Celal Bayar University Journal of Science

Majority Vote Decision Fusion System to Assist Automated Identification of Vertebral Column Pathologies

Akın Özçift[1]*, Mehmet Bozuyla[2]

[1] Department of Software Engineering, Hasan Ferdi Turgutlu Technology Faculty, Manisa Celal Bayar University, Manisa, Türkiye
[2] Akürün Textile Corporation, Turkey
*akinozcift@mcbu.edu.tr
* Orcid No: 0000-0003-2840-1917

Received: 3 March 2022
Accepted: 2 March 2023
DOI: 10.18466/cbayarfbe.1082067

Abstract

This paper presents a majority vote decision fusion system called AIVCP (Automated Identification of Vertebral Column Pathologies). With this aim, we proposed a three-step decision fusion algorithm: In the first step, a pool of algorithms from different groups is obtained and the number of classifiers is decreased to 10 with the use of prediction accuracy and classifier diversity concept. As a second step, different majority vote combinations of 10 algorithms are searched with a grid search strategy guided on top of 10-fold cross validation evaluation and with prediction error analysis. In the second step, we obtained four base classifiers, i.e., Naïve Bayes (NB), Simple Logistics (SL), Learning Vector Quantization (LVQ) and Decision Stump (DS) whose majority vote decision fusion generate the most accurate diagnosis rate in Vertebral Column Pathologies domain. As the third step, we applied a Support Vector Machine based feature selection to increase prediction performance of the proposed system further. The experiments are evaluated with the use of 10-fold cross-validation, Sensitivity, Specificity and Confusion Matrices. The experimental results have shown that NB, SL and LVQ single classifiers generate 0.780, 0.829 and 0.786 average diagnosis f-scores respectively. On the other hand, majority vote decision fusion of these single predictors produces 0.883 f-score value that is higher than each of the constituents. The resultant diagnosis f-score value of Vote algorithm for Vertebral column pathologies is enhanced.

Keywords: Majority voting, decision fusion, multiple-classifier systems, vertebral column pathologies

1. Introduction

There is a continuous effort to design computer-based clinical decision support systems (CDSSs) to assist clinical decision making. In this concept, CDSSs are designed software for furthers helping clinical decision-making based on the computerized clinical knowledge base[1].

CDSSs generally makes use of an inductive engine to learn the decision characteristics of a specific disease and then to use the proposed strategy in the diagnosis of unseen instances of the disease [2]. A high-accurate CDDS design normally comprises a three step approach: (i) preprocessing of data, (ii) feature mining in the form of feature selection or feature transformation and (iii) an intelligent decision algorithm proposal. Though, the order and necessity of those steps may change from one application to another, the general workflow is presented in Figure 1.

Step 1: Input raw data with n features, $D_i=\{f_1,f_2,...,f_n\}$
Step 2: Preprocess input data D_i
else skip this step
Step 3: Make feature selection, $D_i=\{f_1,f_2,...,f_m\}$ for m<n Or
Make feature transformation
else skip this step
Step 4: Classify D_i

Figure1. Basic CDSSs design steps.

Many machine learning (ML) algorithms are used in automated medical diagnosis literature to obtain accurate CDSSs. Taxonomy of these algorithms can be summarized as follows: (i)Logic based algorithms, i.e., decision trees and rule-based classifiers (ii) Perceptron based techniques, i.e., Artificial Neural Networks and Multi-Layer Perceptron (MLP) (iii) Statistical learning algorithms, i.e., Bayesian Networks (BN), Naïve Bayes (NB) classifiers and k-Nearest Neighbor (kNN), and

finally (iv) Support Vector Machine (SVM) classifiers [3]. Design of a CDSS makes use of a single ML algorithm from mentioned groups to apply on a specific disease domain with the most possible diagnosis accuracy. Generally, if the accuracy of the algorithm is irrelevant in terms of disease diagnosis then another strategy called multiple-classifier systems (MCSs) should be preferred to improve classification performance. In this context, a MCS may be defined as the use of multiple learners to obtain better predictive performance than could be obtained from any of the single learners. In clearer terms, human nature consults several experts before making a final decision. Similarly, automated decision making applications weigh opinions of individual members of community to obtain a more accurate final solution. "Also known under various other names, such as committee of classifiers, or mixture of experts, MCSs have shown to produce favorable results compared to those of single-expert systems for a broad range of applications and under a variety of scenarios" [4]. Various MCSs are bagging, bagging, boosting, AdaBoost, stacked generalization, mixture of experts, voting based techniques with a range of combination strategies, and decision templates [4, 5]. In particular, voting based multi-classifier algorithms comprise Maximum, Median, Mean, Minimum, and Majority vote rules to fuse individual decisions of classifiers [6]. Accordingly, the aim of this study is to develop a majority vote decision system for automated identification of vertebral column pathologies from Orthopedics field. The vertebral column system is composed by a group of vertebras, invertebrate discs, nerves, muscles, medulla and joints. The two example dysfunctions of vertebral column are disc hernia (DH) and spondylolisthesis (SP) that may cause intense pain [7]. The analyzed dataset is from UCI repository and it is built by Dr. Henrique da Mota during a medical residence period in the Group of Applied Research in Orthopaedics (GARO) of the Centre Médico-Chirurgical de Réadaptation des Massues, Lyon, France. In the dataset, the two pathologies are defined in terms of six biomechanical attributes of the spino-pelvic system that correspond to the following parameters: angle of pelvic incidence, angle of pelvic tilt, lordosis angle, sacral slope, pelvic radius and grade of slipping [7]. Berthonnaud et al. discusses the correlation between six biomechanical attributes and the two Orthopedic pathologies in [8] with detail. In the literature, there are two significant studies that use this dataset [7–9]: In study [9], authors remove some of the instances from dataset with the help of an outlier analysis and they use 80% train-20% test set divisions for classification. They present three ensembles, i.e., SVM, MLP and GRNN with the average accuracies of 91%, 84.5% and 76.8% respectively. However, study [9] is in Portuguese and this made difficult the interpretation of the results precisely. In their second study, Neto et al. studied an embedded rejection option based SVM algorithm and they obtained 85.9% as their highest accuracy. More recent studies that used this dataset as machine learning problem are given in [27-30]. The studies in the literature using this dataset focus on single learners. It is confidently known that ensemble learners that constitute single learners may improve the performance of a machine learning problem to some extent. Hence the design of the experiments for identification of vertebral column pathologies were evaluated in this direction [4]. As it is widely In view of this introduction, the aim of this study is to develop a multiple-classifier algorithm to discriminate three states, i.e. normal, DH and SP, of an Orthopedic patient with an acceptable accuracy.

