LONTAR KOMPUTER VOL. 13, NO. 1 APRIL 2022 **p-ISSN 2088-1541**
DOI : 10.24843/LKJITI.2022.v13.i01.p06 **e-ISSN 2541-5832**
Accredited Sinta 2 by RISTEKDIKTI Decree No. 158/E/KPT/2021

Optimizing Random Forest using Genetic Algorithm for Heart Disease Classification

Parmonangan R. Togatorop[a1], Megawati Sianturi[a2], David Simamora[a3], Desriyani Silaen[a4]

[a]Faculty of Informatics and Electrical Engineering, Institute of Technology Del
Laguboti, Indonesia
[1]mona.togatorop@del.ac.id
[2]megawatiisianturii@gmail.com
[3]davidsimamora007@gmail.com
[4]desriyanisilaen17@gmail.com

Abstract

Heart disease is a leading cause of death worldwide, and the need for effective predictive systems is a major source of the need to treat affected patients. This study aimed to determine how to improve the accuracy of Random Forest in predicting and classifying heart disease. The experiments performed in this study were designed to select the most optimal parameters using an RF optimization technique using GA. The Genetic Algorithm (GA) is used to optimize RF parameters to predict and classify heart disease. Optimization of the Random Forest parameter using a genetic algorithm is carried out by using the Random Forest parameter as input for the initial population in the Genetic Algorithm. The Random Forest parameter undergoes a series of processes from the Genetic Algorithm: Selection, Crossover Rate, and Mutation Rate. The chromosome that has survived the evolution of the Genetic Algorithm is the best population or best parameter Random Forest. The best parameters are stored in the hall of fame module in the DEAP library and used for the classification process in Random Forest. The optimized RF parameters are max_depth, max_features, n_estimator, min_sample_leaf, and min_sample_leaf. The experimental process performed in RF uses the default parameters, random search, and grid search. Overall, the accuracy obtained for each experiment is the default parameter 82.5%, random search 82%, and grid search 83%. The RF+GA performance is 85.83%; this result is affected by the GA parameters are generations, population, crossover, and mutation. This shows that the Genetic Algorithm can be used to optimize the parameters of Random Forest.

Keywords: *Machine Learning, Random Forest (RF), Genetic Algorithm (GA), Default parameter, Random search, Grid search*

1. Introduction

Heart disease, or coronary heart disease, is one of the biggest causes of death globally. According to WHO (World Health Organization), in 2015, an estimated 8.8 million people died from heart disease; in the United Kingdom (UK), at least 2.3 million people suffered from heart disease, and in 2014 this condition contributed to at least 69,000 total deaths [1]. The key risk factors that affect a person with heart disease are High blood pressure, high cholesterol, and smoking. Many medical issues such as lifestyle choices, including diabetes, obesity, poor nutrition, physical inactivity, and excessive alcohol consumption, may also put people at a higher risk of heart disease [2].

Computer-aided detection (CAD) is designed to provide automated predictions of heart disease [2]. As one of the modern methods of computer-assisted detection, machine learning is an emerging technology to analyze medical data and provide a prognosis on early detection results. Different researchers use machine learning to diagnose heart disease to compare data mining tools and machine learning to classify heart disease using the Cleveland dataset from UCI Machine Learning [2] [3] [4].

Some researchers show that Random Forest (RF) accurately predicts heart disease because it

performs better. Research [5] compared random forest with KNN for predicting heart disease. The results obtained are RF achieving 95% accuracy compared to KNN achieving 73% accuracy. Therefore, predictions made by RF are better than KNN [5]. Sravanthi [6] compared the classification methods of RF, decision tree, artificial neural network, SVM, Naive Bayes, and KNN on coronary datasets. Accuracy results obtained were 0.88%, 0.87%, 0.86%, 0.83%, 81% and 0.77% respectively. RF has the highest accuracy. Besides predicting heart disease, RF is also used for other domains, such as forecasting new students [7]. Based on previous research, it was shown that RF has better accuracy and performance than other algorithms, so in this study, the Random Forest algorithm will be used to perform classification.

