West Science Information System and Technology
Vol. 2, No. 01, April 2024, pp. 188~195

Optimizing Liver Disease Detection Through Combining Genetic Evolutionary Algorithm and Linear Discriminant Analysis (LDA)

Dwi Ari Suryaningrum[1] , Muhammad Romadhoni Indra Firmansyah[2]
[1,2]Tulungagung University

Article Info

Article history:

Received April, 2024
Revised April, 2024
Accepted April, 2024

Keywords:

Liver Disease
Early Detection
Genetic Evolutionary Algorithm (GA)
Linear Discriminant Analysis (LDA)

ABSTRACT

Liver diseases such as cirrhosis, hepatocarcinoma and fatty liver disease are global health problems with high morbidity and mortality. Early detection is crucial but is often hampered by the limitations of conventional methods in analyzing medical images and laboratory results. Machine learning and artificial intelligence technologies, particularly Genetic Evolutionary Algorithm (GA) and Linear Discriminant Analysis (LDA), offer opportunities to improve diagnosis accuracy. This research explores the combination of GA and LDA to improve liver disease detection using the ILPD (Indian Liver Patient Dataset) dataset from the UCI Machine Learning Repository. This study aims to optimize feature selection and classification to improve detection accuracy.

The research method includes the use of GA for feature selection and LDA for dimensionality reduction and classification. Tests were conducted on various parameters such as the number of generations, population size, and the combination of crossover and mutation rates in the genetic algorithm. The test results show that the best parameter combination (generation 400, population size 40, crossover rate 0.9, and mutation rate 0.1) results in an Average Forecast Error Rate (AFER) value of 0.0345%, which indicates that the developed detection model is highly accurate.

This study shows that the combination of GA and LDA can improve the effectiveness of liver disease detection compared to conventional methods, with potential practical applications in clinical diagnosis systems.

Corresponding Author:

Name: Dwi Ari Suryaningrum
Institution: Tulungagung University
Email: dwiarisuryaningrum@unita.ac.id

1. INTRODUCTION

Liver diseases, such as cirrhosis, hepatocarcinoma and fatty liver disease, are significant global health problems with high morbidity and mortality rates. Early detection of liver diseases is essential to improve the effectiveness of treatment and reduce the impact of these diseases. However, conventional detection methods are often inadequate due to limitations in the accuracy of medical image analysis and laboratory results [10].

Advances in machine learning and artificial intelligence technologies offer new

opportunities in improving the accuracy and efficiency of medical diagnosis. Genetic Evolutionary Algorithm (GA) and Linear Discriminant Analysis (LDA) are two promising techniques in this field. GA has been widely used in various applications including feature selection and hyperparameter optimization in machine learning models [2]. Linear Discriminant Analysis (LDA) is a frequently used technique for dimensionality reduction and classification that maximizes the ratio of between-class and within-class variations [3]. The combination of GA and LDA is expected to improve the performance of liver disease detection in a more accurate and efficient way than conventional methods.

Combining GA and LDA has the potential to improve accuracy and efficiency in liver disease detection. GA can be used to select the most relevant features, while LDA helps in data dimensionality reduction and improves classification performance. Previous studies have shown that this combination is effective in medical applications, such as hepatocellular carcinoma detection and other diseases [2].

The data used comes from the UCI Machine Learning Repository, specifically the ILPD dataset (Indian Liver Patient Dataset). This dataset contains two classes, expressed as class 1 and class 2, and the information about these classes is contained in a column called class.

Many studies have been conducted using the ILPD dataset (Indian Liver Patient Dataset). For example, Intan Setiawati and her team conducted research by applying the decision tree method and using the Rapidminer version 7.4 application for ILPD dataset processing. Of the 583 data processed, 433 data were used as training data and 150 data were used as testing data. This study shows that only two attributes, namely SPGT_AA and Age, are the most influential in the classification of liver disease from 10 attributes in the ILPD dataset. The accuracy of this study reached 72.67% [4].