2. Materials and Methods

2.1 Vertebral Column Dysfunctions

Vertebral column is another term that refers to the spine or backbone, and it is the main structure of the axial skeleton of all vertebrate animals. In humans, vertebral column comprises series of vertebrae, i.e., any of the bones or segments composing the spinal column, extending from the axis bone at base of the skull to the tip of the tail. The main functions of vertebral column are: (i) permit movement of the body, (ii) enclosure and protection of the spinal cord and (iii) providing points of attachment for the ribs (bones) and muscles of the torso. Vertebral column can suffer dysfunctions that cause backaches with very different intensities. The two example pathologies of vertebral column are disc hernia and spondylolisthesis. In general, these pathological cases may originate from several traumas in the column that gradually injures the structure of the intervertebral disc [7].

Disc hernia is the result of the migration of inter-vertebral disc from its place. The other type of vertebral disease, i.e. Spondylolisthesis occurs if one of 33 vertebras from vertebral column slides [7]. Each patient of the two diseases is defined in terms of six biomechanical features and we present brief information for these attributes in the next section.

2.2 Description of Dataset

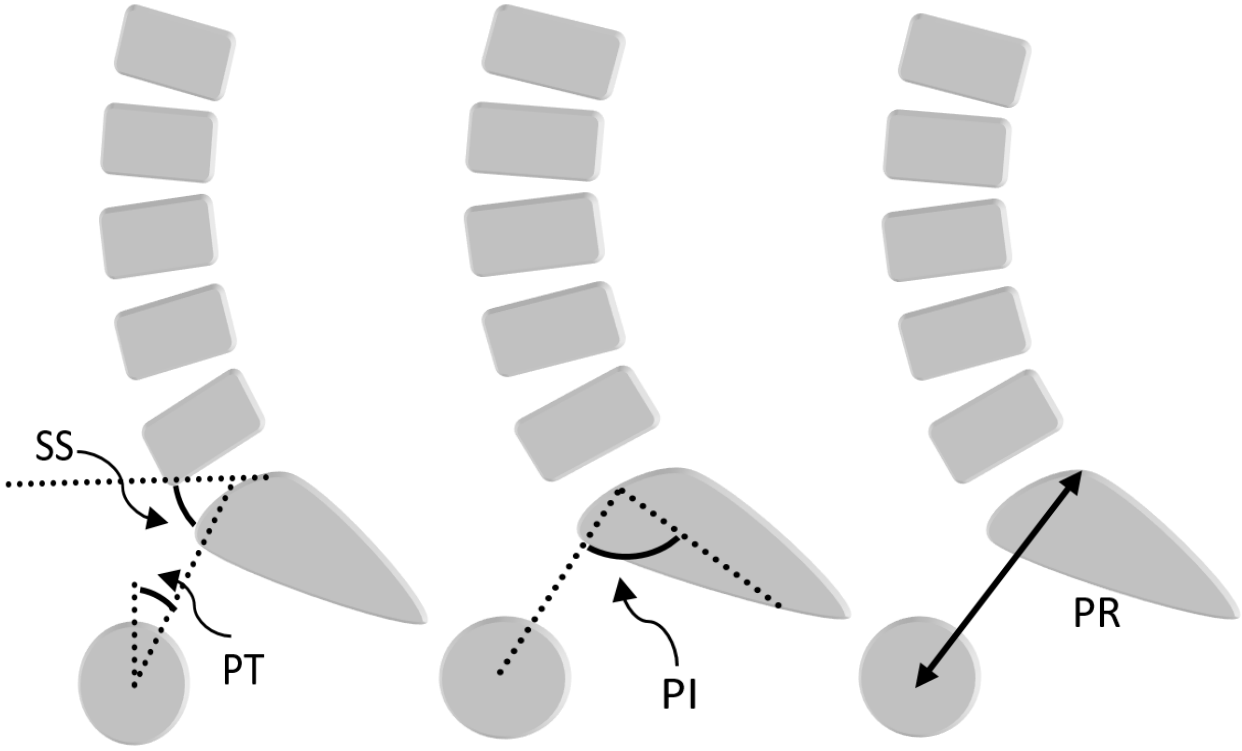
The analyzed dataset is prepared by Dr. Henrique da Mota, who collected it during a medical residence in spine surgery at the Centre M´edico-Chirurgical de R´eadaptation des Massues, placed in Lyon, France. The data is extracted from sagittal panoramic radiographies of the spine of 310 patients. The characteristic of each patient is defined with the help of six biomechanical features. The name of the features defining parameters of spino-pelvic system and the distribution of dataset is provided in Table 1. There is a remarkable correlation between these biomechanical features and the mentioned diseases, and this relation is explained in [8] with detail.

Table 1. Statistical distribution of diseases and their biomechanical characteristics.

No	Distribution of patients	Name of the features
1	100: Normal	Pelvic incidence angle (PI)
2	60: Disc hernia	Pelvic tilt angle (PT)
3	150: Spondylolisthesis	Lumbar lordosis angle (LL)
4		Sacral slope (SS)
5		Pelvic radius (PR)
		Spondylolisthesis degree (SD)

Pelvic and spinal parameters are presented in Figure 2, 3 respectively and we briefly describe these features as follows:

The sacral slope (SS) is the angle between the sacral plate and the horizontal axis. Another mechanic attribute, the pelvic tilt (PT), is defined as the angle between the line connecting the midpoint of the sacral plate to the vertical plane from the centre of femoral head. The pelvic incidence (PI) is an angle between the line perpendicular to the sacral plate and the line through the center of femoral head [10]. Pelvic radius is defined as the distance between center of femoral head to the posterior superior corner of sacral plate. On the other hand, lumbar lordosis is the angle between the superior surface of the second lumbavertebra and the inferior surface of the fifth lumbavertebra [11].



a) Sacral Slope and Pelvic Tilt. b) Pelvic Incidence. c) Pelvic Radius.

Figure 2. Sacral Slope, Pelvic Tilt (a), Pelvic Incidence(b) and Pelvic Radius(c)

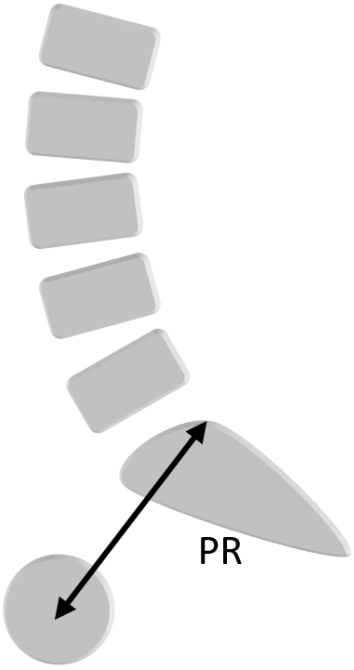


Figure 3. Lumbar Lordosis angle.

2.3 AIVCP Methodology

In this section, we provide design details of the proposed algorithm in two steps: (i) identification of the most relevant features (ii) Design of majority vote decision fusion. Furthermore, we should remind that our dataset is not high-dimensional and feature selection is not a preliminary step in our workflow. Instead, we design majority voting decision system at the first and then we apply SVM-based feature selection as the second step to improve diagnosis accuracy of the overall system.

