International Journal of Data Science and Analytics (2024) 18:405–421
https://doi.org/10.1007/s41060-023-00436-2

REGULAR PAPER

Building the classification model based on the genetic algorithm and the improved Bayesian method

Dinh Pham-Toan[1] · Tai Vo-Van[2]

Received: 8 April 2023 / Accepted: 24 July 2023 / Published online: 7 August 2023

Abstract
This study presents a classification model that incorporates significant enhancements based on the Bayesian method and genetic algorithm (BGA). Firstly, the prior probabilities in each iteration are determined using the ratio of the number of elements in each group, obtained through clustering techniques, to the total number of elements in the training set. Secondly, an automatic selection process optimizes the training set to minimize classification errors. Finally, the traditional genetic algorithm operators are improved by utilizing the Bayes error as the objective function. These improvements combine to create an effective classification model. Additionally, the BGA demonstrates effective performance on real data using the established MATLAB procedure. A numerical example illustrates the superiority of the proposed algorithm compared to existing methods. The study also applies the BGA for image classification using the Gabor filter, which extracts essential image features. The proposed model outperforms popular methods in classifying various numerical and image datasets. These applications highlight the potential of this study in real-world scenarios.

Keywords Bayes error · Genetic algorithm · Intelligent classification · Supervised learning

1 Introduction

Classification is the process of assigning an element to a group in a way that optimizes performance by minimizing error. It is an important problem in statistics and supervised learning. This problem arises in various practical applications across fields such as economics, medicine, and technology. Currently, there are several classification methods available. In the field of statistics, three main methods are prominent: Fisher method [10], logistic regression method [26], and the Bayesian method [41]. However, these methods have certain disadvantages as they rely on assumptions that are challenging to satisfy in real-world scenarios. For example, the Fisher method can only be applied when all populations have the same covariance matrix. Despite some improvements, this method still struggles to overcome its inherent weaknesses. The logistic regression method is effective only when populations are relatively separate and the data exhibit linearity [40]. On the other hand, the naive Bayes method assumes independence among all features. Since this assumption rarely holds in real-life data, naive Bayes often fails to provide satisfactory results [26].

There have been numerous models proposed for classification in machine learning and deep learning. Linear discriminant analysis (LDA) and quadratic discriminant analysis (QDA) methods are extensions of the Fisher method, as they do not require the same covariance matrix for all populations. However, these methods often underperform in real-world scenarios [7]. Support vector machine (SVM) is a popular method widely used today [12, 37]. Nevertheless, SVM requires large datasets and proper selection of the training set [12, 36]. In addition to the aforementioned methods, many other machine learning and deep learning classification approaches have been proposed and applied, including XGBoost [3], random tree [8], Bagged classifier [8], Adaptive Boosting [8], k-nearest neighbors (k-NN) [16], and convolutional neural network (CNN) [25]. In the con-

Dinh Pham-Toan and Tai Vo-Van have contributed equally to this work

✉ Tai Vo-Van
vvtai@ctu.edu.vn

Dinh Pham-Toan
dinh.pt@vlu.edu.vn

[1] Faculty of Mechanical - Electrical and Computer Engineering, School of Technology, Van Lang University, Ho Chi Minh City, Vietnam

[2] College of Natural Science, Can Tho University, Can Tho City, Vietnam

text of text classification, introduced in 2017 by Vaswani et al. [38], the Transformer is a deep learning model that utilizes the Transformer architecture to classify text into different classes. The Transformer model is renowned for its ability to capture contextual relationships and dependencies in a text sequence effectively, making it a powerful tool for text classification tasks. This model is of interest to many scientists in language processing [46]. However, it has not been effectively applied in classification for numerical and image data.

There are numerous methods available for classifying image data, and recent publications have shown promising results in this field. Mehrdad et al. [20] proposed a novel PSO-based multi-objective feature selection method. This approach involved three main phases: graph representation modeling of the original features, calculation of feature centralities, and improvement of the PSO-based search process for final feature selection. The results on five medical datasets demonstrated that this method outperformed previous approaches in terms of both efficiency and effectiveness. Rostami et al. [31] conducted a comparative analysis of various feature selection methods, categorizing them based on state-of-the-art swarm intelligence. Their approach identified a subset of features with the lowest inner similarity and highest relevancy to the target class, reducing data dimensionality by eliminating irrelevant, redundant, or noisy data. Saeid et al. [32] proposed a graph-theoretic-based gene selection method for cancer diagnosis, using several datasets with different properties to demonstrate the efficacy of their model on Colon, Leukemia, SRBCT, Prostate Tumor, and Lung Cancer. While machine learning and deep learning classification methods offer many advantages, they also share some common disadvantages. Firstly, they require a large amount of training data and considerable time for training. Secondly, the performance results depend on the selection of the training set, typically chosen by the user. Finally, the classification results may lack stability.

Bayesian classification is an important method that has attracted the interest of statisticians due to its ability to classify multiple populations without strict data conditions [40, 41]. This approach offers several advantages over other classification methods. Recent research in Bayesian classification includes the work of Pham-Gia et al. [28], who proposed a classification principle using the weighted maximum function of probability density functions (pdfs) and calculated the Bayes error. While this was an important theoretical contribution, it had limitations in practical applications. The research did not address the challenges of determining the prior probability and pdf, which are crucial factors in determining the effectiveness of Bayesian classification. Additionally, this research did not consider classification for images. To overcome the limitations of Pham-Gia et al. [28] and Tai [39], Vovan [41] proposed using the L^1-distance for element classification and Bayes error calculation. This approach considered different distances and built upon the results of Pham-Gia et al. [28]. The authors also addressed the computational challenges and demonstrated superior performance compared to previous methods through numerical examples. However, it is worth noting that this approach may still yield large errors in certain cases.

In Bayesian methods, determining the prior probability is crucial. Currently, commonly used approaches for determining the prior probabilities include the uniform distribution, sample rate, and the Laplace method [28]. Nguyentrang and Vovan [27] proposed a method for determining the prior probability based on a fuzzy clustering algorithm. However, this method solely focused on improving the prior probability and did not address other crucial classification problems, leading to suboptimal results in many cases. In a later work by Vovan et al. [41], significant contributions were made to the theory of Bayesian classification. However, this work only considered the case of two populations, limiting its applicability to scenarios with multiple populations.

The genetic algorithm (GA) is an optimization method that aims to find an optimal solution based on a given criterion. It is a significant development in multi-dimensional statistics. GA is inspired by the natural selection process and involves encoding, mutation, and selection steps to search for the best solution to a given problem. Holland [14] first proposed the genetic algorithm, and since then, it has found widespread applications in various fields [14] and has since been widely applied in many fields [1, 2, 5, 18, 42, 43].

Important studies in this area can be summarized as follows. Bidi and Elberrichi [5] introduced a GA-based method for feature selection in image classification. The study aimed to determine the optimal subset of features for classifier performance and identify the smallest feature subset that achieved higher classification accuracy. However, this research was conducted manually and did not yield satisfactory results for image data. Hemanth and Anitha [13] proposed a modified GA algorithm for MRI brain images. The authors used three different GA approaches to select features for their proposed algorithm, which was based on a neural network. However, this research did not address the optimization problem of errors in the process, and the training and testing sets were subjectively chosen. Hu and Cui [15] proposed a fractional differential mask operator for describing and handling highly self-similar digital medical images. They utilized principal component analysis to extract the main features from these images and employed a support vector machine algorithm for image recognition. However, this model exhibited high errors and had a long processing time. Additionally, the research did not evaluate the results using dedicated training and test sets. In conclusion, in our opinion, the genetic algorithm has not been fully explored in the context of Bayesian methods.

From the above analysis, we observe that the utilization of genetic algorithms in the Bayesian method for data classification, especially in image applications, is still limited. In this study, we aim to enhance the essential steps of the Bayesian method and integrate them with the genetic algorithm to reduce classification error. We specifically address the following issues:

(i) Automatic in building the training set by employing an improved genetic algorithm with enhanced operators. This algorithm can efficiently select the appropriate training set for each dataset and the elements to be classified.

(ii) Optimize the Bayes error by treating it as the objective function of the proposed genetic algorithm. The algorithm will terminate when the error reaches the smallest possible value.

(iii) Determine the suitable prior probability by combining traditional methods with a cluster analysis algorithm. The prior probability is computed by dividing the number of elements in each group with the total number of elements in the training set, where the clustering techniques are used to identify the elements in each group.

In addition, our proposed algorithm offers the capability to classify multiple elements simultaneously, unlike existing algorithms that can only handle one element at a time. Another significant contribution of our research is the development of a method for classifying image data, which remains a challenging problem in numerous real-world applications. We utilize the Gabor filter to extract image features and apply principal component analysis to reduce the data's dimensionality while maintaining its classification accuracy. Additionally, we have implemented an efficient program in MATLAB to realize our proposed model. Our results demonstrate that our model outperforms existing algorithms when applied to complex image datasets. Furthermore, our implementation holds potential for practical applications in various fields.

The remainder of this paper is structured as follows. Section 2 introduces the principles of Bayesian classification, Bayes error, and some upper and lower bounds of the error. Section 3 presents the Bayesian method based on the genetic algorithm and discusses related issues. In Sect. 4, we provide a numerical example that illustrates the step-by-step implementation of our proposed model. Section 5 details the method for feature extraction using the Gabor filter. Finally, the conclusion is presented in the last section.