Furthermore, Popon Handayani and her colleagues conducted research by applying the decision tree method using the C4.5 algorithm and neural networks. The dataset used in this study consists of 11 attributes, including 10 attributes and 1 class. The results showed that the accuracy of the decision tree using the C4.5 algorithm reached 75.56% with an AUC of 0.898, while the accuracy using the neural network reached 74.17% with an AUC of 0.671. Based on these results, it can be concluded that the use of decision trees is more accurate in predicting liver disease [5].

This study aims to explore the combination of GA and LDA to improve liver disease detection. By utilizing the ability of GA to select a subset of relevant features and the power of LDA in classification, a more accurate and reliable detection model is expected. This study will analyze the effectiveness of the combination of GA and LDA in improving detection accuracy compared to other methods and evaluate its practical implications in clinical diagnosis systems.

2. RESEARCH METHOD

2.1 Genetic Evolutionary Algorithm

One form of evolutionary algorithm is genetic algorithm. Genetic algorithms are heuristic search algorithms inspired by the process of biological evolution [6]. Genetic algorithms are widely used in various problems.

Genetic evolutionary algorithms (GAs) are optimization techniques inspired by the process of evolution in nature. They are metaheuristic algorithms that use the concepts of natural selection, genetic recombination, and mutation to find the best solution in a large and complex search space.

In genetic algorithms, the search process is conducted among a number of possible solutions, referred to as the population. Each individual in the population is represented as a chromosome, which is a potential solution. When the initial population is initialized, it is done randomly, while for subsequent populations, the population is generated through the evolution of chromosomes that have gone through a series of iterations called

generations. Each chromosome will go through an evaluation stage at each generation. This evaluation process uses a metric called fitness function, where the fitness value indicates the quality of the chromosomes in the population. The next generation, referred to as the offspring, is formed from the combination of the two previous chromosomes considered as parents, using the crossover operator [6]. The crossover operator is performed by swapping genes between two randomly selected parents. In addition to the crossover operator, there is also a mutation operator, which changes the value of a chromosome gene to the opposite value. For example, a value of 0 can change to 1, or vice versa [7]. The formation of a new generation population is done by selecting based on fitness value. In the development cycle of genetic algorithms to find the best solution, there are several stages involved. These stages are:

a) Chromosome Representation
   A chromosome is a representation of the solution to a particular problem. A chromosome consists of a series of genes that represent the relevant decision variables. The length of the chromosome is determined by the number of variables to be used in the study. The number of variables used in this study is 10. Thus, the total length of the chromosome is 10 genes. Chromosome representation is done in the form of real numbers with a range of values between 0 and 1.
b) Initialization Stage
   The initialization of the initial population is determined by the population size or popsize which determines the number of individuals in the population. A value is chosen to determine the length of the chromosome. Each chromosome contains genes with randomly selected values from the range 0-1. Each individual in the population then undergoes a reproduction stage involving crossover and mutation operations to generate fitness values.
c) Reproduction Stage
   - Crossover (*Extended Intermediate Crossover*) to get the offspring (child) value using Equations (1) and (2).
     $C_1 = P_1 + \alpha\,(P_2 - P_1)$ (1)
     $C_2 = P_2 + \alpha\,(P_1 - P_2)$ (2)
     P1 is the first parent and P2 is the randomly selected second parent. The value of α is a randomly selected constant in the interval [-0.25, 1.25].
   - Mutation (*Random* Mutation) is a mutation method performed by randomly selecting one parent and adding or subtracting the value of the selected gene with a small random number. In this study, the mutation method used is random mutation. Suppose the variable domain x_i is [min_i , max_i] and the resulting *offspring* is C = [x'_1 , ..., x'_n], then the value of the offspring gene can be obtained by Equation (3) as follows (Mahmudy, 2013):
     $x'_i = x_i + r\,(max_i - min_i)$ (3)
     The constant value *r is* chosen randomly, for example, in the range [-0.1, 0.1].
d) Evaluation Stage
   This evaluation process is intended to calculate the fitness value of each individual in the population, both parent and offspring. The fitness value obtained will be used in the next selection process. The fitness value can be calculated using Equation (4) below, using the Mean Squared Error (MSE) value obtained from the calculation process using the LDA method.
   $fitness = \frac{1}{MSE}$ (4)
e) Selection Stage
   In this research, a selection method known as *binary tournament selection* is used. This method is one of the selection techniques used in the system. It works by taking two individuals randomly from the population and offspring. Each individual is then compared based on its *fitness* value. The individual with the highest *fitness* value will be selected to continue to the next generation.

