.31	73.28	78.45	71.55	75	78.79	**79.91**
7	86.49	83.11	86.49	73.65	83.78	88.51	86.49	85.14	86.49	87.84	88.72	**90.54**
8	80	75.48	67.1	75.29	82.58	81.94	82.58	79.35	76.77	80.65	80.97	**83.42**
9	69.09	67.88	70.91	60.18	70.91	70.91	72.73	**77.58**	73.33	72.73	73.82	76.96
10	67.58	67.03	45.05	70.6	69.78	65.93	71.98	67.03	67.03	74.18	73.63	**73.85**
11	94.36	71.28	86.15	88.67	91.28	93.85	91.28	91.28	86.67	93.85	94.26	**95.38**
12	78.79	66.16	75.76	76.26	81.82	78.28	79.29	80.81	**82.32**	77.78	80.05	82.02
13	96.28	97.74	98.14	91.67	94.88	94.88	95.81	95.35	97.21	**98.14**	98.09	97.76
14	81.67	83.75	80.83	82.17	78.75	80.42	81.67	79.58	80	82.08	**82.21**	81.88
15	68.54	68.54	70.04	74.91	68.91	73.41	71.16	70.04	72.28	72.28	72.4	**74.01**
16	73.78	73.41	77.15	82.9	82.4	78.65	81.65	81.27	80.9	73.41	81.76	**83.56**
17	83.33	84.07	84.81	81.41	80	81.85	82.22	83.33	83.7	84.44	**85.08**	83.92

18	75.52	72.73	68.18	72.73	73.08	74.83	74.48	70.98	73.08	72.73	76.57	**76.68**
19	77.59	76.59	81.61	83.61	81.94	78.93	84.62	81.08	82.61	82.61	82.71	**85.08**
20	**60.07**	54.79	51.49	53.8	56.44	53.47	56.77	56.44	56.11	54.13	56.67	59.18
21	73.86	74.51	73.86	74.64	65.69	72.55	70.26	71.57	75.82	75.16	**75.62**	74.25
22	63.42	61.36	41	62.6	52.8	60.47	60.77	61.65	63.13	56.34	64.69	**65.25**
23	65.8	63.19	68.41	65.74	69.28	68.7	74.49	71.88	71.01	72.46	74.58	**74.84**
24	56.56	51.37	44.54	60.74	53.55	57.92	56.01	57.65	56.83	**59.84**	**59.84**	58.6
25	88.5	73.25	89	97.78	97.5	99.5	99.5	99	84	90.05	90.42	**99.8**
26	60.18	61.73	67.04	64.84	**75.66**	70.35	74.78	72.79	73.01	68.14	71.13	74.76
27	67.97	72.08	69.26	71.78	68.61	72.51	67.97	70.13	73.38	67.1	**73.77**	72.9
28	85.32	84.04	64.89	84.68	84.47	85.11	84.68	84.04	80	85.11	**85.53**	85.32
29	97.5	89.04	92.5	90.96	97.12	97.88	98.65	97.88	92.88	98.08	97.69	**99.04**
30	96.84	93.5	96.84	91.39	96.13	96.31	95.43	96.49	97.19	97.36	97.8	**98.1**
31	71.52	65.87	65.01	69.81	67.75	71.36	71.87	70.33	71.87	69.13	70.91	**72.45**
32	91.22	90.24	89.43	91.06	91.87	93.01	92.36	93.33	92.68	91.71	93.2	**93.6**
33	92.39	84.33	95.31	57.24	94.29	94.14	95.02	94.73	94.88	94.58	94.7	**95.47**
34	95.99	95.85	95.99	93.61	95.71	96.28	96.71	95.71	96.28	96.42	96.63	**96.96**
35	75.13	75.78	75.65	73.05	76.69	75.52	76.17	76.04	77.08	74.48	**77.38**	77.21
36	94.99	94.76	94.99	96.15	95.92	95.57	95.22	95.92	95.8	94.99	**96.09**	96
37	80.44	54.53	55.05	82.83	79.4	**83.87**	79.29	81.89	55.88	82.83	83	83.64
38	63.25	62.9	74.63	63.25	68.9	70.98	68.64	70.98	74.89	**74.46**	74.34	74.13
39	91	84.18	65.24	**92**	91.59	91.47	**92**	**92**	91.71	83.88	89.57	**92**
40	76.98	63.64	79.8	74.11	80.23	80.46	87.65	88.63	74.21	83.75	85.46	**88.84**