2 Bayesian method

2.1 Classification principle and Bayes error

Let $q_i \in (0,1), q_1 + q_2 + \cdots + q_n = 1$ and $f_i(x)$ be the prior probability and probability density function (pdf) of the population $i, i = 1, 2, \ldots, k$, respectively.

2.1.1 The principle for classification

According to Pham-Gia et al. [28], an element x_0 is assigned to w_i if

$$g_i(x) = g_{\max}(x_0), \tag{1}$$

where $g_i(x) = q_i f_i(x)$ and $g_{\max}(x) = \max\{q_1 f_1(x), q_2 f_2(x), \ldots, q_k f_k(x)\}$.

Bayes error is given by (2):

$$Be = \sum_{i=1}^{k} \int_{R^n \setminus R_i^n} q_i f_i \mathrm{d}x = 1 - \sum_{i=1}^{k} \int_{R_i^n} q_i f_i(x) \mathrm{d}x, \tag{2}$$

where $R_i^n = \{x | q_i f_i(x) > q_j f_j(x), \forall i \neq j, i, j = 1, 2, \ldots, k\}$, $(q) = (q_1, q_2, \ldots, q_k)$.

From (2), we can prove the following result:

$$Be = 1 - \int_{R^n} g_{\max}(x) \mathrm{d}x. \tag{3}$$

2.1.2 The relationship of Bayes error and other measures

Theorem 1 *Let $f_i(x), i = 1, 2, \ldots, k, k \geq 3$ be k pdfs defined on $R^n, n \geq 1, q_i \in (0; 1)$. We have the relationships of Bayes error with other measures as follows:*

(i)

$$\{(k-1) - \sum_i \sum_j \|g_i, g_j\|_1\}/k \leq Be \leq 1 - (1/2) \max_{i<j} \{\|g_i, g_j\|_1\} - \min_i\{q_i\}, \tag{4}$$

(ii)

$$0 \leq Be \leq \max_i\{q_i\}, \tag{5}$$

where $g_i(x) = q_i f_i(x)$, $\|g_i, g_j\|_1 = \int_{R^n} |g_i(x) - g_j(x)| \mathrm{d}x$.

Proof (i) We have

$$\int_{R^n} \max\{g_1(x), g_2(x), \ldots, g_k(x)\} \mathrm{d}x \geq \max_{i<j} \int_{R^n} \max\{g_i(x), g_j(x)\} \mathrm{d}x.$$

On the other hand,

$$\begin{aligned}
&\max_{i<j}\left\{\int_{R^n}\max\{g_i(x),g_j(x)\}\mathrm{d}x\right\}\\
&=\max_{i<j}\left\{\frac{1}{2}\left\|g_i,g_j\right\|_1+\frac{1}{2}\left(q_i+q_j\right)\right\}\\
&\geq\max_{i<j}\left\{\frac{1}{2}\left\|g_i,g_j\right\|_1\right\}+\min_{i<j}\left\{\frac{1}{2}\left(q_i+q_j\right)\right\}\\
&\geq\max_{i<j}\left\{\frac{1}{2}\left\|g_i,g_j\right\|_1\right\}+\min_{i<j}\left\{(q_1,q_2,\ldots,q_k)\right\}.
\end{aligned}$$

Hence,

$$\begin{aligned}
\int_{R^n}g_{\max}(x)\mathrm{d}x&\geq\frac{1}{2}\max_{i<j}\left\{\left\|g_i,g_j\right\|_1\right\}\\
&+\min_{i<j}\left\{(q_1,q_2,\ldots,q_k)\right\}.
\end{aligned}\tag{6}$$

We also have

$$\begin{aligned}
\sum_{i<}\sum_{j}|g_i-g_j|&\geq\sum_{j=1}^{k}\left[\max\left\{g_1,g_2,\cdots,g_k\right\}-g_j\right]\\
&=k\left[\max\left\{g_1,g_2,\ldots,g_k\right\}\right]-\sum_{j=1}^{k}g_j.
\end{aligned}$$

Therefore,

$$\max\left\{g_1,g_2,\ldots,g_k\right\}\leq\frac{1}{k}\sum_{i<}\sum_{j}|g_i-g_j|+\frac{1}{k}\sum_{j=1}^{k}g_j.$$

Sine $\int_{R^n}g_i(x)\mathrm{d}x=q_i$ and $\sum_{i=1}^{k}q_i=1$, the inequality (5) becomes:

$$\int_{R^n}g_{\max}(x)\mathrm{d}x\leq\frac{1}{k}\sum_{i<j}\left\|g_i,g_j\right\|_1+\frac{1}{k}.\tag{7}$$

Replacing $\int_{R^n}g_{\max}(x)=1-Be$ to (6) and (7), we obtain (4).

(ii) We have

$$\begin{aligned}
q_if_i(x)&\leq\max\left\{q_1f_1(x),q_2f_2(x),\ldots,q_kf_k(x)\right\}\\
&\leq\sum_{i=1}^{k}q_if_i(x),\forall i=1,\ldots,k.
\end{aligned}\tag{8}$$

Integrate both sides of (8), we obtain:

$$q_i\leq\int_{R^n}g_{\max}(x)\mathrm{d}x\leq1.$$

Because the above inequality is true $\forall i=1,\ldots,k$,

$$\max\{q_i\}\leq\int_{R^n}g_{\max}(x)\mathrm{d}x\leq1.\tag{9}$$

Replacing $\int_{R^n}g_{\max}(x)=1-Be$ for (9), we have (5).

□

2.1.3 Empirical error

Set

$$M=\{m_1,m_2,\ldots,m_n\},L=\{l_1,l_2,\ldots,l_n\},1<n\leq N,$$

where

M is the set of labels correctly classified,
L is the set of final classification results,
N is the total number of elements in the dataset.
Then, empirical error is defined as follows:

$$E_{\mathrm{Ner}}=\frac{1}{N}\sum_{i=1}^{N}R\left(m_i-l_i\right),\tag{10}$$

with

$$R(x)=\begin{cases}0 & \text{if } x=0,\\ 1 & \text{otherwise.}\end{cases}$$

2.2 Estimate the probability density function

In fact, the data used to perform the classification problem consist of discrete elements. As a result, to apply the Bayesian method for classification, the first step is to estimate the pdf. There are many parametric as well as nonparametric methods available to estimate the pdf, but currently, the most popular method is using the kernel function approach [29, 39–41]. Therefore, we also use this method for all the experiments in this study. The n-dimensional pdf is estimated as follows:

$$f(x)=\frac{1}{Nh_1h_2\cdots h_n}\sum_{i=1}^{N}\prod_{j=1}^{n}K_j\left(\frac{x_i-x_{ij}}{h_j}\right),\tag{11}$$

where h_j is the smoothing parameter for the jth variable, $K_j(.)$ is the kernel function of the jth variable, x_i is the ith dimension, x_{ij} is the ith number of the jth variable, and N is the number of elements in dataset.

There are several kernel functions available, such as Triangles, Rectangles, and Bi-weight. In this study, we choose the Gaussian kernel function. Moreover, there have been numerous studies to select the smoothing parameter, but none of the methods have been considered optimal. In this article,

we use the smoothing parameter selection method proposed by Scott [33], which is a popular method today.

3 The proposed model

3.1 The model

Let $q_i \in (0; 1)$ and $f_i(x)$ be the prior probability and the pdf of population $w_i, i = 1, 2, \ldots, k$, respectively. Suppose we have a dataset of groups as follows:

$$\widehat{X} = \left\{\widehat{X}_{N_1}, \widehat{X}_{N_2}, \ldots, \widehat{X}_{N_k}\right\},$$

where N_i is the number of elements in w_i, $N_1 + N_2 + \cdots + N_k = N$,

$\widehat{X}_{N_i}$ is dataset of the group w_i.

The proposed model for classification the elements $\widehat{Y} = \{\widehat{Y}_1, \widehat{Y}_2, \ldots, \widehat{Y}_t\}$ into the known groups $(w_1, w_2, \ldots, w_k)$ includes the following steps:

Step 1 Standardize the data of populations and classified elements on a scale of [0, 1] using (12).

$$\begin{aligned} X_{N_i} &= \frac{\widehat{X}_{N_i}}{\max\{\widehat{X}, \widehat{Y}\}}, Y_i \\ &= \frac{\widehat{Y}_j}{\max\{\widehat{X}, \widehat{Y}\}}, i = 1, 2, \ldots, k; j = 1, \ldots, t. \end{aligned} \quad (12)$$

After this step, we obtain $X = \{X_{N_1}, X_{N_2}, \ldots, X_{N_k}\}, Y = \{Y_1, Y_2, \ldots, Y_t\}$ with $X_{N_i}, Y_i \in [0, 1]$.

Step 2 Encode the elements into chromosomes in binary form by (13):

$$u_l = \begin{cases} 1 & \text{if } X_{N_i} \in w_i \\ 0 & \text{otherwise} \end{cases}, l = \overline{1, N}, 1 \le i \le k. \quad (13)$$

Step 3 Build the chromosomes, with each chromosome being determined by (13). The value of each chromosome is the number of elements in each population, as well as the training data generated for each population. After that, we can proceed to solve the following problems:

- Find the prior probability by (14):

$$q_i = \frac{T_i}{kN}, i = 1, \ldots, k, \quad (14)$$

where T_i is the number of elements in the training set of the ith group.

- Estimate the pdf for each population by (11).
- Classify the elements by (1), and calculate Bayes error by (3).