2.2 Linear Discriminant Analysis (LDA) Method

Linear discriminant analysis (LDA), is a generalization of Fisher's linear discriminant [8]. This method is used in statistics, pattern recognition, and machine learning to find linear combinations of features that distinguish two or more objects or events [9]. The resulting combinations are often used as linear classifiers or to reduce the dimensionality before classifiers are performed. The main goal of LDA is to classify objects into classes based on the features they represent. In LDA, objects have two variables: dependent variables and independent variables. The dependent variable is dependent on the attribute variable that describes the object [10].

The independent variables will be used to create a linear combination of these objects. LDA works by analyzing the dispersion matrix to find the optimal projection, allowing the input data to be projected onto a smaller dimensional space in order to best separate all patterns. In LDA, the dependent variable is the class of the object, often a nominal value, while the independent variables are features that describe the object, usually in the form of scalar values. Before predictions are made, LDA requires a training phase to determine the discriminant function. This involves using pre-classified objects and a number of feature variables. The steps of LDA training, as outlined by Ali, L, et.al. are as follows [11]:

1). Form a matrix of independent variables (X) and a vector of dependent variables (y).
2). Grouping the independent variable matrix into k groups, where k is the number of classes (dependent variable).
3). Calculating the average for each variable within each class.
4). Calculating the global average for each independent variable
5). Calculate the average corrected data of each class.
6). Calculate the covariance matrix for each class using the corresponding formula in Equation 5.

$$C_i = \frac{(X_i^0)^T X_i^0}{n_i} \quad (5)$$

Where, n_i is the number of rows in group X_i and $i=1,2,3,....,k$.

7). Calculate the within-group covariance matrix with Equation 6.

$$C(r,s) = \frac{1}{n}\sum_{i=1}^{g} n_i\, C_i(r,s) \quad (6)$$

8). Determine the inverse of the within-group covariance matrix.
9). Calculate the probability of the *i-th* class,

$$p_i = \frac{n_i}{N} \quad (7)$$

10). Calculate the number of classes for each data as in equation 8.

$$f_i(x_k) = \mu_i C^{-1} X_k^T - \frac{1}{2}\mu_i C^{-1}\mu_i^T + \ln(p_i) \quad (8)$$

11). Choose the value f so that each data will be classified into the index of the discriminant function.
12). Calculate the error value to be used in the MSE calculation with Equation (9).

$$E = t - y_k \quad (9)$$

13). Test all conditions.
14). Calculating the MSE (Mean Square Error) value to determine the level of accuracy in the prediction process. The MSE value will be used to calculate the fitness value in the genetic algorithm process. Following Equation (10) to get the MSE value:

$$MSE = \frac{\sum E^2}{n} \quad (10)$$

2.3 AFER Error Method

To assess whether a system is in accordance with the actual results, we can use the level of accuracy measured by the Average Forecasting Error Rate (AFER) method. AFER is used to evaluate the errors that occur in the forecasting system (Jilani et al., 2017). The AFER value can be calculated using Equation (11) (Lee et al., 2006).

$$AFER = \left(\frac{1}{n}\sum_{i=1}^{n}(|A_i - F_i| / A_i)\right) \times 100\% \quad (11)$$

AFER is the percentage difference between predicted data and actual data or reality. The level of accuracy is considered good if the resulting error value is smaller (Rahmadiani, 2012). There is a general rule to assess the quality of a forecasting system, where if the AFER error value is close to 0%, then the forecasting system is considered good. However, in practice, it is rare to find a

prediction case with an AFER error value that actually reaches 0%.