Table 5: AUC classification results of twelve algorithms on 40 datasets

ID	*K*NN	NB	LR	DT	BA DTs	GB DTs	RF	XGB	LSVM	SVM	BA SVMs	GB SVMs
1	60.46	67.34	61.47	60.95	67.68	71.88	56.13	63.38	66.56	60.84	64.07	**73.74**
2	87.5	89.92	93.28	80.58	90.53	88.66	89.81	84.81	94.45	91.18	**94.96**	92.24
3	92.26	86.62	88	92.22	91.06	94.44	94.44	93.43	90.62	92.45	92.36	**95.72**
4	70.59	65.85	64.2	78.47	82.62	85.9	88.75	71.47	65.48	71.38	91.75	**91.78**
5	84.75	78.25	83.05	85.45	73.22	80.93	82.33	82.06	81.58	**86.65**	85.95	86.14
6	72.91	70.43	71.65	79.46	75.74	73.03	73.13	78.73	71.26	74.92	80.14	**80.21**
7	89.76	79.18	93.53	62.69	73.51	91.12	74.98	74.33	80.73	93.58	**94.04**	92.11
8	68.1	55.91	62.29	60.94	73.38	72.89	73.62	67.11	52.3	72.07	**84.18**	79.17
9	67.15	66.18	69.5	57.02	69.13	69.09	71.39	76.31	71.99	71.07	72.37	**75.07**
10	44.47	40.17	42.87	54.27	54.15	51.28	64.86	55.42	58.89	**86.72**	86.52	81.43

11	93.22	68.55	80.95	84.89	90.71	93.49	90.71	90.01	82.26	94.36	93	**94.67**
12	75.94	58.33	69.6	**86.53**	78.39	69.45	79.63	74.17	76.02	68.89	73.99	81.11
13	95.49	95.62	98.66	87.26	94.35	94.36	95.29	93.34	96.42	**98.58**	98.5	96.72
14	**83.49**	83.81	80.83	82.17	78.91	80.59	81.89	79.83	80.01	82.74	82.96	82.05
15	67.88	67.69	69.71	75.37	67.86	73.28	70.27	69.11	71.4	71.46	71.58	**74.18**
16	55.3	68.12	68.58	73.01	73.8	60.08	73.45	70.99	70.23	68.12	71.9	**75.78**
17	83.31	84	84.75	81.31	79.81	81.67	82.21	83.31	83.65	84.72	**85.56**	83.84
18	**74.94**	66.57	64.67	67.25	67.09	73.13	69.27	63.89	67.76	66.82	72.51	72.95
19	77.12	74.51	78.83	83.13	79.34	79.79	83.28	61.87	80.65	80.65	**89.41**	83.39
20	59.32	58.24	61.62	55.19	57.64	56.92	58.22	59.09	56.16	51.4	56.6	**58.41**
21	64.29	65.85	66.15	66.3	54.82	61.66	58.66	59.76	71.58	67.04	**67.93**	65.04
22	**56.2**	55.28	55.96	54.59	51.35	52.14	57.83	58.1	55.9	54.95	55.65	54.54
23	64.76	62.1	68.06	64.66	69.23	68.94	74.46	71.21	70.07	71.75	**74.67**	74.53
24	53.84	60.25	58	57.76	54	49.74	53.8	**56.18**	55.04	51.05	51.05	49.97
25	87.46	74.16	86.17	97.87	97.33	99.47	99.6	99.07	82.89	89.87	91.53	**99.84**
26	56.68	68.56	70.29	59.48	71.01	68.2	68.01	69.34	69.91	70.47	67.22	**71.07**
27	63.71	69.45	68.57	68.61	64.71	71.15	63.78	66.58	70.52	68.33	**71.17**	70.08
28	92.64	56.34	55.76	59.34	59.4	91.95	59.34	49.69	55.63	67.7	**92.74**	76.05
29	96.99	88.56	91.7	90.97	96.91	97.81	98.62	97.64	92.29	98.06	97.52	**99.03**
30	96.79	92.89	96.54	91.31	96.23	96.37	95.28	96.41	97.08	97.22	97.72	**98.12**
31	62.45	61.42	**67.35**	52.46	57.87	61.18	63.89	62.8	63.29	62.7	54.97	65.54
32	68.25	68.66	84.76	68.7	73.76	76.46	72.62	76.05	75.43	76.81	75.51	**77.4**
33	97.04	96.36	98.73	64.6	97.89	97.54	98.56	98.19	98.22	98.26	98.04	**98.4**
34	95.65	95.45	95.41	92.52	95.42	95.8	96.24	95.17	95.88	95.71	95.95	**96.51**
35	72.64	73.4	73.57	71.32	74.48	73.27	74.04	73.71	75.47	72.37	**76.5**	75.61
36	83.61	77.47	78.1	81.99	83.14	81.78	82.44	81.99	81.19	78.12	81.88	**84.69**
37	81.31	54.23	62.91	83.77	79.33	**84.18**	79.22	81.85	63.24	82.87	83.15	83.54
38	64.42	62.73	75.39	66.63	69.03	70.94	68.76	70.9	75.37	**75.74**	74.51	74.33
39	62.4	64.37	**74.4**	66.68	66.2	65.72	64.57	67.14	66.98	51.32	64.8	67.68
40	81.67	76.07	85.74	73.09	83.04	82.3	86.58	87.56	78.13	87.63	**87.73**	88.1