Optimizing RF parameters can improve the accuracy of the prediction model [8] [9]. RF involves several hyperparameters controlling the structure of each tree, structure, size, and randomness of the forest [10]. Grid Search and Random Search can automatically find the optimal hyperparameter in RF. Grid Search is an optimization algorithm that searches all possible combinations in the search space [9]. Random Search [11] is an approach that randomly samples parameters defined by search space.

Meanwhile, the Genetic Algorithm (GA) is one of the best-known machine learning algorithms for solving optimization problems [12] and gives the optimal value of a function. Genetic algorithms (GA) is an optimization strategy inspired by evolution. GA work by adopting the evolutionary process on a population of solutions [13]. GA is already used to solve various optimization cases. Currently, many researchers are using a GA to optimize the RF hyperparameter [9] [12] [14] [15]. Research [16] has conducted a literature study on the use of GA for heart disease and concluded that the use of GA achieved an accuracy of up to 97.7%. Results show that GA can be used to optimize RF parameters.

This research's novelty is optimizing RF hyperparameter for heart disease Classification. Due to the ability of GA to perform optimization, GA will be used as an optimization algorithm to optimize RF parameters. After that, the optimization results will be compared with Grid Search and Random Search. The purpose of using GA is to get an optimized hyperparameter and produce higher accuracy for Heart Disease Classification. The result is that using Random Forest with genetic algorithms has higher accuracy than using only Random Forest.

2. Method

The method designed to implement random forest optimization using a genetic algorithm consists of several stages. The design begins with the data preprocessing process, namely data merging, data cleaning to clean the data, and data reduction to remove features with high missing values. After doing the preprocessing stage, it will proceed to one of the two processes that have been passed. The Random Forest classification process is intended for Random Forest classification without going through an optimization process. The second process is the Random Forest optimization process using the Genetic Algorithm. This process produces the best parameters, which will be classified again using Random Forest. The classification results from both approaches will be evaluated. The research method used to optimize random forest parameters using genetic algorithms can be seen in Figure 1.

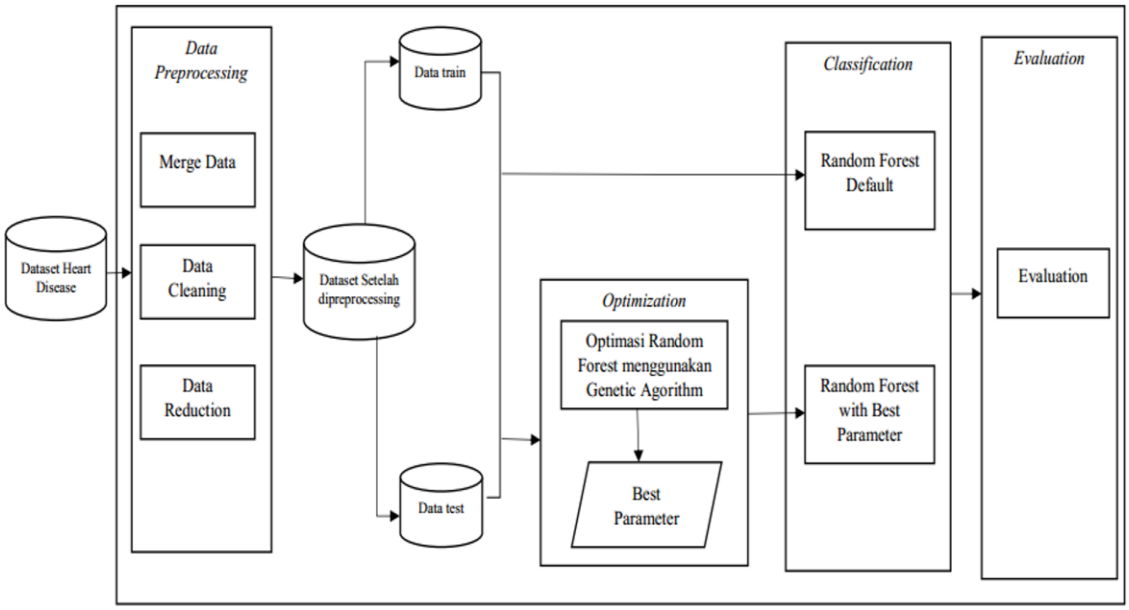

Figure 1. Design system optimizing Random Forest Using Genetic Algorithm

2.1. Data Preprocessing

The dataset used in this research is a dataset taken from the UCI Machine Learning website [17]. The dataset has 13 attributes: attributes sex, fbs, exang, and target with binary data type; attributes cp, restecg, slope, ca, and thal with categorical data type; attributes age, trestbps, chol, thalach, and oldpeak with continu data type.