2.3.1 Feature Selection with SVM

In a classification problem, if the number of input features is relatively larger than the number of instances then the dataset is said to be high-dimensional. There are three main difficulties in the analysis of high-dimensional datasets: (i) high computational cost, (ii) overfitting of classifier algorithm and (iii) risk of low classification accuracy [12]. In this concept, an efficient feature selection algorithm can reduce the computational cost, and increase classifier classification accuracy and efficiency [13]. In particular, feature selection is a search optimization process to identify the smallest and the most valuable features in accordance with class labels of an input dataset. For a large feature space, feature selection strategies often use greedy search rather than exhaustive. Feature-ranking methods are among widely used techniques that are used to select fixed number of top relevant features. This suggests that feature ranking may be used to design a high-accurate class predictor based on a pre-selected (ranked) subset of features.

Though, our dataset is not high-dimensional, for the sake of higher classification accuracy, we make use of a SVM based feature selection strategy while developing the proposed AIVCP system.

In our feature selection scheme, we use a two class approach, i.e., normal, abnormal, to identify the most relevant features. Therefore, for a given set of training instances $\{x_1, x_2, ...x_k, ...x_l\}$ with class labels $\{y_1, y_2, ...y_k, ...y_l\}$ for $y_k \in \{normal, abnormal\}$. The test instances $\mathbf{x}$ are classified with respect to sign of the decision function $D(\mathbf{x})$ as follows:

(i) $D(\mathbf{x}) > 0 \Rightarrow \in$ class *normal*

(ii) $D(\mathbf{x}) < 0 \Rightarrow \in$ class *abnormal*

(iii) $D(\mathbf{x}) = 0$, decision boundary

In this context, decision functions are defined as the simple weighted sums of the training instances and additional bias that are called linear discriminant functions [14]. Mathematically, this relationship is shown in equation (2.1).

$$D(\mathbf{x}) = \mathbf{w}.\mathbf{x} + b \qquad \textbf{(2.1)}$$

In equation (2.1), $\mathbf{w}$ is weight vector and b is bias. Furthermore, a dataset is *linearly separable* under the condition that a linear discriminant function is found to separate classes of data without error.

Feature ranking coefficients may be used as classifier weights, and reciprocally the weights multiplying the inputs of a classifier may also be used as feature ranking coefficients [14, 15]. In this scheme, linear discriminant functions may be trained with an algorithm such as SVM to provide feature ranking. In this algorithm, the magnitude of weights of the linear function is proportional to the weight (relevancy) of features.

Linear SVMs are suitable to be used as linear discriminant functions, and we used polynomial kernel with training inputs normalization option as feature evaluator to identify the best feature subset from vertebra column dataset. The flow of algorithm is adopted from [14] and the SVM based feature selection process is given in Figure 4.

Step 1: input $\mathbf{x} = \{x_1, x_2, ...x_k, ...x_l\}$ and
$\mathbf{y} = \{y_1, y_2, ...y_k, ...y_l\}$
Step 2: set $\mathbf{f} = \{f_1, f_2, ..., f_n\}$ initial feature vector
set $\mathbf{r} = \{\}$ ranked feature list
Step 3: until $\mathbf{f} = \{\}$ repeat steps from 4 to 8

Step 4: train SVM with $\mathbf{f}$ and $\mathbf{x}$ and obtain classifier parameters $\boldsymbol{\alpha}$
Step 5: compute weight vector $\mathbf{w} = \sum \alpha_k y_k \mathbf{x}_k$
Step 6: compute ranking criteria $c_i = (w_i)^2$ for all i
Step 7: eliminate feature with smallest ranking criterion, c_i
Step 8: update $\mathbf{r}$
Step 9: output optimum feature subset

Figure 4. SVM-based feature selection algorithm

The resultant feature subset through the application of SVM feature selection is provided in Table 2.

Table 2. Feature subset of vertebral column dataset based on SVM-feature selection

Subset of features
Sacral slope
Spondylolisthesis degree
Lumbar lordosis angle
Pelvic radius

2.3.2 Multiple Classifier Systems

Multiple classifier fusion may generate more accurate classification than each of the constituent classifiers. Fusion is often based on based combination rules like the median and average.

MCSs, particularly vote decision fusion algorithms, are developed to obtain higher predictive performances compared to each of constituent predictors. Voting based multi-classifier algorithms use miscellaneous classifiers as experts and they fuse individual decisions of these learners to obtain a more accurate final decision. The continuous output provided by each classifier is combined with the help of fusion rules such as Maximum, Median, Mean, Minimum, and Majority voting [6, 16]. From classification perspective, the output provided by a classifier for a given class is interpreted to be the posterior probability estimate for that class. In this interpretation, the outputs are normalized to add up to 1 over all classes [4]. Mathematically, " $p_j(\mathbf{x})$ is bounded between 0 and 1 computed for test objects $\mathbf{x}$ for each of the c classes. Once the set of posterior probabilities $\{p_{ij}(\mathbf{x}), i = 1, m; j = 1, c\}$ for m classifiers and c classes is computed for test object $\mathbf{x}$, they have to be combined into a new set $q_j(\mathbf{x})$ that can be used for the final classification" [6]. In general, new confidence $q_j(\mathbf{x})$ for class j depending on a combination rule i is computed with equation (2.2).

$$q_j^{'}(\mathbf{x}) = combinationRule_i(p_{ij}(\mathbf{x})) \quad \textbf{(2.2)}$$

From equation (2.2), $q_j(\mathbf{x})$ can be rewritten as in equation (2.3).

$$q_j(\mathbf{x}) = \frac{q_j^{'}(\mathbf{x})}{\sum_j q_j^{'}(\mathbf{x})} \quad \textbf{(2.3)}$$

Final classification of instance $\mathbf{x}$ is computed with equation (2.4).

$$\omega(\mathbf{x}) = \arg\max_j (q_j(\mathbf{x})) \quad \textbf{(2.4)}$$

Through equations (2.2), (2.3) and (2.4), we can define combination rules as follows: (i) maximum voting selects the predictor with the highest estimated confidence, however (ii) minimum voting strategy selects the classifier with the least objection. (iii) median, and (iv) mean voting strategies average the posterior probability estimates. In the last voting strategy, (v) maximum voting counts the votes of individual classifier and selects majority class as final decision [6].

2.3.2.1 Majority Vote Decision Fusion Systems

A majority vote decision fusion comprises of n independent classifiers, and each of these predictors produces a unique decision for an unknown pattern. The final decision for the class label of the pattern is obtained with the k number of agreement among the classifiers [17]. Moreover, the relationship between k and n is defined as follows:

i) $k = \left\{ \left(\frac{n}{2} \right) + 1 \right.$ for even n

ii) $k = \left\{ \left(\frac{n+1}{2} \right) \right.$ for odd n

In our implementation, we selected four diverse classifiers, i.e., Simple Logistics (SL), Decision Stump (DS), Naïve Bayes (NB), and Learning Vector Quantization (LVQ), to design the proposed majority vote decision fusion system. In the following section, we provide brief information about the classifiers and the classifier selection methodology while designing the proposed fusion algorithm.