Table 6. Statistical Tests for Comparing Classification Algorithms

Comparisons	p-values	Wins	Ties	Loses	p-values	Wins	Ties	Loses
Comparison of the performance of GB-SVMs & the base classifiers								
GB-SVMs & *KNN*	**1.47e-7**	38	1	1	**2.43e-4**	34	0	6
GB-SVMs & NB	**2.32e-8**	36	0	4	**1.43e-6**	35	0	5
GB-SVMs & LR	**2.70e-7**	36	0	4	**7.58e-4**	27	0	13
GB-SVMs &DT	**1.30e-5**	34	1	5	**5.66e-6**	33	0	7
GB-SVMs & LSVM	**5.63e-5**	35	0	5	**7.28e-5**	34	0	6
GB-SVMs & SVM	**4.95e-6**	31	0	9	**2.83e-4**	30	0	10
Comparison of the performance of GB-SVMs &the ensemble models based DTs								
GB-SVMs & RF	**1.15e-4**	38	1	1	**2.89e-5**	37	0	3
GB-SVMs & XGB	**1.62e-7**	38	1	1	**7.59e-5**	36	0	4
GB-SVMs& BA-DTs	**5.86e-01**	39	0	1	**4.18e-1**	39	0	1
GB-SVMs& GB-DTs	**2.78e-8**	39	0	1	**1.66e-3**	36	0	4
Comparison of the performance of BA-SVMs & the base classifiers								
BA-SVMs & *KNN*	**2.18e-6**	35	0	5	**6.69e-4**	31	0	9
BA-SVMs & NB	**7.31e-8**	38	0	2	**2.47e-5**	34	0	6
BA-SVMs & LR	**4.16e-6**	36	0	4	**7.47e-3**	30	0	10
BA-SVMs & DT	**3.19e-4**	32	0	8	**2.03e-4**	32	0	8
BA-SVMs & LSVM	**1.23e-3**	32	0	8	**2.17e-3**	30	0	10
BA-SVMs & SVM	**2.37e-4**	33	1	6	**3.90e-3**	28	1	11
Comparison of the performance of BA-SVMs & the ensemble models of DTs								
BA-SVMs & RF	**1.02e-2**	28	0	12	**9.03e-3**	27	0	13
BA-SVMs & XGB	**4.29e-3**	30	0	10	**6.73e-3**	26	0	14
BA-SVMs & BA-DTs	**1.38e-3**	33	0	7	**5.75e-03**	30	0	10
BA-SVMs & GB-DTs	**1.10e-2**	30	0	10	**1.19e-2**	27	0	13
Comparision GB-SVMs and BA-SVMs								
GB-SVMs & BA-SVMs	**1.16e-2**	28	0	12	**5.29e-1**	24	0	16