Step 4 Utilize the operators such as selection, crossover, and mutation.

- Crossover: Two individuals are randomly selected, and the crossing points are chosen arbitrarily. Let P_1 and P_2 be two parents, the crossover point be donated by "|". The two children C_1 and C_2 are produced as follows:

$$\begin{cases} P_1 = \{a_1, a_2, |a_3|, a_4, a_5, |a_6|, a_7\} \\ P_2 = \{b_1, b_2, |b_3|, b_4, b_5, |b_6|, b_7\} \end{cases} \Rightarrow \begin{cases} C_1 = \{a_1, a_2, b_3, a_4, a_5, b_6, a_7\} \\ C_2 = \{b_1, b_2, a_3, b_4, b_5, a_6, b_7\} \end{cases}$$

- Mutation: First, the algorithm selects a fraction of the vector entries of an individual for mutation, where each entry has a probability of being mutated. In this study, we use a mutation rate of 0.01. In the second step, the algorithm replaces each selected entry with a random number selected uniformly from the range specified by (15) for that entry:

$$x_i' = \begin{cases} 1 - x_i, & \text{if } U(0, 1) < P_m, \\ x_i, & \text{otherwise}, \end{cases} \quad (15)$$

where $P_m = 0.01$ is the mutation probability.

- Selection: This section uses the Roulette wheel. The algorithm selects one of the sections using a random number, with the probability of selection being equal to the area of that section. The probability P_i for each individual is defined as follows:

$$P_i = \frac{Be_i}{\sum_{j=1}^{N} Be_j}.$$

The parameters of genetic algorithm are summarized in Table 1.

Step 5 Estimate the pdf by (11), calculate the Bayes error by (3), the prior probability by (14), classify the elements by (1), and compute the empirical error by (2) from the new chromosome obtained from Step 4.

Step 6 Let t be the number of iterations, $Be_{\text{best}}^{(t)}$ be the Bayes error of best chromosome in the tth iteration, and $E(Be_i^{(t)})$ be the average of the Bayes error of 100 chromosomes in the population. If

$$|Be_{\text{best}}^{(t)} - E(Be_i^{(t)})| \le \varepsilon, i = \overline{1, N} \quad (16)$$

Table 1 Used parameters of genetic algorithm

Parameter	Value
Population size	100
Encoding variable	Binary
Chromosome length	Data length
Generations	1000
Selection Operator	Roullette
Crossover probability (P_c)	0.85
Mutation probability (P_m)	0.01
Maximum iterations (t)	1000

then the algorithm will stop and publish Bayes and empirical errors; else, we return Step 4 and Step 5. At the end of this step, we obtain the optimal chromosome and the best value of the objective function. Using this chromosome, we can determine the rate at which to divide the training set for each population simultaneously, as well as the pdf for each population. By applying the proposed classification rule (1), we can obtain the final classification result.

The flowchart of the proposed model is presented in Fig. 1.

Figure 1 shows that the proposed algorithm begins by normalizing the input data to the interval [0,1]. Subsequently, the data generated in Step 1 are encoded using a binary scale. This step aims to calculate the prior probability and partition the training set for estimating the probability density functions (pdfs). Following the execution of the Bayesian method, the algorithm compares the Bayes error as the stopping condition to identify the training set that best represents the groups with their corresponding prior probabilities and pdfs.

To guarantee the convergence of the proposed model, we take into account condition (16), which has been demonstrated in previous studies like [22, 45].

In this model, the genetic algorithm is employed to determine the optimal training set and effectively classify new elements. While there are currently various methods available for constructing training data, such as [19, 24, 35] (with the most popular method being CNN), they do not automatically determine the number of classes for the training data [24]. Typically, researchers need to manually select fixed values for each subtraining data or utilize all the original data to create the training set. In this study, we utilize the genetic algorithm to select the key objects that participate in the training process and employ the Bayesian method to classify new elements based on these training data.

3.2 Comparing the computational complexity of models

In this section, some symbols have been introduced and new ones are presented in Table 2.

Based on the work of Wang and Lin [44], the computational complexity of the models is presented in Table 3:

In general, Table 3 shows that the proposed model exhibits lower computational complexity compared to the Multi-SVM, PSO-SNN, and transfer learning models, while being higher than the remaining models. However, it should be noted that the computational complexity in these models depends on parameters that are not entirely identical; therefore, the aforementioned conclusion is not absolute.

4 The numerical example and application

4.1 The numerical example

This section illustrates the proposed model step by step using a benchmark dataset.

The dataset used is the Breast Tissue dataset (http://archive.ics.uci.edu/ml/datasets/breast+tissue), which consists of 106 elements divided into 6 groups: carcinoma (Car), fibro-adenoma (Fad), mastopathy (Mas), glandular (Gla), connective (Con), and adipose (Adi). Each element has 9 observed variables. Table 4 summarizes the characteristics of this dataset.

Select eight elements randomly:

$$Y_1 = \widehat{X}_1, Y_2 = \widehat{X}_2, Y_3 = \widehat{X}_3, Y_4 = \widehat{X}_4, Y_5 = \widehat{X}_5, Y_6 = \widehat{X}_{101}, Y_7 = \widehat{X}_{102}, Y_8 = \widehat{X}_{103}$$

where $\{Y_1, Y_2, Y_3, Y_4, Y_5\} \in w_1, \{Y_6, Y_7, Y_8\} \in w_6$ to classify at the same time.

Step 1 First of all, we normalize 106 elements of the training data in the range [0, 1]. For example, the classified 8 elements are normalized in Table 5.

Step 2 Initializing the first chromosomes by the binary encode with the length of 106 values, we have the following chromosome:

0 1 0 0 0 1 1 0 1 0 0 1 1 0 1 0 1 0 1 0 0 0 0 0 0 0 1 0 1 1 1 1 0 0 0 1 1 0 1 0 1 1 0 1 1 1 1 0 1 0 0 0 1 0 1 0 0 1 1 1 0 1 1 0 0 0 1 1 0 0 0 0 0 0 1 1 1 1 0 1 0 0 0 0 0 1 1 1 0 1 0 0 1 1 0 1 1 0 0 0 0 1 0 0 0 0

The value 1 shows that it exists in training set, and 0 is contrariety.

Step 3 The result of training process in first time is shown in Table 6.

This training set is used to estimate the pdfs of six populations that they are given in Fig. 2.

At this time, we obtain the result as follows:

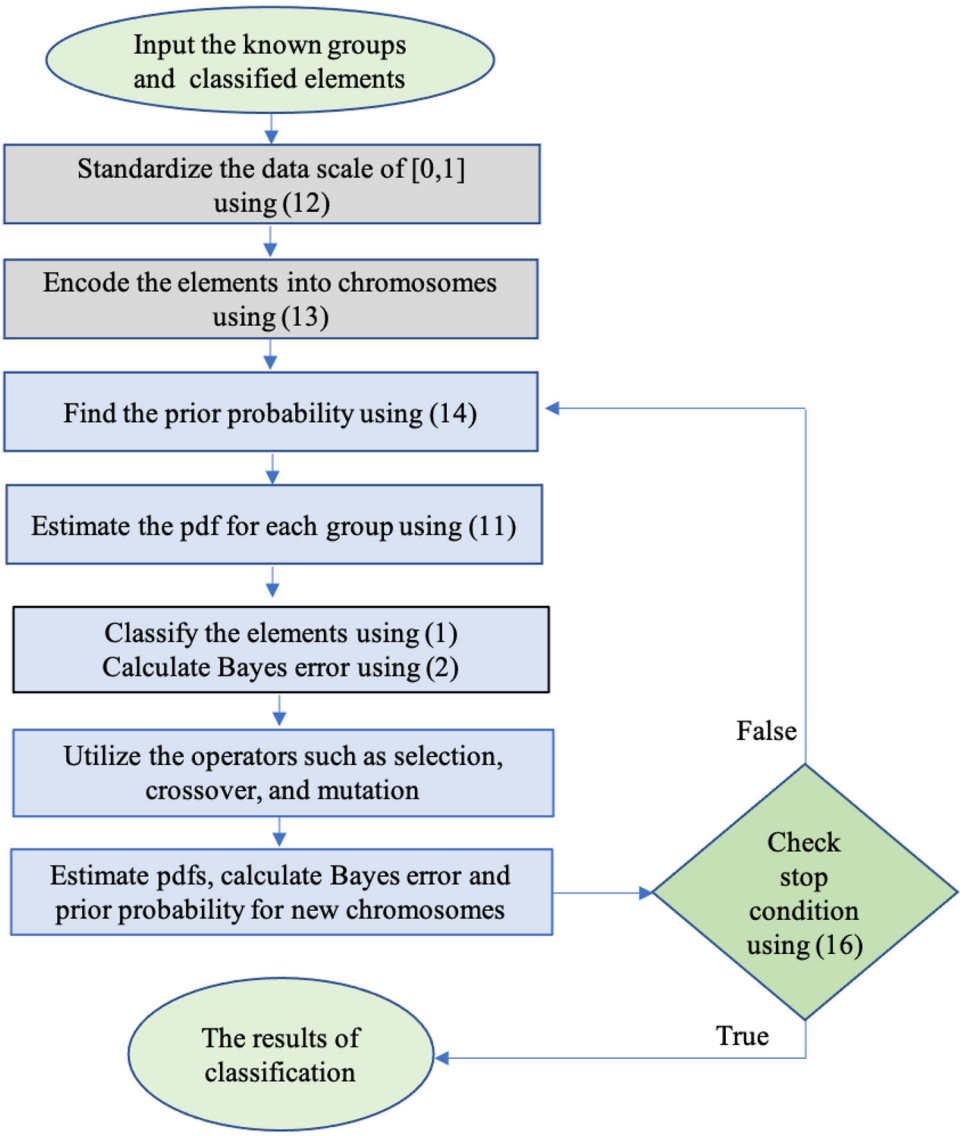


Fig. 1 Flowchart of the proposed model

Table 2 Used symbols

Symbol	Explanation
k	The number of classes
N	The number of elements
n	The number of dimensions
d	The number of variables
t	The number of iterations
L	The number of layers in the network (CNN)
S	The size of each layer
M	The number of particles (PSO-SNN)
P	The number of chromosomes in the genetic algorithm
$H \times W$	The dimensions of input feature map (transfer learning)
$F \times F$	The filter size (transfer learning)
K	The number of filters in the layer