2.3.3 Classifier Selection Strategy for Fusion

Design of fusion strategies from a large pool of different classifiers is not a straightforward task. It is almost impossible to define an exact design strategy that will guarantee the optimum solution to a particular problem [18]. However, it is recommended in the literature that a two-step design strategy may be helpful to obtain the optimum combination for a multiple-classifier fusion application: i) creating a limited collection of promising classifiers with diversity, ii) selection of classifiers from the collection recurrently with a search strategy. Furthermore, it should be noted that the best fusion combination is not always guarantied with the combination of the best individual classifiers [19]. Therefore, to evaluate possible fusion combinations, a search strategy is required. In our implementation, we used a grid-search strategy, i.e., an exhaustive method to obtain possible fusion combinations, on top of the 10-fold cross-validation for the evaluation of models. Since, grid-search for high number of classifiers require a serious computational load, we therefore limited the number classifiers in the collection. Our selection strategy is as follows:

i) At the first step, we wrote a java interface to WEKA suit to evaluate classifiers from diverse groups, i.e., Bayes learners, neural network classifiers, instance-based learners, rule based learners, decision trees and relatively new algorithms based on immune-colony inspired systems.

ii) We made experiments for vertebral column dataset to obtain 2 or 3 significant classifiers from each group and we obtained 10 classifiers in total. The names of the classifiers are provided in Table 3. In selection of classifiers, we used two criteria; a) prediction performance and b) the error analysis regard to predictions of the classifiers. In simpler terms, we examine the output of the classifiers for their predictions, and we choose classifiers making different predictions (either false or true) for the same instance.

iii) We applied a grid-search methodology guided by 10-fold cross-validation strategy to obtain different classifier combinations for majority voting.

iv) We obtained the best fusion combination producing the highest classification accuracy for the SL, LVQ, NB and DS.

Table 3. Classifier pool used in the selection of base classifiers

No	Classifier	No	Classifier
1	Decision Stump	6	Multi-Layer
2	iBk	7	Simple Logistics
3	Naïve Bayes	8	Clonal Selection
4	Hyper Pipes	9	Artificial Immune
5	Learning Vector	10	ZeroR

2.3.3.1 Selected Classifiers

In this section, we provide brief information for Simple Logistics, Decision Stump, Naïve Bayes, and Learning Vector Quantization:

i) Simple Logistics: Two popular supervised learning tasks are tree induction methods and linear regression models. The two approaches may be combined into model trees whose leaves are regression functions. In a similar fashion, SL algorithm is implemented with model trees having logistic regression models at the leaves. Linear logistics regression models posterior class probabilities $\Pr(G = j \mid X = x)$ for the *J* classes via linear functions of *x*. The simple form of model is given in equation (2.5).

$$\Pr(G = j \mid X = x) = \frac{e^{F_j(x)}}{\sum_{k=1}^{J} e^{F_k(x)}}, \sum_{k=1}^{J} F_k(x) = 0 \quad \textbf{(2.5)}$$

Linear regression functions, i.e., $F_j(x) = (\beta_j)^T.x$, are usually fit by obtaining maximum likelihood estimates for parameters β_j. These estimates are efficiently computed with the use of LogitBoost algorithm and hence the model in equation (2.5) is simplified as $F_j(x) = \sum_m f_{mj}(x)$. Here, f_{mj} is with the use of simple linear regression summation and this simplified algorithm is called as Simple Logistics by its implementers [20].

ii) Decision Stump: A decision stump is a decision tree with one internal node which is immediately connected to the terminal nodes. DS tree makes a prediction based on the value of just a single input feature. In the literature, these 1-level decision trees are also known as 1-rule algorithms [21]. At each given node i of the tree for a subset of training examples D_i, the goal is to select a feature such that the instances are divided into their relevant class. In other words, segmentation of D_i is accomplished with purity, i.e., all instances in a node should be in the same class. In DS, this feature selection is made to maximize information gain. More clearly, DS selects the feature that maximizes information gain in the whole dataset. Decision stumps are often used as components of classifier ensemble algorithms to obtain high accurate systems.

iii) Naïve Bayes: A NB classifier is an algorithm that assumes the presence of a particular feature of a class is unrelated to the presence of any other feature, for a given class variable. In other words, even if those features depend on each other, a NB classifier considers all of those properties to independently contribute to the class

variable. A NB classifier uses a small amount of training data to estimate the parameters, i.e., means and variances of the variables, for classification. NB classifier model can be given in equation (2.6) [22, 23].

$$p(C \mid f_1,..., f_n) = \frac{1}{Z} p(C \prod_{i}^{n} p(f_i \mid C) \quad \textbf{(2.6)}$$

In equation (2.6), $p(C \mid f_1,..., f_n)$ is conditional probability model for NB classifier over a dependent class variable C conditional on feature variables $(f_1,..., f_n)$. And in the equation, Z is a constant depending on features. NB classifier combines this model with a decision rule such as the *maximum a posteriori* rule to make classification [22, 24].

iv) Learning Vector Quantization: LVQ is an algorithm that learns classifiers from labeled data. LVQ, models the class discrimination function with the use of labeled codebook vectors and the nearest neighborhood search between the codebook and data. For classification purposes, an instance is first assigned to the closest codebook vector and then it takes class label of that codebook as its class. LVQ training is accomplished with iterative gradient update of the winner unit. The winner unit m^c is defined by equation (2.7).

$$c = \arg\min_k \left\| D_i - m^k \right\| \quad \textbf{(2.7)}$$

In equation (2.7), D_i is the instance to be classified. With the nearest neighborhood search, the update equation for a data sample $D(t)$ is given in equation (2.8).

$$m^c(t+1) := m^c(t) \pm \alpha(t)[D(t) - m^c(t)] \quad \textbf{(2.8)}$$

In equation (2.8), the sign is taken as positive for correctly classified data instances and as negative for a misclassification. In the same equation, $\alpha(t)$ is the learning rate that may take values between 0 and 1. In a classification problem, this procedure is repeated iteratively until convergence is reached [25, 26].

2.3.4 Flow of AIVCP Algorithm

In this section, we provide layout of the proposed algorithm and we present the general flow in Figure 5. However, there are a few reminders about the designed algorithm: (i) expecting a higher accuracy, we applied SVM feature selection after obtaining the best majority vote fusion decision system, (ii) In the case of equality in the votes of four classifiers, we selected the decision of SL as the correct classification label for majority voting evaluation and (iii) we evaluated performance of the whole algorithm on top of a 10-fold cross-validation scheme.

In the Figure 5, we provide two sample instances from the classifier outputs and then we present combination of these two decisions with majority voting scheme: For the first instance, the four classifiers make false predictions and hence majority vote of these decisions produces a false prediction. In the second case, though NB and DS produce false predictions for Hernia, SL and LVQ make true predictions. As we declared, in the case of the equality of votes, the algorithm chooses the vote of SL's group to identify true label of the instance. Hence, voting makes a true prediction for the second instance.