Finally, in the comparison between GB-SVMs with BA-SVMs, GB-SVMs is superior to BA-SVMs with 28 wins, 12 defeats, *p-value*=1,16e-2 for using ACC metric. However, GB-SVMs is slightly superior to BA-SVMs with 24 wins and 16 defeats, *p-value*=5.29e-1 (not significantly different) using AUC scores.

We can see that SVM ensembles perform better than single SVM classifiers in most cases, particularly SVMs with an RBF kernel function. There are many ensemble classifiers that can perform the best for all of the evaluation metrics in Table 6. These results allow us to believe that GB-SVMs and BA-SVMs efficiently handle medical data to improve medical diagnosis.

In addition, BA-SVMs employ a divide-and-conquer strategy by aggregating many SVM models, trained on small sub-samples of the training set. It helps total training time decrease significantly,

even though more models need to be trained. Training p classifiers on sub-samples of size $\frac{n}{p}$, results in an approximate complexity of $\Omega\frac{n^2}{p}$. This reduction in complexity helps in dealing with large datasets and nonlinear kernels. Comparison of the computational times for training, GB-SVMs and BA-SVMs need time more than single SVM because these models train many SVM models. For BA-SVMs, this algorithm can run parallel with multiple cores. In our experiment, SVM need 0.86s to train, BA-SVMs only uses 0.27s to train with 16 cores and GB-SVMs uses 6.23s. However, in our opinion, time training of the suggested models is appropriate, particularly when the final prediction model yields the greatest results in terms of classification ACC and AUC.

Comparison with Related Works and Discussion

In order to compare the result of our study with previous studies, we present some related studies that use the same datasets as our experiment. Also, to see how the proposed method compares to similar studies, Table 7 shows the proposed method compares to studies that used medical datasets. Remark, there are only compare to with studies without feature selection in the pre-processing data stage.

Table 7. Comparison with related works

Dataset	**Researches**	**ACC and algorithms used in research**	**ACC of Our work**
Cervical cancer	[19]	87.5% (logistic regression)	**91.53%**
Cryotherapy	[7]	92.22% (decision tree), 94.44 (random forest), 95.56 (k nearest neighbor)	**95.78%**
Immunotherapy	[40]	85.6% (feedforward neural network)	**89%**
Lymphography	[41]	84.46% (MLP) 77.03% (J48)	**90.54%**
Hepatitis	[3]	84.07% (SVM)	83.42%
Hepatocellular carcinoma	[42]	72,1% (logistic regression)	**76.96%**
EEG signals (Planning Relax)	[43]	85,73% (fuzzy-rough trees)	73.85%
Parkinsons	[44]	95% (deep forest)	**95.38%**
Thyroid	[8]	94.4% (k-nearest neighbor)	97.76%
Parkinson	[45]	86.2%	81.88%
SPECT heart	[46]	71.12 % (SVM)	**74.01%**
SPECTF heart	[47]	74.11% (Linear SVM)	**83.56%**
Statlog heart	[48], [49]	94% (deep learning), 87.4%	83.92%
Haberman's Survival	[50]	75.18 % (L-perceptron)	74.25%
Bupa Liver Disorders	[6]	80% (random forest)	74.84%
Chronic	[51]	97% (deep ANN)	**99.8%**
Arrhythmia	[52]	66% (SVM), 72% (random forest)	**74.76%**
Thoraric surgery	[53]	85,53 (XGB), 81,89% (RF), 84,36% (SVM)	**85.32%**
Diabetes sylhet	[54]	96% (neural network)	**99.04%**
Breast Cancer Wisconsin (Diagnostic)	[9]	97%(SVM, decision jungle), 95% (decision tree)	**98.1%**
Indian Liver	[55]	71.87% (random forest)	**72.45%**
Hepatitis C virus	[56]	94,4% (KNN)	93.6%
Soybean	[57]	89,22% (SVM), 91,18%(ANN)	**95.47%**