Table 3 Computational complexity of models

Model	Computational complexity
QDA	$O(N.n^2 + k.n^3)$
LDA	$O(N.n^2)$
Naive Bayes	$O(N.n)$
Fisher	$O(N.n^2 + n^3)$
Nguyentrang and Vovan [27]	$O(N.k.n)$
Vovan [39]	$O(N.k.n)$
Multi-SVM	$O(k^2.d^3.n)$
CNN	$O(N.L)$
PSO-SNN	$O(M.L.S^2.N.t)$
Transfer learning	$O(H.W.F^2.K)$
Proposed model	$O(t.k.P.N)$

Table 4 Description for the Breast dataset

Group	Symbol	No	Numerical order
Car	w_1	21	$\widehat{X}_1, \ldots, \widehat{X}_{21}$
Fad	w_2	15	$\widehat{X}_{22}, \ldots, \widehat{X}_{36}$
Mas	w_3	18	$\widehat{X}_{37}, \ldots, \widehat{X}_{54}$
Gla	w_4	16	$\widehat{X}_{55}, \ldots, \widehat{X}_{70}$
Con	w_5	14	$\widehat{X}_{74}, \ldots, \widehat{X}_{83}$
Adi	w_6	22	$\widehat{X}_{84}, \ldots, \widehat{X}_{106}$

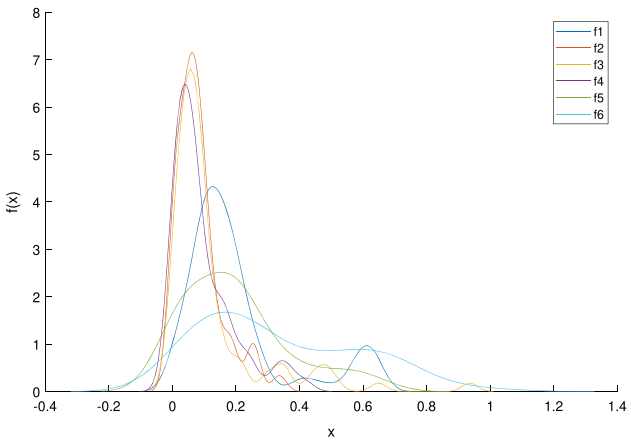


Fig. 2 Pdfs of six populations in the first iteration

Table 6 Size of the training set in first iteration

Population	w_1	w_2	w_3	w_4	w_5	w_6
No. of elements	9×9	6×9	10×9	8×9	5×9	9×9

Table 7 Size of the training set of populations for the second iteration

Class	w_1	w_2	w_3	w_4	w_5	w_6
No. elements	9×9	6×9	10×9	8×9	5×9	9×9

- The prior probabilities:

$$q_1 = 0.192, q_2 = 0.128, q_3 = 0.213, q_4 = 0.170,$$
$$q_5 = 0.101, q_6 = 0.192.$$

- The Bayes error: $Be = 0.023$.
- The result of classifying: $\{Y_1, Y_2, Y_4\} \in w_1$, $\{Y_3, Y_5, Y_7\} \in w_3$, $\{Y_6, Y_8\} \in w_6$.
- The empirical error: $E_{\text{Ner}} = 0.375$.

Step 4 Perform the crossover and mutation operators of the genetic algorithm with rate of 85% and 15%, respectively. After this, we select the two best chromosomes to create the new population and continue to the next iteration.
Step 5 From the new chromosomes obtained from Step 4, BGA model continues the second iteration and obtains the following results:

- The size of the training set for the six groups is given in Table 7.
- The pdfs of populations are presented in Fig. 3.
- The prior probabilities:

$$q_1 = 0.192, q_2 = 0.127, q_3 = 0.213, q_4 = 0.170,$$
$$q_5 = 0.101, q_6 = 0.192.$$

- The Bayes error: $Be = 0.018$.
- The result of classifying:

$$\{Y_1, Y_2, Y_4\} \in w_1, \{Y_3, Y_7\} \in w_3, \{Y_5\} \in w_5 \{Y_6, Y_8\} \in w_6.$$

Table 5 Standardized data of eight elements

Y_1	0.187	0.523	0.069	0.215	0.039	0.182	0.138	0.226	0.192
Y_2	0.118	0.633	0.567	0.114	0.018	0.159	0.160	0.101	0.138
Y_3	0.197	0.649	0.136	0.249	0.068	0.274	0.178	0.260	0.227
Y_4	0.136	0.672	0.612	0.129	0.031	0.239	0.204	0.108	0.170
Y_5	0.130	0.560	0.522	0.117	0.019	0.161	0.159	0.106	0.147
Y_6	0.714	0.188	0.266	0.311	0.088	0.284	0.388	0.290	0.712
Y_7	0.714	0.299	0.225	0.489	0.230	0.470	0.468	0.490	0.721
Y_8	0.929	0.560	0.445	1.000	1.000	1.000	0.960	1.000	0.920

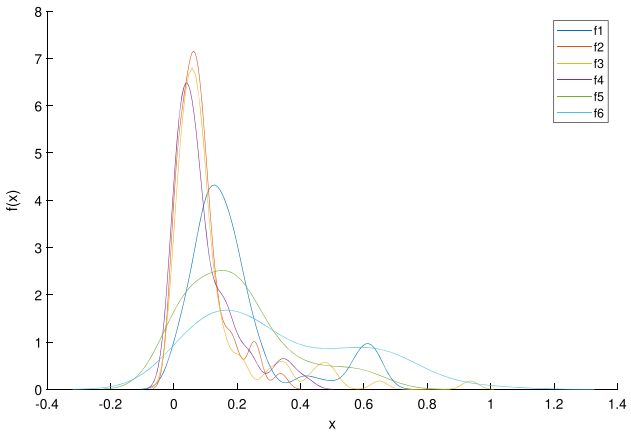


Fig. 3 Pdfs of the six populations in second iterations

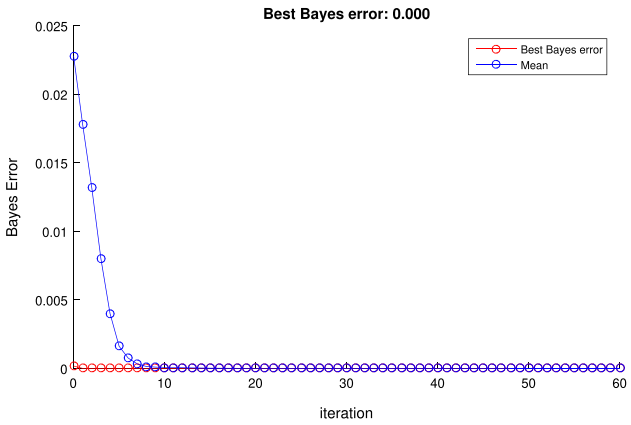


Fig. 4 Convergence of the proposed model for Breast dataset after 60 iterations

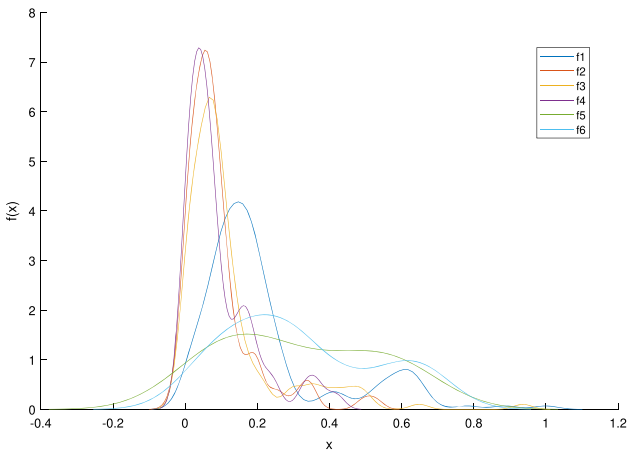


Fig. 5 Pdfs of six populations for the final iteration

- The empirical error: $E_{\text{Ner}} = 0.375$.

Step 6: Performing the next iteration, the proposed model ends with 60 iterations shown in Fig. 4 and obtains the results as follows:

Table 8 Size of the training set of six populations in final iteration

Class	w_1	w_2	w_3	w_4	w_5	w_6
No. elements	19×9	11×9	17×9	13×9	4×9	6×9

- The pdfs are shown in Fig. 5.
- The size of the training set is given in Table 8.
- The prior probabilities:

$$q_1 = 0.175, q_2 = 0.095, q_3 = 0.190, q_4 = 0.175, q_5 = 0.111, q_6 = 0.254.$$

- The Bayes error: $Be = 0.000$.
- The result of classifying:

$$\{Y_1, Y_2, Y_3, Y_4, Y_5\} \in w_1; \{Y_6, Y_7, Y_8\} \in w_6.$$

- The empirical error: $E_{\text{Ner}} = 0.000$.