2.4 Combination of Genetic Evolutionary Algorithm and LDA

Genetic algorithms involve several stages. The first stage is parameter initialization, which involves setting inputs such as gene coding and chromosomes. Once the parameters are set, the next process is to generate the initial population, calculate the match value, perform crossover, perform mutation, and sort the chromosomes to produce the final solution in the form of the chromosome or individual with the highest match value, which is the best solution. In this study, the LDA algorithm is used during evaluation in the genetic evolution algorithm to obtain the match value.

2.5 Datasets

The study used the ILPD (Indian Liver Patient Dataset) dataset which includes 583 data entries [12]. Although there are no missing values in this dataset, there are 221 data entries that are duplicates. This dataset consists of 11 attributes, and the details of these attributes can be found in the following table:

i). Age: Age of the patient
ii). Gender: The gender of the patient
iii). TB: Total Bilirubin
iv). DB: Direct Bilirubin
v). Alkphos: Alkaline Phosphotase
vi). SGPT: Alamine Aminotransferase
vii). SGOT: Aspartate Aminotransferase
viii). TP: Total Protein
ix). ALB: Albumin
x). A/G Ratio: Albumin and Globulin Ratio
xi). Class: There are two classes (1 and 2)

3. RESULTS AND DISCUSSION

The tests included assessing various parameters such as the number of generations, population *size*, and the combination of crossover rate and *mutation* rate in the genetic algorithm. Each configuration was tested five times, and the average *fitness* value was calculated to select the best combination. Evaluation was also conducted on the model's ability to predict the test data using the AFER *error* metric.

3.1 *Generation Trial*

The first test conducted was on population size. In this testing process, several population sizes were input. The population sizes used are 20, 50, 100, 150, 200, 250, 300, 350, 400, 450 and 500. The parameter values used are popsize (population size) is 20, cr is 0.2 and mr is 0.1. The test results can be seen in the graph in Figure 1.

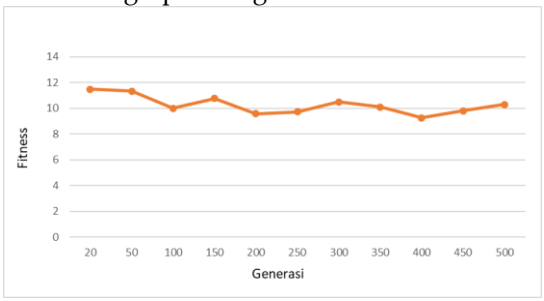

Figure 1. Generation Testing Chart

Based on Figure 1, it can be seen that the graph line displayed experiences up and down conditions. The lowest point is in generation 400 with an average fitness value of 9.263. While the highest average value is in generation of 20 with an average fitness value of 11.343. The number of generations parameter affects the average fitness value. The greater the generation value, the smaller the average fitness value. This is due to the generation of initial values in each individual randomly. In addition, it can also be caused by the use of *binary tournament selection* in the selection process.

3.2 *Population Size Trial*

The first test conducted was on population size. In this testing process, several population sizes were input. The population sizes used are 5, 10, 15, 20, 25, 30, 35, 40, 45 and 50. The parameter values used are generation of 20, cr is 0.2 and mr is 0.1. The test results can be seen in the graph in Figure 2.