In Table 7, we present some studies related to medical data classification. Each row show information to compare between the related work and our models. The first column is the dataset name that is used in the experiment. The second column is a source reference for research. Their classification accuracy is shown in Column 3. The last column presents classification accuracy for our models (GB-SVMs or BA-SVMs). The values improved are bold in Table 7. These comparisons show that an ensemble of support vector machines can predict disease based on medical data with good accuracy for many types of complex diseases. However, for a few medical datasets, these proposed algorithms are not suitable due to their low classification accuracy compared to other algorithms.

Furthermore, Table 7 also shows the characteristics of each data mining modality. Despite all the work, no widely used method exists for organizing medical data. The most reasonable explanation is because the medical field demands extremely precise data, particularly a very low percentage of false negatives. However, there are general methods that can be applied to a wide range of data. Specialized methods for specific applications that take prior knowledge into account can lead to improved performance.

However, there are challenges, including data mining techniques, unbalanced data, noisy data, performance, and scalability. Therefore, choosing an appropriate approach to a classification problem can thus be a difficult decision, and medical data classification tasks can be improved. Since there hasn't been a lot of research on how to use data mining techniques to classify medical data, there is room for more research and interesting research opportunities.

Conclusion and Future Works

In this research, we have suggested two methods for classifying medical data: GB-SVMs and BA-SVMs. The study aims to discover the best algorithms for medical data through experimentation and analysis on various datasets. In order to create an ensemble of classifiers that is more accurate than a single SVM model, our suggested method entails training several SVM classifiers in the techniques of bagging and gradient boosting. The results indicate that the GB-SVMs and BA-SVMs outperform the individual model, base classification models, and ensemble method-based decision trees on forty medical datasets.

Future research can use SVM classifiers to implement the additional ensemble learning methods of stacking and voting for the classification of medical data. Additionally, a hybrid model that combines feature selection and classification algorithms can be utilized to enhance the accuracy of predictions for medical datasets.

Appendix:

List of Abbreviations

Abbreviation	Definition
CDSS	Clinical decision support system
SVM	Support vector machine
RF	Random forests
KNN	K nearest neighbors
DT	Decision trees
NB	Naive Bayes
LR	Logistics regression
RBF	Radial basis function

LSVM	Linear support vector machine
ACC	Accuracy
AUC	Area Under the ROC Curve
BDT	boosted decision tree
BA-SVMs	Bagging of support vector machines
GB-SVMs	Gradient boosting of support vector machines
BA-DTs	Bagging of decision trees
GB-DTs	Gradient boosting of decision trees
XGB	Extreme Gradient Boosting
UCI	University of California at Irvine
LOO	Leave one out

References

[1] F. Hak, T. Guimarães, and M. Santos, "Towards effective clinical decision support systems: A systematic review," *PloS One*, vol. 17, no. 8, p. e0272846, 2022.

[2] R. T. Sutton, D. Pincock, D. C. Baumgart, D. C. Sadowski, R. N. Fedorak, and K. I. Kroeker, "An overview of clinical decision support systems: benefits, risks, and strategies for success," *NPJ Digit. Med.*, vol. 3, no. 1, pp. 1–10, 2020.

[3] M. Joshi and A. Jetawat, "Evaluation of Classification Algorithms used in Medical Decision Support Systems," in *2020 Fourth World Conference on Smart Trends in Systems, Security and Sustainability (WorldS4)*, 2020, pp. 27–31.