For this dataset, the empirical error of linear discriminant analysis (LDA), quadratic discriminant analysis (QDA), Fisher, naive Bayes, and Bayesian methods of Nguyentrang and Vovan [27] and Vovan [39] are 0.125, 0.125, 0.250, 0.125, 0.100, and 0.063, respectively. It means that the proposed model obtains the best result in comparing to other models.

4.2 The real application

This section applies the proposed model and compares it to other models using an actual dataset on chronic renal failure at a hospital in Can Tho City, Vietnam. The dataset consists of 259 patients, out of which 22 patients died during treatment (W_1), and 237 patients showed positive treatment outcomes (W_2). The variables under consideration are presented in Table 9.

Selecting a test set randomly, we choose 10 patients denoted as Y_1, Y_2, Y_3, Y_4, Y_5 from W_1, and $Y_{255}, Y_{256}, Y_{257}, Y_{258}, Y_{259}$ from W_2. The remaining patients are assigned to the training set.

The proposed model achieves convergence after 300 iterations, as illustrated in Fig. 6. After this convergence, we obtain the following results:

- The pdfs of two groups are shown in Fig. 7.
- The size of the training set is 127×9 for W_1 and 5×9 for W_2.
- The prior probabilities: $q_1 = 0.981, q_2 = 0.019$.
- The Bayes error: $Be = 0.000$.
- The result of classifying:

$$\{Y_1, Y_2, Y_3, Y_4, Y_5\} \in w_1; \{Y_{255}, Y_{256}, Y_{257}, Y_{258}, Y_{259}\} \in w_2.$$

Table 9 Used variables and their meaning

Variable	Name	Description
X_1	Ferritin	Ferritin level in blood
X_2	Albumin	Albumin level in blood
X_3	Urea	Urea level in blood
X_4	Time 1	The length of time the patient is on dialysis
X_5	Time 2	The length of time the patient has chronic renal failure
X_6	MCHC	Mean corpuscular hemoglobin concentration
X_7	Creatinine	Creatinine level in blood
X_8	Na	Sodium level in blood
X_9	Ca	Calcium level in blood

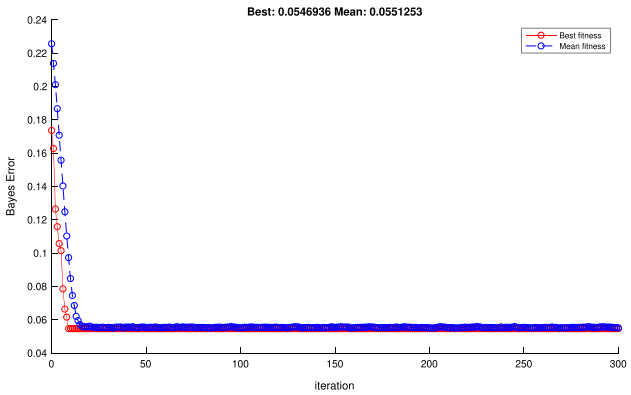


Fig. 6 Convergence of the proposed model for chronic renal failure data

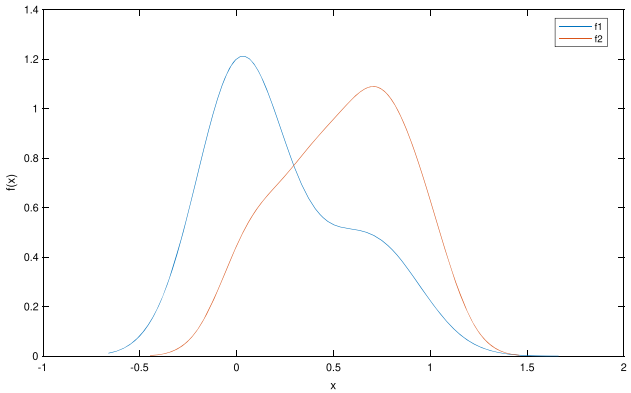


Fig. 7 Pdfs of chronic renal failure data

- The empirical error: $E_{\text{Ner}} = 0.000$.

Compared to the developed and other models, we have Table 10.

With $E_{\text{Ner}} = 0$ in Table 10, the CNN, PSO-SNN, and proposed models have demonstrated the best performance in classifying this real dataset.

Table 10 Empirical error and execution time of the methods for chronic renal failure data

Method	E_{Ner}	Execution time (s)
LDA	0.062	2.057
QDA	0.062	0.309
Naive Bayes	0.500	1.636
Fisher	0.500	0.159
Nguyentrang and Vovan [27]	0.200	1.342
Vovan [39]	0.200	1.108
Multi-SVM	0.032	3.163
CNN	0.000	9.320
PSO-SNN	0.200	4.384
Transfer learning	0.000	121.376
Proposed model	0.000	368.460

Fig. 8 Two sample images of cars and motorcycles

5 Applying in classifying images

5.1 Extracting the features for image

The first step in classifying images is to extract their features. There are several popular methods for doing this, such as using interval data from texture characters [6, 30], the gray-level co-occurrence matrix (GLCM) [9, 41, 42], and pdfs [40]. In this study, we use the method of extracting texture characters from images and reduce their size using the Gabor

Fig. 9 Photographs of cars and motorcycles for classification

filter [11, 21], which is widely used in many current research studies [11, 17, 21]. Compared to existing filters [17, 21, 34], the Gabor filter has several advantages. We use the 2D Gabor filter, which is based on frequency and orientation representations. It is a Gaussian kernel function modulated by a sinusoidal plane wave and oriented at a specific angle:

$$G(x, y) = \frac{f^2}{\pi\gamma\eta}\exp\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right)\exp^{(j2\pi fx'+\phi)}, \quad (17)$$

where

$$\begin{cases} x' = x\cos\theta + y\sin\theta, \\ y' = -x\sin\theta + y\cos\theta, \end{cases}$$

where

f is the frequency of the sinusoid,

θ represents the orientation of the normal to the parallel stripes of a Gabor function,

ϕ is the phase offset,

σ is the standard deviation of the Gaussian envelope,

γ is the spatial aspect ratio which specifies the ellipticity of the support of the Gabor function.

The specific values for Gabor filter used in this study are given in Table 11.

For this filter, each image will be transformed into a multi-dimensional vector. For example, if a grayscale image has dimensions of 280 × 200, the resulting feature vector will have a size of 65,000. Typically, five scales and nine orientations are used for the 2D Gabor subfilters, each with a size of 9 × 9 [11]. However, for many datasets, using two scales and five orientations can still achieve maximum consistency in the images.

Since adjacent pixels in an image often exhibit high correlation, we reduce this redundant information using the Gabor filter. As a result, the character vector for an image will have a size of 65000/(32 × 32). These vectors are then normalized to have a zero mean and unit variance. Additionally,

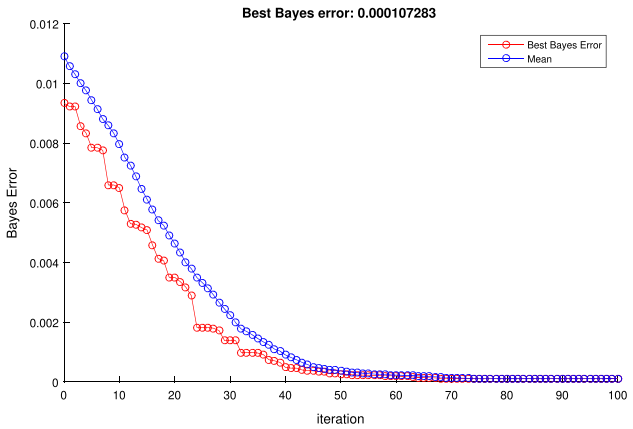


Fig. 10 Convergence of the proposed model for car and motorbike dataset

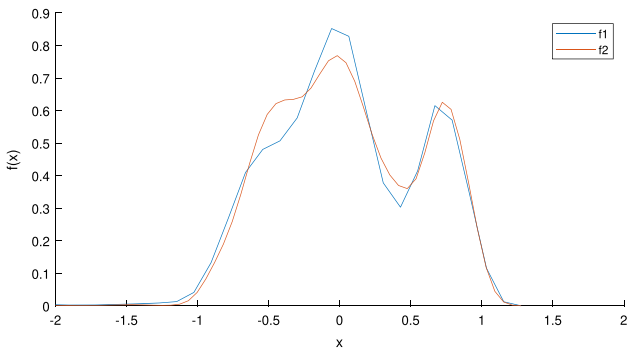


Fig. 11 Pdfs of two populations in final iteration

we apply the principal component analysis (PCA) method to further reduce the size of the character vector to 100 [15].