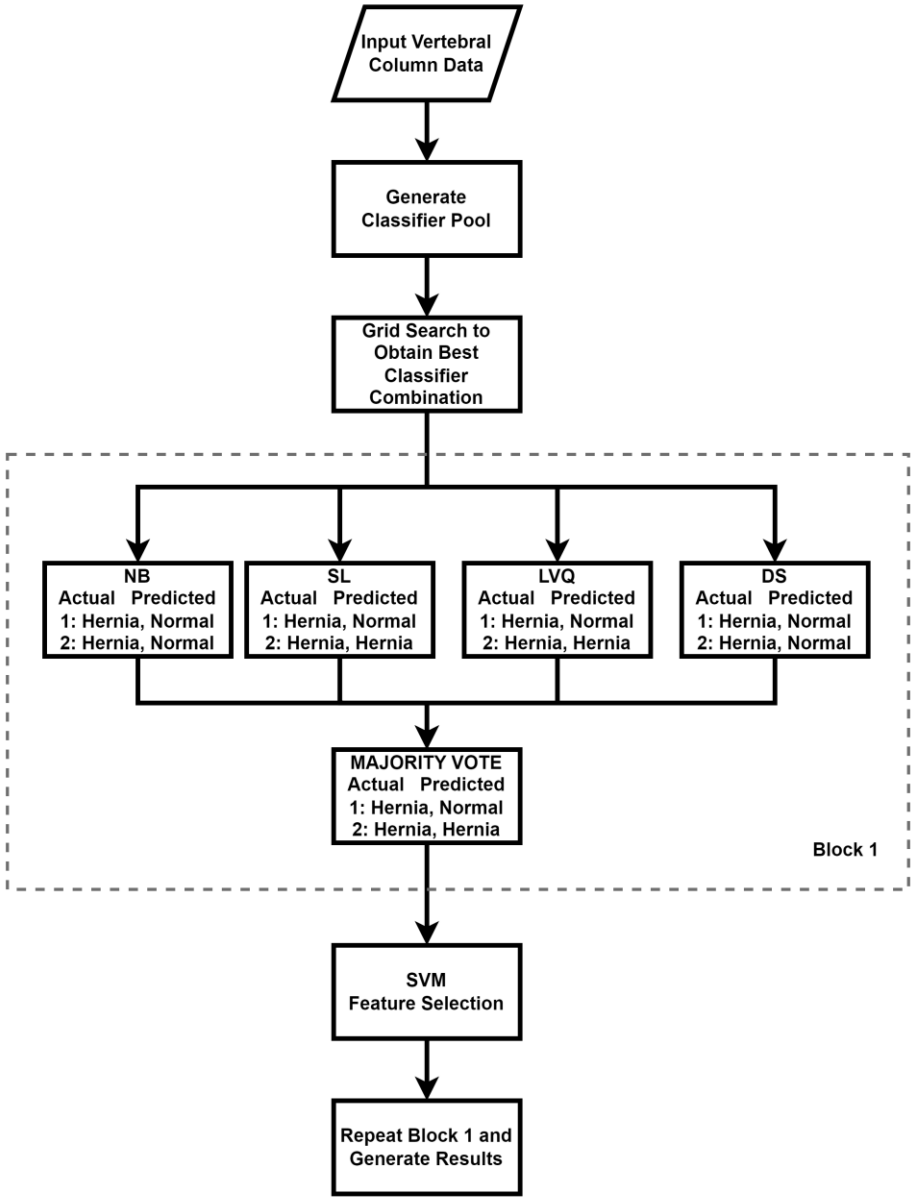


Figure 5. General flow of the AIVCP Algorithm.

3. Experiments

3.1 Statistical Evaluation Metrics

Any classification problem, produces four possible outcomes defined as *true positive* (TP), *false positive* (FP), *true negative* (TN) and *false negative* (FN). These outcomes are related to each other with *confusion matrix*. This matrix is used to derive well known performance metrics such as sensitivity, specificity, accuracy, positive prediction value, f-score, AUC and ROC curve. In our study, we use f-score (FS), Sensitivity (Sn) and Specificity (Sp) to evaluate the results of our experiment. We also make use of confusion matrices to inspect classifier prediction performances in detail. FS, Sn and Sp are defined with the following relations:

Sensitivity (Sn): The number of true positive decisions/number of actual positive cases.

Specificity (Sp): The number of true negative decisions/number of actual negative cases.

F-Score (FS): *FS* is used to measure the performance of machine learning classifiers and it can be used for balanced or imbalanced problems. The metric is defined in equation (3.1).

$$FS = 2TP/(2TP + FP + FN) \quad \textbf{(3.1)}$$

Table 4. Experimental results of AIVCP algorithm with and without feature selection

		Without Feature Selection					With SVM Feature Selection				
		NB	**SL**	**LVQ**	**DS**	**Vote**	**NB**	**SL**	**LVQ**	**DS**	**Vote**
Diseases	**F1-Score**	0.786	0.831	0.816	0.744	**0.853**	0.780	0.829	0.786	0.660	**0.883**
Hernia	Sn	0.717	0.633	**0.750**	0	0.717	0.667	0.667	0.683	0	**0.800**
Spondylolisthesis	Sn	0.973	0.960	0.947	0.953	**0.980**	**0.980**	0.960	0.947	0.953	0.973
Normal	Sn	0.690	0.860	0.780	**0.970**	0.830	0.690	0.860	0.740	**0.970**	0.860
	Sn (Avg)	0.793	0.818	0.826	0.641	**0.842**	0.779	0.829	0.79	0.641	**0.877**
Hernia	Sp	0.991	0.944	0.904	**1.000**	0.944	0.912	0.948	0.896	**1.000**	0.956
Spondylolisthesis	Sp	0.912	**0.981**	0.975	0.750	0.969	0.925	**0.981**	0.975	0.750	0.975
Normal	Sp	**0.929**	0.881	0.919	0.686	0.914	0.905	0.886	0.890	0.686	**0.929**
	Sp (Avg)	**0.944**	0.935	0.933	0.812	**0.942**	0.914	0.938	0.921	0.812	**0.954**

3.2 Experiments and Results

The results of the AIVCP algorithm with and without feature selection are given in Table 4. It is known in advance that a good medical decision support system is the one producing high f-score, specificity and sensitivity concurrently. With this point in mind, we provide Figure 6 for the results corresponding to without feature selection and we provide Figure 7 for the SVM-based feature selection.