[4] S. Ghosh, A. Dasgupta, and A. Swetapadma, "A study on support vector machine based linear and non-linear pattern classification," in *2019 International Conference on Intelligent Sustainable Systems (ICISS)*, 2019, pp. 24–28.

[5] M. A. Putra, N. A. Setiawan, and S. Wibirama, "Wart treatment method selection using AdaBoost with random forests as a weak learner," *Commun. Sci. Technol.*, vol. 3, no. 2, pp. 52–56, 2018.

[6] M. R. Haque, M. M. Islam, H. Iqbal, M. S. Reza, and M. K. Hasan, "Performance evaluation of random forests and artificial neural networks for the classification of liver disorder," in *2018 international conference on computer, communication, chemical, material and electronic engineering (IC4ME2)*, 2018, pp. 1–5.

[7] G. A. Rahmat, R. Primartha, A. Wijaya, and others, "Comparative analysis of classification method for wart treatment method," in *Journal of Physics: Conference Series*, 2019, vol. 1196, no. 1, p. 012012.

[8] P. Viswanath and T. H. Sarma, "An improvement to k-nearest neighbor classifier," in *2011 IEEE Recent Advances in Intelligent Computational Systems*, 2011, pp. 227–231.

[9] K. Alshouiliy, A. Shivanna, S. Ray, A. AlGhamdi, and D. P. Agrawal, "Analysis and Prediction of Breast Cancer using AzureML Platform," in *2019 IEEE 10th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*, 2019, pp. 0212–0218.

[10] I. Handayani and I. Ikrimach, "Accuracy Analysis of K-Nearest Neighbor and Naïve Bayes Algorithm in the Diagnosis of Breast Cancer," *J. INFOTEL*, vol. 12, no. 4, pp. 151–159, 2020.

[11] J. Sultana and A. K. Jilani, "Predicting breast cancer using logistic regression and multi-class classifiers," *Int. J. Eng. Technol.*, vol. 7, no. 4.20, pp. 22–26, 2018.

[12] S. P. Adam, S.-A. N. Alexandropoulos, P. M. Pardalos, and M. N. Vrahatis, "No free lunch theorem: A review," *Approx. Optim.*, pp. 57–82, 2019.

[13] S. Rauschert, K. Raubenheimer, P. Melton, and R. Huang, "Machine learning and clinical epigenetics: a review of challenges for diagnosis and classification," *Clin. Epigenetics*, vol. 12, no. 1, pp. 1–11, 2020.

[14] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4700–4708.

[15] R. Rosly, M. K. Makhtar, M. I. Awang, and M. N. A. Rahman, "Analyzing performance of classifiers for medical datasets," *Int J. Eng. Technol.*, vol. 7, no. 2.15, pp. 136–138, 2018.

[16] F. Khan, B. V. V. Siva Prasad, S. A. Syed, I. Ashraf, and L. K. Ramasamy, "An Efficient, Ensemble-Based Classification Framework for Big Medical Data," *Big Data*, vol. 10, no. 2, pp. 151–160, 2022.

[17] H. S. Khamis, K. W. Cheruiyot, and S. Kimani, "Application of k-NN classification in medical data mining," *Int. J. Inf. Commun. Technol. Res.*, vol. 4, no. 4, 2014.

[18] I. K. A. Enriko, M. Suryanegara, and D. Gunawan, "Heart disease prediction system using k-Nearest neighbor algorithm with simplified patient's health parameters," *J. Telecommun. Electron. Comput. Eng. JTEC*, vol. 8, no. 12, pp. 59–65, 2016.

[19] R. Machmud, A. Wijaya, and others, "Behavior determinant based cervical cancer early detection with machine learning algorithm," *Adv. Sci. Lett.*, vol. 22, no. 10, pp. 3120–3123, 2016.

[20] E. Turanoglu-Bekar, G. Ulutagay, and S. Kantarcı-Savas, "Classification of thyroid disease by using data mining models: a comparison of decision tree algorithms," *Oxf. J. Intell. Decis. Data Sci.*, vol. 2, pp. 13–28, 2016.