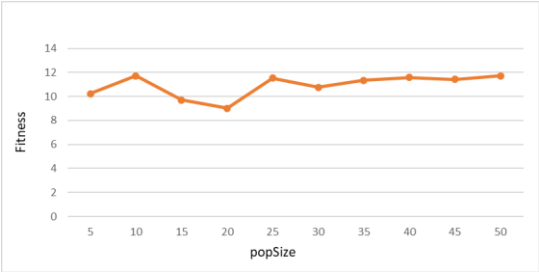

Figure 2. Population Size Testing Chart

Based on Figure 2, it can be seen that the graph does not change much after the population size reaches 30. Significant changes in fitness value occur when the population size is 5, 10, 15, and 20. The lowest point occurs when the population size is 20, with an average fitness of 9.0169. While the highest average value is achieved when the population size is 500, with an average fitness value of 11.7202. This change is likely due to the variation in randomly generated initial values for each individual, as well as the use of the binary tournament selection method. This selection method can lead to early convergence due to its nature of selecting the best individual from each tournament, which may reduce genetic variation in the population.

3.3 *Cr and Mr Combination Trial*

In the second trial, various combinations of crossover rate (cr) and mutation rate (mr) values were assessed. In this testing process, various combinations of cr and mr were inputted, with cr varying from 0.9 to 0.1 and mr varying from 0.1 to 0.9. Other parameter values used were generation of 20, and population size (popsize) of 20. The results of this test are represented in Figure 3.

Figure 3. Testing Graph of Cr and Mr Combination

Based on Figure 3, it can be seen that the graph shows significant fluctuations. The lowest point is found in the combination of cr 0.5 and mr 0.5, with an average fitness of 8.4073. Meanwhile, the highest average value occurs in the combination of cr 0.1 and mr 0.9, with an average fitness of 12.0012. From the test results, it can be seen that the combination of a smaller cr value than the mr value tends to produce a better solution than vice versa. There is a tendency that the greater the difference between cr and mr values, with cr being greater than mr, will result in a more optimal solution. However, this is not absolute, as there is a possibility that a smaller cr value than mr value can also produce an optimal solution, depending on the context and problem at hand.

3.4 *AFER Trial*

The fourth test aims to calculate the Average Forecast Error Rate (AFER) value on each test data, as an indicator of the accuracy of the prediction results against the actual target. Two sets of parameters were used in this test.

First, the parameters selected are from the previous test that produced the best fitness value. The results are as follows: generation 20, popsize 20, cr 0.2, and mr 0.1. With these parameter settings, the AFER value obtained is 2.1456% and the MAPE is 30.418%.

Second, the test was conducted by randomly selecting parameter values, with the results: generation 400, popsize 40, cr 0.9, and mr 0.1. In this setting, the AFER value obtained is 0.0345% and the MAPE is 16.576%.

Furthermore, by using the weights and biases generated from the combination process of genetic algorithm and LDA, the output value (y) is 0.479, while the target value is 0.5. So the difference between the target value and the prediction results of the system is 0.021.

4. CONCLUSION

The conclusion of the research on Optimizing Liver Disease Detection Through Combining Genetic Evolutionary Algorithms and Linear Discriminant Analysis (LDA) has been proven to produce more optimal predictions. This is reflected in the AFER,

MAPE and fitness values obtained. Genetic algorithms are applied by optimizing several parameters such as the number of generations, population size, crossover rate, and mutation rate. This research uses the extended intermediate crossover model and random mutation for the reproduction process, and selection using binary tournament selection.

The test results show that the best parameters produce an AFER value of 0.0345%. Since this AFER value is close to 0% on the test data prediction, the system can be considered good. While the MSE value is 16.576%. Where if the algorithm system made produces a MAPE value between 10%-20%, the ability of the forecasting system made is good. The best parameters chosen are generation of 400, population size of 40, crossover rate of 0.9, and mutation rate of 0.1.

ACKNOWLEDGEMENTS

Author thanks to all my Friends and college students at Tulungagung University who have supported me.