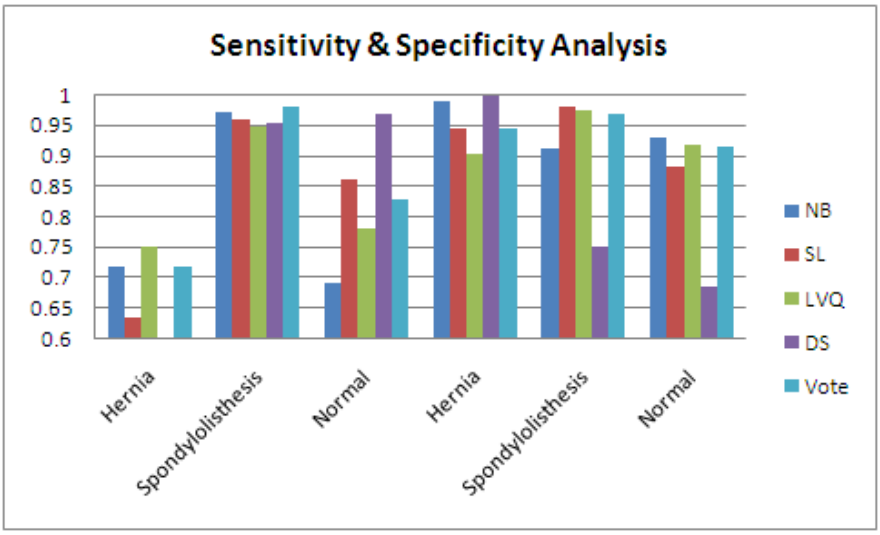


Figure 6. Sensitivity and Specificity analysis with all features used.

It is observed from Table 4 that *FS* of the majority Vote algorithm is higher than from each of the constituents of the decision system. From Table 6, for the case of Sn, the average value of voting algorithm is significantly better from the four of the classifiers. On the other hand, overall inspection of Figure 6 demonstrates that performance of classifiers, i.e., in terms of Sn and Sp, changes relatively in Hernia and Normal cases. For example, LVQ generates the highest Sn for Hernia and in contrast DS produces the worst performance. Furthermore, in Normal case, DS has the most significant Sn value of 0.970 among the classifiers. Additionally, for Sp values, it is observed from the right part of the Figure 6 that the most significant value for Hernia is produced by DS. For the Normal case, whereas DS is the worst predictor, the remaining three classifiers generate acceptable results. Interestingly, though overall classification performance of DS is the lowest among the classifiers, it contributes positively to the performance of the vote algorithm. We will analyze contribution of DS with more detail with the help of confusion matrices and some additional experiments later.

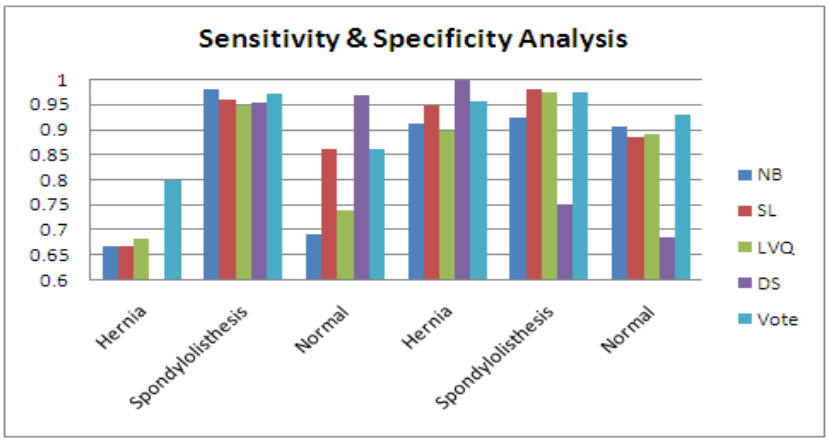


Figure 7. Sensitivity and Specificity analysis for SVM-based feature selection.

From Table 4, it is obviously inspected that vote algorithm generates the highest f-score value of 0.883. For Sn-Sp evaluation, results of Figure 7 are observed to be similar to those of Figure 6. On the other hand, we may observe from Figure 7 that for Hernia-Normal pair, vote algorithm has significantly better Sn-Sp values compared to values of Figure 6. It is furthermore observed from Figure 7 that overall performance of vote is increased significantly.

After this point, we provide some other experimental results to present contributions of classifiers to the community decision. In simpler terms, we observed all classifiers, except DS, more or less produce similar results. Since, overall performance of DS is significantly lower than the three classifiers in terms of f-score values, it is necessary to question the effect of classifiers on the final decision of vote algorithm.

Table 5. Comparison of experimental results of Vote algorithm with DS and without DS.

	Vote without DS			Vote with DS		
Diseases	**Sn**	**Sp**	**FS**	**Sn**	**Sp**	**FS**
Hernia	0.717	0.928	-	0.800	0.956	-
Spondylolisthesis	0.973	0.981	-	0.973	0.975	-
Normal	0.800	0.908	-	0.860	0.929	-
Average	0.830	0.939	0.867	0.877	0.954	0.879

Table 5 evidently shows that, though single overall classification performance of DS algorithm is relatively small compared to LVQ, NB and SL, it has a considerable contribution to the performance evaluation metrics of Vote algorithm. All of the performance metrics are improved obviously. The only exception is Spondylolisthesis whose metrics have a minor decrease. The reason behind the contribution of DS algorithm to the entire decision fusion may be explained in terms of confusion matrices given in Table 6. For the sake of convenience, we make use of abbreviations of three cases as *H* for Hernia, *S* for Spondylolisthesis and *N* for Normal. It is repeated that the original numbers of instances are 60, 150 and 100 for three class labels respectively. For an intense evaluation, we provide confusion matrices of voting algorithm for two cases: (i) voting without DS (Vote1) and (ii) voting with DS (Vote2).

Table 6. Confusion matrices of classifiers.

Confusion Matrices

NB		Predicted H	S	N
Expected	H	40	3	17
	S	0	147	3
	N	22	9	69

SL		Predicted H	S	N
Expected	H	40	1	19
	S	1	144	5
	N	12	2	86

LVQ		Predicted H	S	N
Expected	H	41	1	18
	S	3	142	5
	N	23	3	74

DS		Predicted H	S	N
Expected	H	0	1	59
	S	0	143	7
	N	0	3	97

VOTE1 (-DS)		Predicted H	S	N
Expected	H	43	1	16
	S	0	146	4
	N	18	2	80

VOTE2 (+DS)		Predicted H	S	N
Expected	H	48	1	11
	S	0	146	4
	N	11	3	86

As confusion matrices of classifiers are examined in Table 6, the following results can be drawn:
The relative true prediction performances of four base classifiers for class Spondylolisthesis is similar to each other. There are 150 Spondylolisthesis cases in the dataset and the classifiers make true predictions that change from 143 to 147. On the other hand, there are some remarkable results for Hernia and Normal classes. There are 60 Hernia instances in the dataset and classifiers make predictions through 0 to 41. It is interesting to observe that DS in this case has no true prediction at all. Other than DS, base classifiers generate similar true predictions between 40 and 41. In spite of poor prediction performance of DS, Vote1 and Vote2 algorithms combine remaining predictions and they obtain 43 and 48 true predictions respectively. For the

Normal class, it is observed that the most significant true prediction performance, i.e., 97 predictions out of 100, belong to DS algorithm. The second remarkable result for this class is that of SL algorithm with the 86 true predictions. When the two voting algorithms are examined from confusion matrices, it is observed that they have 80 and 86 true predictions for Normal class. In this case, SL and DS contribute to the Vote2 algorithm and they lead to produce an acceptable true prediction. In this context, Vote2 benefits from the fusion of the decisions of DS and SL. From confusion matrices, it is seen that the most successful prediction performances among six classifiers belong to SL and Vote2. Both SL and Vote2 have almost the same prediction ratios in Spondylolisthesis and Normal cases. However, it is seen that SL and Vote2 algorithms have 40 and 48 true predictions in Hernia class. This is the main source of performance increase of Vote2 algorithm. Fusion of predictions of SL, LVQ and NB for different data instances increases overall performance of Vote2 algorithm.