Data preprocessing used in this research are data cleaning, data integration, and data reduction. Data preprocessing needs to be done to ensure the quality of the data used. The quality of the data decreases if the data obtained is incomplete, inconsistent, and contains special characters that are not needed. The Cleveland dataset and the Hungarian dataset were merged in data integration because the dataset has the same features. The Heart Disease dataset is data that is not too large and complex. When this dataset is put together, it creates 596 rows and 14 attributes. The percentage of missing values can be shown in Table 1.

Table 1. Missing value of datasets

No	Atribut	Total missing value
1	Ca	49.41%
2	Thal	44.39%
3	Slope	31.83%
4	Chol	3.85%
5	Fbs	1.34%
6	Exang	0.17%
7	Thalach	0.17%
8	Restecg	0.17%
9	Trestbps	0.17%

Data cleaning helps in the process of overcoming missing values, data inconsistencies, and detecting outliers. To overcome this, a preprocessing technique was carried out to see the number of missing values contained in the dataset; for continuous attributes, the missing values will be handled by their mean. Meanwhile, the categorical attributes will be input with '0'. Due to the three attributes we have dropped, the final data result is 596 rows, and 11 attributes are used for making machine learning models. The distribution of training and testing data states that the data splitting process uses the train_test_split library, with 80:20 data partitions, random_state is used to ensure that each run splitting the data will always be the same.

2.2. Random Forest

Random Forest is the most popular ensemble technique for probability prediction and estimation. The ensemble method is a way to improve the accuracy of the classification method by combining classification methods [18]. Random Forest uses a decision tree as a basic classification method; this Random Forest ensemble method is used for classification and regression purposes or often referred to as CART (Classification and Regression Technique), which consists of several classifiers that have been trained where the predictors will be combined and classify the sample that has been selected [19]. Random Forest is a general term used as an aggregation scheme in a decision tree. Before it was called a random forest, this Algorithm was named Breimen Forest because Breiman proposed it. Mathematically the calculation of Breiman Forest can be expressed as:

$$m_{M,n}(x, \theta_1, \dots, \theta_m, \delta_n) = \frac{1}{M} \sum_{m=1}^{M} m_n(x, \theta_m, \delta_n) \quad (1)$$

Random Forest is a collection of randomized trees that will be averaged. The above formula states that m_n is a random forest so that m_(M,n) is a random forest that you want to create with M randomized tree, with x stating the predicted value at the x-th tree, where, θ_1,…,θ_m is a random variable distributed with sample data _n. M expresses a randomized tree. So the output of Breiman Forest is the average prediction given by M trees.

2.3. Genetic Algorithm

The Genetic Algorithm (GA) is based on the principle of natural selection. Holland developed the genetic Algorithm as a helpful tool for search and optimization problems. The Genetic Algorithm is applied to a population of individuals P where individuals are categorized by chromosome Ck = (1,…, P). Chromosomes consist of several strings of symbols, known as genes Ck = Ck1,….., Ckn, and we can write N as the length of the string. Individuals are evaluated based on their respective fitness functions. Genetic Algorithms operate with three basic operators: selection, crossover, and mutation. Selection plays a role in selecting individuals with the best fitness values from the current generation to survive in the next generation. A crossover is a process of combining two parents to produce children. The mutation function is to make small changes to certain gene elements from the population and provide more ability to produce problem solutions optimization [20]. The genetic Algorithm looks for the best optimal solution during the evolution of chromosomes in terms of a defined fitness function [21].