5.2 Some the specific applications

In this section, the study uses the classification models for 3 image datasets. For each model, we perform 50 times to calculate the average empirical error. In order to evaluate the effectiveness of the proposed model, we compare it to other models, including linear discriminant analysis (LDA), quadratic discriminant analysis (QDA), naive Bayes, Fisher,

Table 11 Value of the parameters in Gabor filter

Parameter	Symbol	Values
Orientation	θ	$\left\{0, \frac{\pi}{8}, \frac{\pi}{8}, \frac{2\pi}{8}, \frac{3\pi}{8}, \frac{4\pi}{8}, \frac{5\pi}{8}, \frac{6\pi}{8}, \frac{7\pi}{8}\right\}$
Wavelength	λ	$\left\{4, \sqrt[4]{2}, 8, \sqrt[8]{2}, 16\right\}$
Phase	Ψ	$\left\{0, \frac{\pi}{2}\right\}$
Gaussian envelope	γ	$\gamma = \lambda = \left\{4, \sqrt[4]{2}, 8, \sqrt[8]{2}, 16\right\}$

Nguyentrang and Vovan [27], Vovan [39], Multi-SVM, PSO-SNN [23] (the system includes the combination of PSO and typical shallow neural network), transfer learning (from source https://www.mathworks.com/help/deeplearning/ref/alexnet.html), and convolutional neural network (CNN).

In order to evaluate the statistical significance of results, Shapiro–Wilk test, one-way ANOVA, and post hoc with Tukey test are used.

5.2.1 Car and motorbike

The first application consists of 530 images, comprising 250 cars and 280 motorbikes. Details of the data can be found at http://www.vision.caltech.edu/html-files/archive.html. Figure 8 shows two sample images from this dataset.

Selecting 9 images randomly to classify (five motorbikes and four cars), these images are given in Fig. 9.

Extracting the image features and using the Gabor filter, we have the dataset with size of 530×6380. Next, we applied the PCA method, resulting in a dataset with a size of 530×100.

Performing the proposed model after 100 iterations, the algorithm stops with its convergence shown in Fig. 10.

At this time, we also have the following results:

- The training set of two populations have the size of 128×100 and 142×100, respectively.
- The pdfs of populations are given in Fig. 11.
- The prior probabilities: $q_1 = 0.474$, and $q_2 = 0.526$.
- The Bayes error: $Be = 0.0002$.
- The result of classification: $\{Y_1, Y_2, Y_3, Y_4, Y_5, Y_6\} \in w_1$ and $\{Y_7, Y_8, Y_9\} \in w_2$.
- The empirical error: $E_{\text{Ner}} = 0.000$.

After conducting 50 trials, we compared the average empirical error and execution time of the developed model with existing models in the field. The results are shown in Table 12.

Table 12 shows that the CNN, transfer learning, and proposed models give the best results ($E_{\text{Ner}} = 0.000$). However, we would like to point out that the proposed model has been optimized for training, which is why it performs better. The CNN and transfer learning models give good result because they uses the optimal training set from the proposed model. We randomly chose some other training sets to evaluate the CNN and transfer learning models and found that their E_{Ner} were greater than 0 in some cases. Moreover, the execution times of the proposed model has the smallest value while the execution times of transfer learning model is biggest.

Table 12 Empirical error and execution time of the methods for car and motorbike data

Method	E_{Ner}	Execution time (s)
LDA	0.111	1.190
QDA	0.111	0.355
Naive Bayes	0.222	2.805
Fisher	0.333	0.042
Nguyentrang and Vovan [27]	0.111	1.473
Vovan [39]	0.055	1.201
Multi-SVM	0.019	12.403
CNN	0.000	12.041
PSO-SNN	0.404	7.798
Transfer learning	0.000	282.617
Proposed model	0.000	11.154

Next, the Shapiro–Wilk test was used to check the normality of the E_{Ner} data, and one-way ANOVA was used to evaluate the significant differences between the models. The results are summarized in Table 13.

Compare each pair of the proposed and CNN models with others by the post hoc with Tukey test, we always obtain P value as 0.

In short, the results from Tables 12 and 13 show that the proposed model has the smallest E_{Nerr}, and this result is significantly different from other models.

5.2.2 People face

This application examines the Faces95 dataset, which comprises 240 images of 12 people, with 20 images per person. The dataset is available at https://cswww.essex.ac.uk/mv/allfaces/faces95.html, and some sample images are shown in Fig. 12.

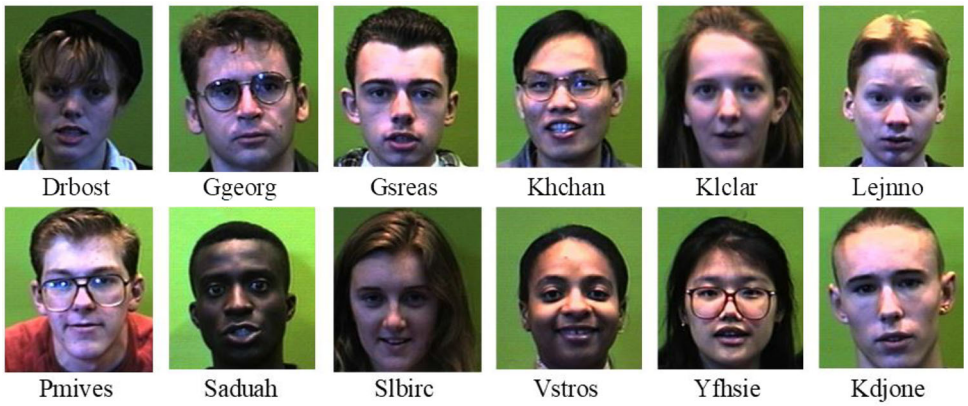
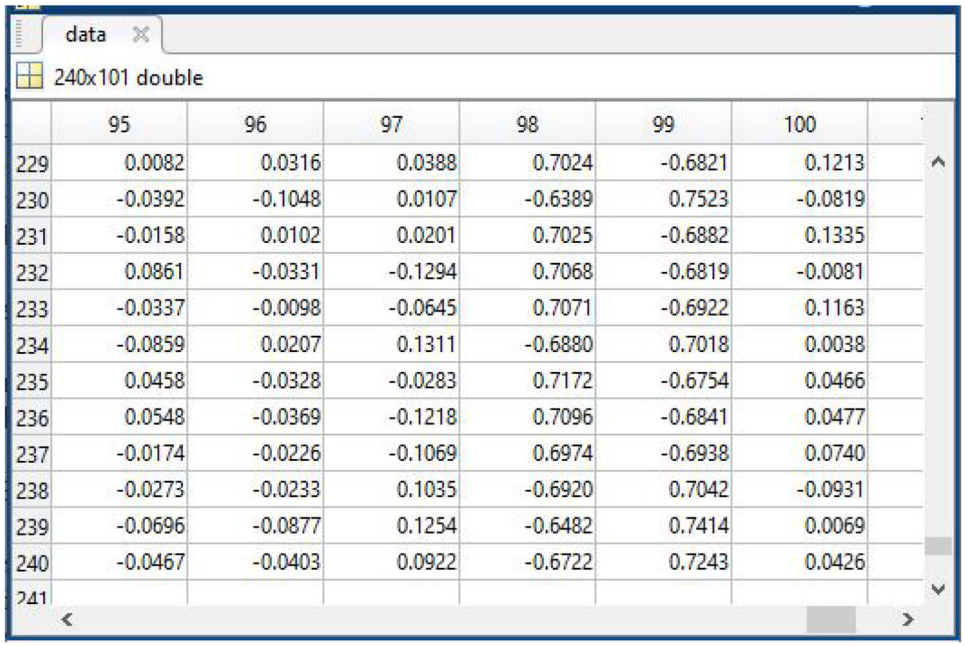


Fig. 12 Some sample images of the Faces95 dataset



	95	96	97	98	99	100
229	0.0082	0.0316	0.0388	0.7024	-0.6821	0.1213
230	-0.0392	-0.1048	0.0107	-0.6389	0.7523	-0.0819
231	-0.0158	0.0102	0.0201	0.7025	-0.6882	0.1335
232	0.0861	-0.0331	-0.1294	0.7068	-0.6819	-0.0081
233	-0.0337	-0.0098	-0.0645	0.7071	-0.6922	0.1163
234	-0.0859	0.0207	0.1311	-0.6880	0.7018	0.0038
235	0.0458	-0.0328	-0.0283	0.7172	-0.6754	0.0466
236	0.0548	-0.0369	-0.1218	0.7096	-0.6841	0.0477
237	-0.0174	-0.0226	-0.1069	0.6974	-0.6938	0.0740
238	-0.0273	-0.0233	0.1035	-0.6920	0.7042	-0.0931
239	-0.0696	-0.0877	0.1254	-0.6482	0.7414	0.0069
240	-0.0467	-0.0403	0.0922	-0.6722	0.7243	0.0426
241						

Fig. 13 Extracted data for the Faces95 dataset

Extracting the texture features of the images using the Gabor filter and PCA method, we obtained a dataset for classification. Some examples of the dataset are shown in Fig. 13.

Suppose we need to classify 5 images into 12 groups. The images that have been classified are shown in Fig. 14.

After 40 iterations of applying the proposed model (as shown in Fig. 15), the algorithm stops, and we obtain the following results:

- The size of the training set of populations is given in Table 7.
- The pdfs of populations are given in Fig. 16.
- The prior probabilities:

$$q_1 = 0.089; q_2 = 0.089; q_3 = 0.089; q_4 = 0.074;$$
$$q_5 = 0.104; q_6 = 0.082; q_7 = 0.059; q_8 = 0.082;$$
$$q_9 = 0.074; q_{10} = 0.096; q_{11} = 0.082; q_{12} = 0.082;$$

- The Bayes error: $Be = 0.000026$.
- The result of classification: 'Drbost', 'Ggeorg', 'Gsreas',' Khchan', 'Khchan'
- The empirical error: $E_{\text{Ner}} = 0.000$.