[21] C.-M. Chao, Y.-W. Yu, B.-W. Cheng, and Y.-L. Kuo, "Construction the model on the breast cancer survival analysis use support vector machine, logistic regression and decision tree," *J. Med. Syst.*, vol. 38, no. 10, pp. 1–7, 2014.

[22] P. Janardhanan, F. Sabika, and others, "Effectiveness of support vector machines in medical data mining," *J. Commun. Softw. Syst.*, vol. 11, no. 1, pp. 25–30, 2015.

[23] T. A. Assegie, "SVM And kNN Based Liver Disease Classification Model," *Indones. J. Electron. Electromed. Eng. Med. Inform.*, vol. 3, no. 1, pp. 9–14, 2021.

[24] F. Khozeimeh, R. Alizadehsani, M. Roshanzamir, A. Khosravi, P. Layegh, and S. Nahavandi, "An expert system for selecting wart treatment method," *Comput. Biol. Med.*, vol. 81, 2017.

[25] J. Kittler, M. Hatef, R. P. Duin, and J. Matas, "On combining classifiers," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 3, pp. 226–239, 1998.

[26] A. T. Azar and S. M. El-Metwally, "Decision tree classifiers for automated medical diagnosis," *Neural Comput. Appl.*, vol. 23, no. 7, pp. 2387–2403, 2013.

[27] D. Lavanya and K. U. Rani, "Ensemble decision making system for breast cancer data," *Int. J. Comput. Appl.*, vol. 51, no. 17, 2012.

[28] Z. Vujović, "Classification model evaluation metrics," *Int. J. Adv. Comput. Sci. Appl.*, vol. 12, no. 6, pp. 599–606, 2021.

[29] X. Dong, Z. Yu, W. Cao, Y. Shi, and Q. Ma, "A survey on ensemble learning," *Front. Comput. Sci.*, vol. 14, no. 2, pp. 241–258, 2020.

[30] Y. Liu and Q. Zhao, "Ensemble learning," in *HANDBOOK ON COMPUTER LEARNING AND INTELLIGENCE: Volume 2: Deep Learning, Intelligent Control and Evolutionary Computation*, World Scientific, 2022, pp. 635–660.

[31] D. A. Pisner and D. M. Schnyer, "Support vector machine," in *Machine learning*, Elsevier, 2020, pp. 101–121.

[32] L. Breiman, "Bagging predictors," *Mach. Learn.*, vol. 24, no. 2, pp. 123–140, 1996.

[33] A. Kalantari, A. Kamsin, S. Shamshirband, A. Gani, H. Alinejad-Rokny, and A. T. Chronopoulos, "Computational intelligence approaches for classification of medical data: State-of-the-art, future challenges and research directions," *Neurocomputing*, vol. 276, pp. 2–22, 2018.

[34] G. Valentini and T. G. Dietterich, "Low Bias Bagged Support Vector Machines," in *Proceedings of the Twentieth International Conference on International Conference on Machine Learning*, Washington, DC, USA, 2003, pp. 752–759.

[35] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *J. Comput. Syst. Sci.*, vol. 55, no. 1, pp. 119–139, 1997.

[36] L. Beretta and A. Santaniello, "Nearest neighbor imputation algorithms: a critical evaluation," *BMC Med. Inform. Decis. Mak.*, vol. 16, no. 3, p. 74, 2016.

[37] F. Pedregosa *et al.*, "Scikit-learn: Machine Learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, 2011.

[38] C.-C. Chang and C.-J. Lin, "LIBSVM: a library for support vector machines," *ACM Trans. Intell. Syst. Technol. TIST*, vol. 2, no. 3, p. 27, 2011.

[39] A. Asuncion and D. Newman, *UCI machine learning repository*. Irvine, CA, USA, 2007.

[40] A. Çifci and M. Şimşir, "A Study on Method Prediction for a Better Directed Treatment of Warts," 2019.