We provide Figure 8 and 9 to show the contribution of each base classifier to final decision of Vote algorithm. The procedure to obtain the two figures is as follows:

i)We first generated the predictions of all classifiers (including Vote algorithm) for all instances of data.

ii)We excluded predictions for Spondylolisthesis class and we obtained the true and false predictions of all classifiers for Hernia and Normal cases. And then, we divided this outcome into two parts regard to Hernia and Normal cases.

iii) For each of the two divisions of step (ii), we maintained the outcomes that contain at least one false prediction and we eliminated the instances that are classified correctly by all classifiers. While realizing this step, we kept the order of the instances unchanged to observe coincidence of predictions.

iv) We obtained false predictions of all classifiers for Hernia and Normal classes as two separate figures.

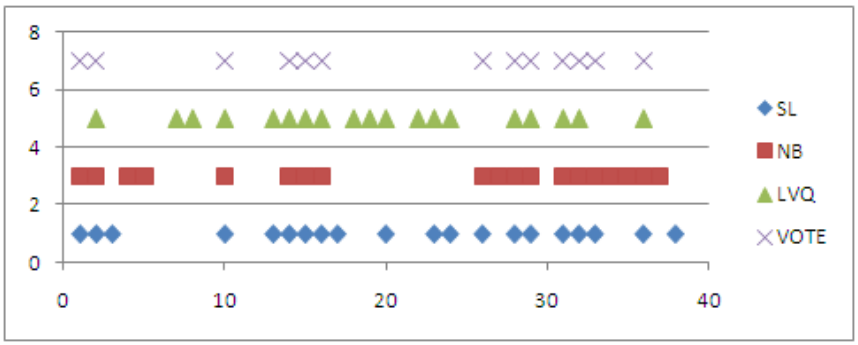


Figure 8. False predictions of base classifiers and Vote algorithm for Hernia class.

Since it has shown in Table 6 that DS produces misclassifications for all Hernia instances, we provide the predictions of SL, LVQ, NB and Vote algorithm in Figure 8 and we disregard DS predictions. As the figure is examined, DS (with having no true prediction), NB, LVQ and SL make misclassifications much more than Vote algorithm. However, it is observed from the figure that, the base predictors make misclassification in different portions (instances) of dataset. Therefore, the fusion of those decisions with majority vote produces less false predictions compared to the each of the base predictors.

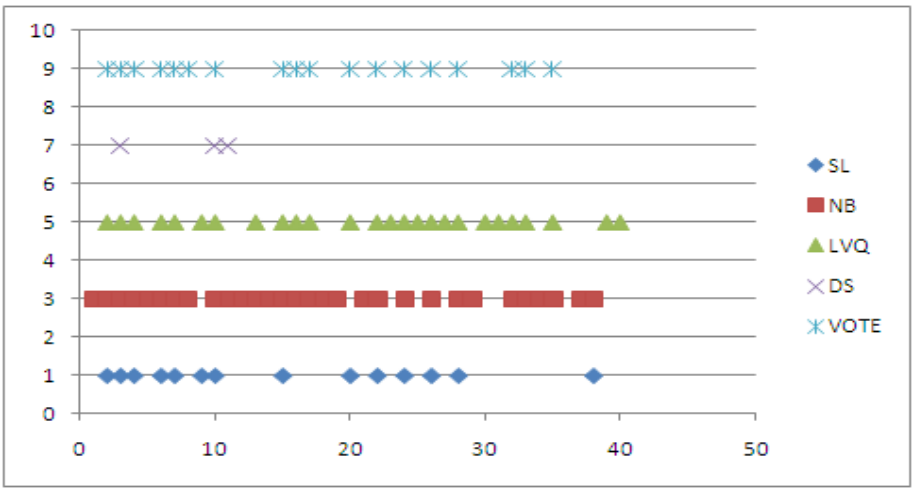


Figure 9. False predictions of base classifiers and Vote algorithm for Normal class.

Figure 9 demonstrates false predictions of base classifiers and their majority vote based fusion counterpart. In this case, though LVQ and NB make more false predictions compared to those of SL and DS, the majority vote combination of the whole predictions decreases chance of false prediction of Vote algorithm. Hence, the misclassification rate of Vote algorithm benefit from DS and SL prediction rate positively.

As a last experiment for SL, LVQ, DS and NB base classifiers, we provide average f-score values for different combination rules of voting algorithms. The respective f-score values for rules, i.e., Average of probabilities, Product of probabilities, Minimum probability, Maximum probability, and Majority voting, are given in respective order as 0.823, 0.513, 0.819, and 0.883.

4. Conclusion

Vertebral column pathologies are diagnosed with the use of a majority vote decision fusion system. The diagnosis of these pathologies is first analyzed with the use of single predictors, NB, SL, LVQ, and DS. The prediction performances of the classifiers with SVM-based feature selection are 0.780, 0.829, 0.786, 0.660 respectively. In order to obtain a higher diagnosis performance, we designed a decision fusion system and therefore we obtained an acceptable f-score of 0.883 compared with the results in the literature.

The proposed AIVCP algorithm is evaluated in terms of fscore, sensitivity, specificity, confusion matrices and prediction error ratios. Through these steps the following significant results are obtained:

1) The single base predictor performances may be combined to obtain higher prediction f-score values.

2) High performances of the individual classifiers do not guarantee to generate high prediction f-score. Instead, diverse classifiers and different combinations should be evaluated. Different combinations may be generated with the use of limited exhaustive search or with a heuristics algorithm such as Genetic search.

3) In the selection of algorithms, an error analysis depending on the predictions of base classifiers may contribute the success of the vote algorithm.

4) Analysis of confusion matrices and sensitivity-specificity pairs may also contribute positively to the overall performance of the vote algorithm.

5) Though, we obtained majority voting as the best combination rule in this specific problem, other rules may also be searched for a better prediction f-score.
The relative success of the proposed algorithm, decision fusion approach in particular, may be used to increase single classifier based CDSSs with different combination strategies.

As a future direction, we intend to extend the proposed algorithm with the use of heuristics search strategies to obtain higher prediction performances for different medical decision domains.

Author's Contributions

Dr. Akın Özçift designed the algorithm. Dr. Mehmet Bozuyla evaluated experiments and their results. Both authors contributed to the preparation of the whole article.

Ethics

There are no ethical issues after the publication of this manuscript.

References

[1]. Sim I, Gorman P, Greenes RA et al. 2001. Clinical decision support systems for the practice of evidence-based medicine. *Journal of the American Medical Informatics Association*; 8(6): 527–534. doi: 10.1136/JAMIA.2001.0080527/2/JAMIA0080527.F01.JPEG.