By comparing the proposed model with existing models using 50 trials for each model, we obtained the results summarized in Table 15.

After conducting the Shapiro–Wilk test, one-way ANOVA test, and post hoc with Tukey test to evaluate the significant differences between the models, we obtained a p value of 0 for all tests. Therefore, we can conclude that the proposed model achieved the best result, and there is a significant difference in E_{Ner} compared to the other models (excluding CNN and transfer learning models). For this dataset, the CNN gives the best result in terms of execution time, while the transfer learning model takes a lot of time to complete. The proposed model has a longer execution time than the CNN model. In our opinion, the reason for this can be the com-

Fig. 14 Images of five people need to classify

Table 13 Result for Shapiro–Wilk test and one-way ANOVA for E_{Nerr} with motorbike data

Test	The value of test	P value	Conclusion
Shapiro–Wilk	0.758	$2.2e^{-16}$	Data have normal distribution
One-way ANOVA	3724	0.000	E_{Nerr} of models are different

Table 14 Size of the training set of populations

Population	w_1	w_2	w_3	w_4	w_5	w_6
No. elements	12×100	12×100	12×100	10×100	14×100	11×1000
Population	w_7	w_8	w_9	w_{10}	w_{11}	w_{12}
No. elements	8×100	11×100	10×100	13×100	11×100	11×100

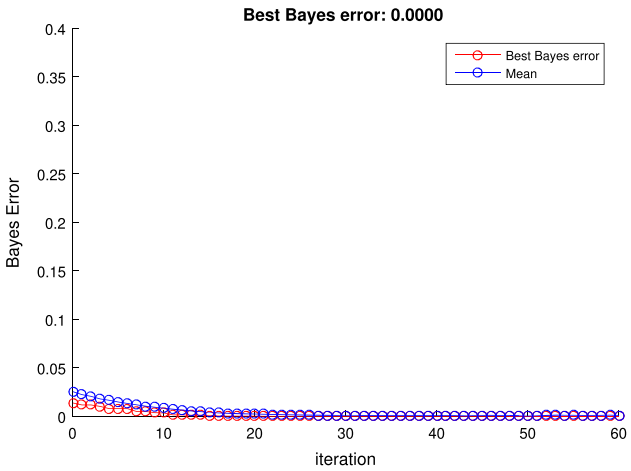


Fig. 15 Convergence of model for Faces95 data

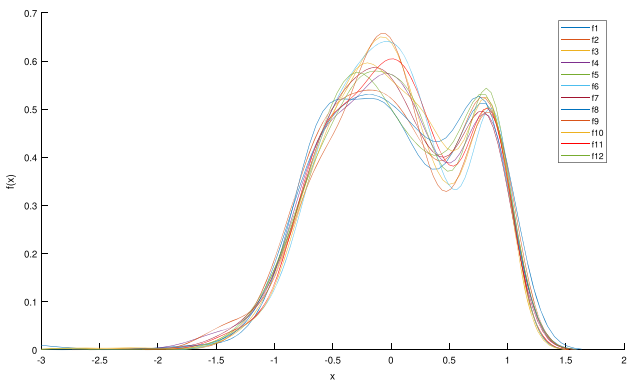


Fig. 16 Pdfs of 12 populations

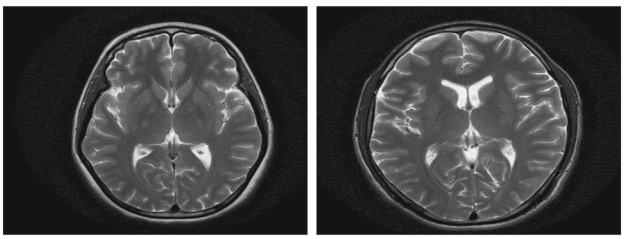


Fig. 17 Two brain MRI samples of abnormal and normal images

plexity of the image dataset. This dataset has many colors and layers, so the proposed model has to go through many iterations to converge. The CNN model has better class separation for learning in this case. The limit for setting up a program, which we have developed for the proposed model, can also be reason for the high time costs.

5.2.3 Brain MRI

The Brain MRI database used in this study consists of 356 images with a size of 800×600. Of these images, 209 are normal, and 147 are abnormal, showing conditions such as bleeding, clotting, acute infarction, tumors, and trauma. Detailed information about this dataset can be found at https://www.nitrkl.ac.in/CS/NITR-DHH.htm. This is popular benchmark used in many studies to evaluate the effectiveness of a classification model [4]. This is also considered a complex dataset in the classification problem.

Examples of two images from the dataset are shown in Fig. 17.

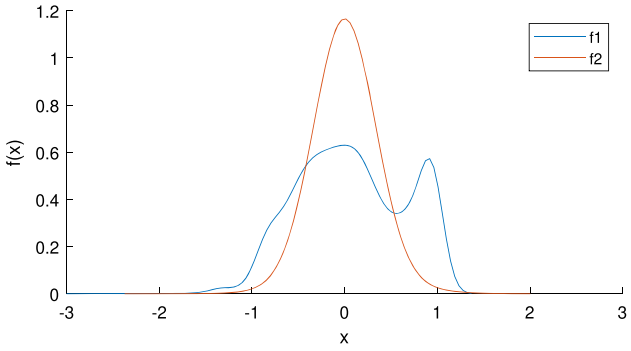


Fig. 18 Pdfs of two populations

Table 15 Empirical error and execution time of the methods in Face95 data

Method	E_{Ner}	Execution time (s)
LDA	0.200	1.194
QDA	0.200	0.347
Naive Bayes	0.200	1307.830
Fisher	0.200	0.072
Nguyentrang and Vovan [27]	0.150	13.256
Vovan [39]	0.100	12.472
Multi-SVM	0.200	8.512
CNN	0.000	15.453
PSO-SNN	0.170	63.858
Transfer learning	0.000	526307.257
Proposed model	0.000	63.759

In this application, we use 56 images for normal group and 10 for abnormal group to classify. After 80 iterations, we obtain the results as follows:

- The training of two populations has the size 60×100 and 240×100, respectively.
- The pdfs of two populations are shown in Fig. 18.
- The prior probabilities: $q_1 = 0.2$, and $q_2 = 0.8$.
- The Bayes error: $Be = 0.000226$.
- The result of classifying:

$$\{Y_1, Y_2, \ldots, Y_{10}, Y_{13}, \ldots, Y_{20}, Y_{32}, Y_{33}, Y_{38}\} \in w_1;$$
$$\{Y_{11}, Y_{12}, Y_{21}, \ldots, Y_{31}, Y_{34}, \ldots, Y_{37}, Y_{39}, Y_{40}\} \in w_2$$

- The empirical error: $E_{\text{Ner}} = 0.0758$.

Comparing the proposed model with existing models when performing 50 times for each model, we have Table 16.

Similar to the previous two applications, we obtained a p value of 0 for all tests when using the Shapiro–Wilk test, one-way ANOVA, and post hoc with Tukey test to compare the differences between models. Therefore, we can conclude that the proposed model achieved the best result and has a significant difference in E_{Ner} compared to the other models.

Table 16 Empirical error and execution time of the methods for Brain MRI data

Method	E_{Ner}	Execution time (s)
LDA	0.197	8.236
QDA	0.197	6.266
Naive Bayes	0.182	1.261
Fisher	0.561	0.136
Nguyentrang and Vovan [27]	0.167	7.213
Vovan [39]	0.136	5.723
Multi-SVM	0.106	35.112
CNN	0.091	34.010
PSO-SNN	0.206	35.021
Transfer learning	0.374	162.318
Proposed model	0.076	33.210

From this study, we observed that the proposed model outperformed the CNN model and achieved the best results for complex data, such as Brain MRI images. Furthermore, we found that the proposed model consistently produced superior classification results compared to the transfer learning and CNN models for gray image sets.

6 Conclusion

This study makes a significant contribution to the field of classifying. After summarizing the issues related to the classification problem using the Bayesian method and surveying the boundaries of the Bayes error, the research proposes an automatic classification model based on the Bayesian method and genetic algorithm. This model incorporates several improvements, including finding the prior probability, optimizing the classification error as the objective function in the genetic algorithm, and rationalizing the selection, crossover, and mutation processes. The proposed model automatically selects the training set based on the population to reduce both the Bayes error and empirical error. The research addresses all computational issues in the practical application of the proposed model by establishing a MATLAB procedure. Another significant contribution of the proposed model is its application in image classification. By extracting texture features using the Gabor filter and the PCA method, the size of the input data is reduced, and numerical and applied examples demonstrate the rationality and stability of the proposed model. The advantages of the model over existing models are also highlighted, demonstrating the potential of this study in practical applications.

While the proposed algorithm has the aforementioned advantages, theoretical analysis and practical experience have also revealed limitations. The algorithm's high complexity results in long implementation times for large datasets. To address this issue, future work will focus on applying parallel algorithms to improve efficiency. Additionally, the researchers intend to apply the proposed model to specific real-world problems in different domains, further expanding its practical applicability.

Acknowledgements This research is funded by Van Lang University, Viet Nam under grant number 13/2022/HD-NCKH.

Declarations

Conflict of interest No potential conflict of interest was reported by the authors.