[41] R. Arora, "Comparative analysis of classification algorithms on different datasets using WEKA," *Int. J. Comput. Appl.*, vol. 54, no. 13, 2012.

[42] M. S. Santos, P. H. Abreu, P. J. García-Laencina, A. Simão, and A. Carvalho, "A new cluster-based oversampling method for improving survival prediction of hepatocellular carcinoma patients," *J. Biomed. Inform.*, vol. 58, pp. 49–59, 2015.

[43] R. B. Bhatt and M. Gopal, "FRCT: fuzzy-rough classification trees," *Pattern Anal. Appl.*, vol. 11, no. 1, pp. 73–88, 2008.

[44] L. V. Utkin and M. A. Ryabinin, "Discriminative metric learning with deep forest," *Int. J. Artif. Intell. Tools*, vol. 28, no. 02, p. 1950007, 2019.

[45] L. Naranjo, C. J. Perez, J. Martin, and Y. Campos-Roca, "A two-stage variable selection and classification approach for Parkinson's disease detection by using voice recording replications," *Comput. Methods Programs Biomed.*, vol. 142, pp. 147–156, 2017.

[46] M. L. Samb, F. Camara, S. Ndiaye, Y. Slimani, and M. A. Esseghir, "A novel RFE-SVM-based feature selection approach for classification," *Int. J. Adv. Sci. Technol.*, vol. 43, no. 1, pp. 27–36, 2012.

[47] I. Syarif, A. Prugel-Bennett, and G. Wills, "SVM parameter optimization using grid search and genetic algorithm to improve classification performance," *TELKOMNIKA Telecommun. Comput. Electron. Control*, vol. 14, no. 4, pp. 1502–1509, 2016.

[48] S. Sajeev *et al.*, "Deep learning to improve heart disease risk prediction," in *Machine Learning and Medical Engineering for Cardiovascular Health and Intravascular Imaging and Computer Assisted Stenting*, Springer, 2019, pp. 96–103.

[49] M. S. Amin, Y. K. Chiam, and K. D. Varathan, "Identification of significant features and data mining techniques in predicting heart disease," *Telemat. Inform.*, vol. 36, pp. 82–93, 2019.

[50] H. Mansourifar and W. Shi, "Toward efficient breast cancer diagnosis and survival prediction using L-perceptron," *ArXiv Prepr. ArXiv181103016*, 2018.

[51] H. Kriplani, B. Patel, and S. Roy, "Prediction of chronic kidney diseases using deep artificial neural network technique," in *Computer aided intervention and diagnostics in clinical and medical images*, Springer, 2019, pp. 179–187.

[52] A. Gupta, A. Banerjee, D. Babaria, K. Lotlikar, and H. Raut, "Prediction and classification of cardiac arrhythmia," in *Sentimental Analysis and Deep Learning*, Springer, 2022, pp. 527–538.

[53] S. Xu, "Machine Learning-Assisted Prediction of Surgical Mortality of Lung Cancer Patients.," in *ICDM (Posters)*, 2019, pp. 46–51.

[54] J. Ma, "Machine Learning in Predicting Diabetes in the Early Stage," in *2020 2nd International Conference on Machine Learning, Big Data and Business Intelligence (MLBDBI)*, 2020, pp. 167–172.

[55] A. Gulia, R. Vohra, and P. Rani, "Liver patient classification using intelligent techniques," *Int. J. Comput. Sci. Inf. Technol.*, vol. 5, no. 4, pp. 5110–5115, 2014.

[56] K. Ahammed, M. S. Satu, M. I. Khan, and M. Whaiduzzaman, "Predicting infectious state of hepatitis c virus affected patient's applying machine learning methods," in *2020 IEEE Region 10 Symposium (TENSYMP)*, 2020, pp. 1371–1374.

[57] M. Morgan, C. Blank, and R. Seetan, "Plant disease prediction using classification algorithms," *IAES Int. J. Artif. Intell.*, vol. 10, no. 1, p. 257, 2021.