REFERENCES

[1] Dritsas, Elias, and Maria Trigka. 2023. "Supervised Machine Learning Models for Liver Disease Risk Prediction" Computers 12, no. 1: 19. https://doi.org/10.3390/computers12010019
[2] Bhupathi D, Tan CN-L, Tirumula SS and Ray SK, "Liver disease detection using machine learning techniques" in Proceedings of the 13th Annual CITRENZ Conference: Unifying Educational Delivery and Collaborating Towards Technical Excellence, 2022, https://mro.massey.ac.nz/items/60cdbe76-2c0b-4dc8-b21f-bed87ead6879.
[3] Wang, C., Wang, W. & Li, M. Regularized linear discriminant analysis via a new difference-of-convex algorithm with extrapolation. J Inequal Appl 2023, 90 (2023). https://doi.org/10.1186/s13660-023-03001-4.
[4] Setiawati, Intan, et. all. 2019. "Implementation of Decision tree to Diagnose Liver Disease" in JOISM: JOURNAL OF INFORMATION SYSTEM MANAGEMENT: Vol 1, No 1 Yogyakarta: Yogyakarta University of Technology.
[5] Handayani, Popon, et. all. 2019. "Liver Disease Prediction Using Decision tree and Neural Network Methods" in CESS (Journal of Computer Engineering System and Science): Vol. 4, No. 1. Jakarta: STMIK Nusa Mandiri Jakarta.
[6] Kusmadewi S, Purnomo H. (2005). "Solving Optimization Problems with Heuristic Techniques". Yogyakarta: Graha Ilmu
[7] Zamani, Adam Mizza, et al. (2012). "Implementation of Genetic Algorithm on Backpropagation Neural Network Structure for Breast Cancer Classification". Journal of POMITS Engineering. Volume 1, No. 1. 1-6
[8] Fisher, R.A. "The Use of Multiple Measurements in Taxonomic Problems." Annals of Eugenics, 1936.
[9] S. Kaya and M. Yağanoğlu, "An Example of Performance Comparison of Supervised Machine Learning Algorithms Before and After PCA and LDA Application: Breast Cancer Detection," 2020 Innovations in Intelligent Systems and Applications Conference (ASYU), Istanbul, Turkey, 2020, pp. 1-6, doi: 10.1109/ASYU50717.2020.9259883
[10] P. R. Kshirsagar, D. H. Reddy, M. Dhingra, D. Dhabliya and A. Gupta, "Detection of Liver Disease Using Machine Learning Approach," 2022 5th International Conference on Contemporary Computing and Informatics (IC3I), Uttar Pradesh, India, 2022, pp. 1824-1829, doi: 10.1109/IC3I56241.2022.10073425
[11] Ali, L., Wajahat, I., Amiri Golilarz, N. et al. LDA-GA-SVM: improved hepatocellular carcinoma prediction through dimensionality reduction and genetically optimized support vector machine. Neural Comput & Applic 33, 2783-2792 (2021). https://doi.org/10.1007/s00521-020-05157-2
[12] UCI Machine Learning Repository, https://doi.org/10.24432/C5D02C. Accessed on January 14, 2024.

BIOGRAPHIES OF AUTHORS

Dwi Ari Suryaningrum, S.Kom., M.Kom. Born in Tulungagung, June 14, 1995. Graduated S1 in 2017 at the Informatics Engineering S1 Study Program, Brawijaya University, Malang. Graduated S2 in 2020 at the Informatics Engineering Postgraduate Program, Sepuluh Nopember Institute of Technology, Surabaya. As a Lecturer in the Electrical Engineering Undergraduate Study Program at the Faculty of Engineering, Tulungagung University starting in 2021. Field of Programming and Artificial Intelligence. Email: dwiarisuryaningrum@unita.ac.id or dwiari.suryaningrum@gmail.com.

Muhammad Romadhoni Indra Firmansyah Born in Gresik, October 26, 2004. Student of Electrical Engineering S1 Study Program Tulungagung University. Email: masindrafirmansyah26@gmail.com.