References

1. Bandyopadhyay, S., Maulik, U.: Non-parametric genetic clustering: comparison of validity indices. IEEE Trans. Syst. Man Cybern. Part C **31**(1), 120–125 (2001)
2. Bandyopadhyay, S., Maulik, U.: Genetic clustering for automatic evolution of clusters and application to image classification. Pattern Recogn. **35**(6), 1197–1208 (2002)
3. Behera, D.K., Das, M., Swetanisha, S.: Follower link prediction using the XGBoost classification model with multiple graph features. Wirel. Pers. Commun. **127**, 695–714 (2021)
4. Behera, T.K., Khan, M.A., Bakshi, S.: Brain MR image classification using superpixel-based deep transfer learning. IEEE J. Biomed. Health Inform. (2022). https://doi.org/10.1109/JBHI.2022.3216270
5. Bidi,N., Elberrichi, Z.: Feature selection for text classification using genetic algorithms. In: 8th International Conference on Modelling, Identification and Control, Algiers, Algerial, pp. 806–810 (2016)
6. Celebi, E., Alpkocak, A.: Clustering of texture features for content-based image retrieval. In: International Conference on Advances in Information Systems (ADVIS 2000), pp. 216–225 (2000)
7. Chen, Z., Hongbo, Z., Chao, S., Wenquan, F.: Detection and classification of GNSS signal distortions based on quadratic discriminant analysis. IEEE Access **8**, 25221–25236 (2020)
8. Dietterich, T.G.: An experimental comparison of three methods for constructing ensembles of decision trees: bagging, boosting, and randomization. Mach. Learn. **40**, 139–157 (2000)
9. Fadl, S., Megahed, A., Han, Q., Qiong, L.: Frame duplication and shuffling forgery detection technique in surveillance videos based on temporal average and gray level co-occurrence matrix. Multimed. Tools Appl. **79**, 17619–17643 (2020)
10. Fisher, R.A.: Statistical methods for research workers. In: Breakthroughs in Statistics, pp. 66–70 (1992)
11. Haghighat, M., Zonouz, S., Abdel-Mottaleb, M.: Cloudid: trustworthy cloud-based and cross-enterprise biometric identification. Expert Syst. Appl. **42**(21), 7905–7916 (2015)
12. Hearst, M.A., Dumais, S.T., Osuna, E., Platt, J., Scholkopf, B.: Support vector machines. IEEE Intell. Syst. Their Appl. **13**(4), 18–28 (1998)
13. Hemanth, D.J., Anitha, J.: Modified genetic algorithm approaches for classification of abnormal magnetic resonance brain tumour images. Appl. Soft Comput. **75**, 21–28 (2019)
14. Holland, J.H.: Genetic algorithms and the optimal allocation of trials. SIAM J. Comput. **2**(2), 88–105 (1973)
15. Hu, L., Cui, J.: Digital image recognition based on fractional-order-PCA-SVM coupling algorithm. Measurement **145**, 150–159 (2019)
16. Imandoust, S.B., Bolandraftar, M.: Application of k-nearest neighbor (KNN) approach for predicting economic events: theoretical background. Int. J. Eng. Res. Appl. **3**(5), 605–610 (2013)
17. Kamarainen, J.K., Kyrki, V., Kalviainen, H.: Invariance properties of Gabor filter-based features-overview and applications. IEEE Trans. Image Process. **15**(5), 1088–1099 (2006)
18. Malarvizhi, N., Selvarani, P., Raj, P.: Adaptive fuzzy genetic algorithm for multi biometric authentication. Multimed. Tools Appl. **79**(13), 9131–9144 (2020)
19. Mazurowski, M.A., Habas, P.A., Zurada, J.M., Lo, J.Y., Baker, J.A., Tourassi, G.D.: Training neural network classifiers for medical decision making: the effects of imbalanced datasets on classification performance. Neural Netw. **21**(2), 427–436 (2008)
20. Mehrdad, R., Saman, F., Kamal, B., Mina, S.: Integration of multi-objective PSO based feature selection and node centrality for medical datasets. Genomics **112**(6), 4370–4384 (2020)
21. Meshgini, S., Aghagolzadeh, A., Seyedarabi, H.: Face recognition using Gabor-based direct linear discriminant analysis and support vector machine. Comput. Electr. Eng. **39**(3), 727–745 (2013)
22. Mishra, S., Saha, S., Mondal, S.: GAEMTBD: genetic algorithm based entity matching techniques for bibliographic databases. Appl. Intell. **47**(1), 197–230 (2017)
23. Mousavi, S.M.H., MiriNezhad, S.Y., Mosleh, M.S., Dezfoulian, M.H.: A PSO fuzzy-expert system: as an assistant for specifying the acceptance by NOET measures, at PH.D level. In: Artificial Intelligence and Signal Processing Conference (AISP), Shiraz, Iran, pp. 11–18 (2017)
24. Nalepa, J., Michal, K.: Selecting training sets for support vector machines: a review. Artif. Intell. Rev. **52**(2), 857–900 (2019)
25. Neto, J.G., Ozorio, L.V., De Abreu, T.C.C., Dos Santos, B.F., Pradelle, F.: Modeling of biogas production from food, fruits and vegetables wastes using artificial neural network (ANN). Fuel **285**, 119081 (2021)
26. Nhu, V.H., Zandi, D., Shahabi, H., Chapi, K., Shirzadi, A., Al-Ansari, N., Singh, S.K., Dou, J., Nguyen, H.: Comparison of support vector machine, Bayesian logistic regression, and alternating decision tree algorithms for shallow landslide susceptibility mapping along a mountainous road in the west of Iran. Appl. Sci. **10**(15), 5047 (2020)
27. Nguyentrang, T., Vovan, T.: A new approach for determining the prior probabilities in the classification problem by Bayesian method. Adv. Data Anal. Classif. **11**, 629–643 (2017)
28. Pham-Gia, T., Turkkan, N., Vovan, T.: Statistical discrimination analysis using the maximum function. Commun. Stat. Simul. Comput. **37**(2), 320–336 (2008)
29. Phamtoan, D., Vovan, T., Phamchau, A., Nguyentrang, T., Hokieu, D.: A new binary adaptive elitist differential evolution based automatic k-medoids clustering for probability density functions. Math. Probl. Eng. **6380568**, 1–16 (2019)
30. Phamtoan, D., Vovan, T.: Automatic fuzzy genetic algorithm in clustering for images based on the extracted intervals. Multimedia Tools and Applications **80**, 35193–35215 (2021)
31. Rostami, M., Berahmand, K., Nasiri, E., Forouzandeh, S.: Review of swarm intelligence-based feature selection methods. Eng. Appl. Artif. Intell. **100**, 104210 (2021)
32. Saeid, A., Mehrdad, R., Kamal, B., Parham, M., Mourad, O.: Graph-based relevancy-redundancy gene selection method for cancer diagnosis. Comput. Biol. Med. **147**, 105766 (2022)
33. Scott, D.W.: Multivariate Density Estimation. Wiley, New York (1992)

34. Shen, L., Bai, L., Fairhurst, M.: Gabor wavelets and general discriminant analysis for face identification and verification. Image Vis. Comput. **25**(5), 553–563 (2007)
35. Shen, M., Tang, X., Zhu, L., Du, X., Guizani, M.: Privacy-preserving support vector machine training over block chain-based encrypted IoT data in smart cities. IEEE Internet Things J. **6**(5), 7702–7712 (2019)
36. Sun, F., Xu, Y., Zhou, J.: Active learning SVM with regularization path for image classification. Multimed. Tools Appl. **75**(3), 1427–1442 (2016)
37. Tanveer, M., Tiwari, A., Choudhary, R., Jalan, S.: Sparse pinball twin support vector machines. Appl. Soft Comput. **78**, 164–175 (2019)
38. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Aidan N.G., Kaiser, L., Polosukhin L.: Attention is all you need. In: NIPS'17: Proceedings of the 31st International Conference on Neural Information Processing Systems, December 2017, pp. 6000–6010 (2017)
39. Vovan, T.: L^1-distance and classification problem by Bayesian method. J. Appl. Stat. **44**(3), 385–401 (2017)
40. Vovan, T., Nguyentrang, T., Chengoc, H.: The prior probability in classifying two populations by Bayesian method. Appl. Math. Eng. Reliab. **6**, 35–40 (2016)
41. Vovan, T., Chengoc, H., Nguyentrang, T.: Textural features selection for image classification by Bayesian method. In: 13th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD), Guilin, China, pp. 733–139 (2018)
42. Vovan, T., Phamtoan, D., Nguyenthithuy, D.: Automatic genetic algorithm in clustering for discrete elements. Commun. Stat. Simul. Comput. **50**(6), 1679–1694 (2021)
43. Vovan, T., Phamtoan, D., Lehoang, T., Nguyentrang, T.: An automatic clustering for interval data using the genetic algorithm. Ann. Oper. Res. **303**, 359–380 (2021)
44. Wang, P.W., Lin, C.J.: Iteration complexity of feasible descent methods for convex optimization. J. Mach. Learn. Res. **15**, 1523–1548 (2014)
45. Weishui, W., Chen, X.: Convergence theorem of genetic algorithm. In: IEEE International Conference on Systems, Man and Cybernetics. Information Intelligence and Systems, vol. 3, pp. 1676–1681 (1996)
46. Yin, P., Neubig, G., Yih, W, Riedel, S.: TaBERT: pretraining for joint understanding of textual and tabular data. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pp. 8413–8426 (2020)