[2]. Shahmoradi L, Safdari R, Ahmadi H, Zahmatkeshan M. 2021. Clinical decision support systems-based interventions to improve medication outcomes: A systematic literature review on features and effects. *Medical Journal of the Islamic Republic of Iran*; 3527. doi: 10.47176/MJIRI.35.27.

[3]. Shaikh F, Dehmeshki J, Bisdas S et al. 2021. Artificial Intelligence-Based Clinical Decision Support Systems Using Advanced Medical Imaging and Radiomics. *Current Problems in Diagnostic Radiology*; 50(2): 262–267. doi: 10.1067/J.CPRADIOL.2020.05.006.

[4]. Polikar R. 2006. Ensemble based systems in decision making. *Circuits and Systems Magazine*; 6(3): 21–44. doi: 10.1109/MCAS.2006.1688199.

[5]. Hanson CC, Brabyn L, Gurung SB. 2022. Diversity-accuracy assessment of multiple classifier systems for the land cover classification of the Khumbu region in the Himalayas. *Journal of Mountain Science 2022 19:2*; 19(2): 365–387. doi: 10.1007/S11629-021-7130-7.

[6]. Duin RPW, Tax DMJ. 2000. Experiments with Classifier Combining Rules. In: Int. Work. Mult. Classif. Syst. Springer-Verlag. pp 16–29.

[7]. Neto ARDR, Sousa R, Barreto GDA, Cardoso JS. 2011. Diagnostic of Pathology on the Vertebral Column with Embedded Reject Option. In: Iber. Conf. Pattern Recognit. Image Anal. Las Palmas de Gran Canaria, Spain, Springer, Berlin, Heidelberg. pp 588–595.

[8]. Berthonnaud E, Dimnet J, Roussouly P, Labelle H. 2005. Analysis of the sagittal balance of the spine and pelvis using shape and orientation parameters. *Journal of Spinal Disorders and Techniques*; 18(1): 40–47. doi: 10.1097/01.BSD.0000117542.88865.77.

[9]. Neto ARR, Barreto GA. 2009. On the application of ensembles of classifiers to the diagnosis of pathologies of the vertebral column: A comparative analysis. *Latin America Transactions*; 7(4): 487–496. doi: 10.1109/TLA.2009.5349049.

[10]. Baker JF, Joseph Baker CF, Y W O R D S child KE. 2021. Computed tomography study of the relationship between pelvic incidence and bony contribution to lumbar lordosis in children. *Clinical*

Anatomy; 34(6): 934–940. doi: 10.1002/CA.23756.

[11]. Açar G, Çiçekcibaşı AE, Koplay M, Seher N. 2021. Surface anatomy and lumbar lordosis angle. *Anatomical Science International*; 96(3): 400–410. doi: 10.1007/S12565-021-00602-1/FIGURES/6.

[12]. Jain AK, Duin RPW, Mao J. 2000. Statistical pattern recognition: A review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*; 22(1): 4–37. doi: 10.1109/34.824819.

[13]. Tu C-J, Chuang L-Y, Chang J-Y, Yang C-H. 2006. Feature selection using PSO-SVM. IAENG Int. J. Comput. Sci. 33

[14]. Guyon I, Weston J, Barnhill S, Vapnik V. 2002. Gene Selection for Cancer Classification using Support Vector Machines. *Machine Learning*; 46(1): 389–422. doi: 10.1023/A:1012487302797.

[15]. Guyon I, Elisseeff A. 2003. An Introduction to Variable and Feature Selection. *Journal of Machine Learning Research*; 31157–1182.

[16]. Rwigema J, Mfitumukiza J, Tae-Yong K. 2021. A hybrid approach of neural networks for age and gender classification through decision fusion. *Biomedical Signal Processing and Control*; 66102459. doi: 10.1016/J.BSPC.2021.102459.

[17]. Rahman A, Fairhurst M. 2000. Decision combination of multiple classifiers for pattern classification: Hybridisation of majority voting and divide and conquer techniques. In: Appl. Comput. Vis. Fifth IEEE Workshop. pp 58–63.

[18]. Wang H, Liang T, Cheng Y. 2021. Evolution and quality analysis algorithm of consumer online reviews based on data fusion and multiobjective optimization. J Sensors. doi: 10.1155/2021/6252425

[19]. Ruta D, Gabrys B. 2005. Classifier selection for majority voting. *Information Fusion*; 6(1): 63–81. doi: 10.1016/J.INFFUS.2004.04.008.

[20]. Landwehr N, Hall M, Frank E. 2005. Logistic Model Trees. *Machine Learning*; 59(1): 161–205. doi: 10.1007/S10994-005-0466-3.

[21]. Holte RC. 1993. Very Simple Classification Rules Perform Well on Most Commonly Used Datasets. *Machine Learning*; 11(1): 63–90. doi: 10.1023/A:1022631118932.

[22]. Bhargavi P, Jyothi S. 2009. Applying Naive Bayes Data Mining Technique for Classification of Agricultural Land Soils. *IJCSNS International Journal of Computer Science and Network Security*; 9(8): 117–122.

[23]. Wickramasinghe I, Kalutarage H. 2021. Naive Bayes: applications, variations and vulnerabilities: a review of literature with code snippets for implementation. *Soft Computing*; 25(3): 2277–2293. doi: 10.1007/S00500-020-05297-6/FIGURES/2.

[24]. Domingos P, Pazzani M. 1997. On the Optimality of the Simple Bayesian Classifier under Zero-One Loss. *Machine Learning*; 29(2): 103–130. doi: 10.1023/A:1007413511361.

[25]. Kohonen T. 2001. Self-Organizing Maps. doi: 10.1007/978-3-642-56927-2

[26]. Hollmén J, Tresp V, Simula O. 2000. A learning vector quantization algorithm for probabilistic models. In Tampere, Finland, IEEE.

[27]. Reshi, A. A., Ashraf, I., Rustam, F., Shahzad, H. F., Mehmood, A., & Choi, G. S. 2021. Diagnosis of vertebral column pathologies using concatenated resampling with machine learning algorithms. *PeerJ Computer Science*, *7*, e547.

[28]. Cruz, A. D., Santhosini, P., Santhini, P., & Shirly, S. 2022. Comparative Study and Detection of Spinal Deformities using Supervised Machine Learning Algorithms. In *2022 International Conference on Computing, Communication, Security and Intelligent Systems (IC3SIS)* (pp. 1-6). IEEE.

[29]. Handayani, I. 2019. Application of K-nearest neighbor algorithm on classification of disk hernia and spondylolisthesis in vertebral column. *Indonesian Journal of Information Systems*, *2*(1), 57-66.

[30]. Riveros, N. A. M., Espitia, B. A. C., & Pico, L. E. A. 2019. Comparison between K-means and self-organizing maps algorithms used for diagnosis spinal column patients. *Informatics in Medicine Unlocked*, *16*, 100206