The parameters used in the genetic Algorithm are the fitness function, the population size in each generation, the probability of crossover, the probability of mutation, and the number of generations formed. The following are the basic steps in the Genetic Algorithm.

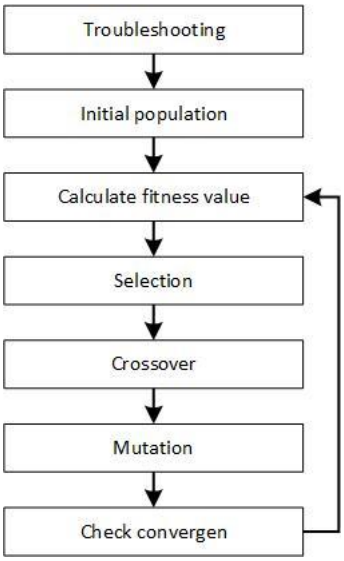

Figure 2 Basic steps of Genetic Algorithm

Figure 2 shows the steps for the genetic algorithm process. First, initialize the population that designs a chromosome to represent the solution. Usually designed in the form of a binary string. After generating the initial population, genetic operators (selection, crossover, mutation) are applied to that population. The selection operator selects the most suitable chromosome by evaluating the fitness value of each chromosome. In general, accuracy is used as a fitness function for classification problems. Then the crossover operator swaps the genes of the two-parent chromosomes to get a new child to reach a better solution. The mutation operator replaces randomly selected bits with very low probability. By applying this operator, a new population is formed. The above step is a step to create a new population and is carried out until the stopping condition is met [22].

2.4. Classification Task: Random Forest Algorithm

The Random Forest Algorithm has several hyperparameters that can affect performance. Hyperparameters are parameters needed by machine learning methods to classify. Choosing the correct parameters can make a significant difference in the prediction results. Specifying this hyperparameter can be done manually by trying all possible values. However, doing so is time-consuming because the number of possible combinations is very large.

This study will conduct experiments on classifying random forests with default parameters, random search, grid search, and Genetic Algorithm. Random Search and Grid Search are used to see the performance of another optimization method without using a Genetic Algorithm.

a. Default Parameter: When running the Random Forest algorithm, RF has parameters used to build the model. These parameters have their respective default values. Random Forest with default parameters also has good accuracy compared to other classification algorithms such as decision trees, Naïve Bayes, etc. This study examines the effect of RF parameters, including max_features, max_depth, n_estimator, min_sample_split, and min_sample_leaf. To determine the possible values of these parameters. The possible values obtained for each parameter are contained in Table 2.

Table 2. Parameters and possible value of Random Forest

Parameter	Possible Value
Max_features	'sqrt', 'log2'
Max_depth	2, 5, 10, 20, 50, None
N_estimator	100 – 1000 (interval 100)
Min_sample_split	2 – 5
Min_sample_leaf	1 – 4

b. Random search: The strategy widely used to perform hyperparameter tuning is Random Search. Random search works by searching for every possibility in the parameter. Based on research [23] states that Grid Search has advantages when browsing a search space that is too large, while Random Search does not always produce good results. Research [5] performed hyperparameter optimization using Random Search and got higher results than the default parameters. Study [24] states that Random Search (RS) has advantages in multidimensional hyperparameters compared to grid search. The Random Search will return the best parameter by its process and do the classification.
c. Grid search: Grid Search (GS) is one of the most commonly used methods for exploring the hyperparameter configuration space. The main disadvantage of GS is that when the configuration space is relatively high, GS is not efficient because the number of evaluations increases exponentially, so it requires a long computation time. Research [23] uses grid search for hyperparameter optimization because of its simplicity in implementation and parallelization and its reliability in low-dimensional space. Study [25] proposed system helps set hyperparameters using the grid search method. Based on the experiments, the Algorithm with the grid search hyperparameter setting gives more accurate results than the traditional approach (without setting the hyperparameter). The Grid Search will return the best parameter by its process and do the classification. And then, the three methods: Default Parameter, Random Search, and Grid Search, will be evaluated to see the model's accuracy. Grid Search and Random Search will be implemented using python's scikitlearn library.

2.5. Proposed Method: RF-GA Optimization

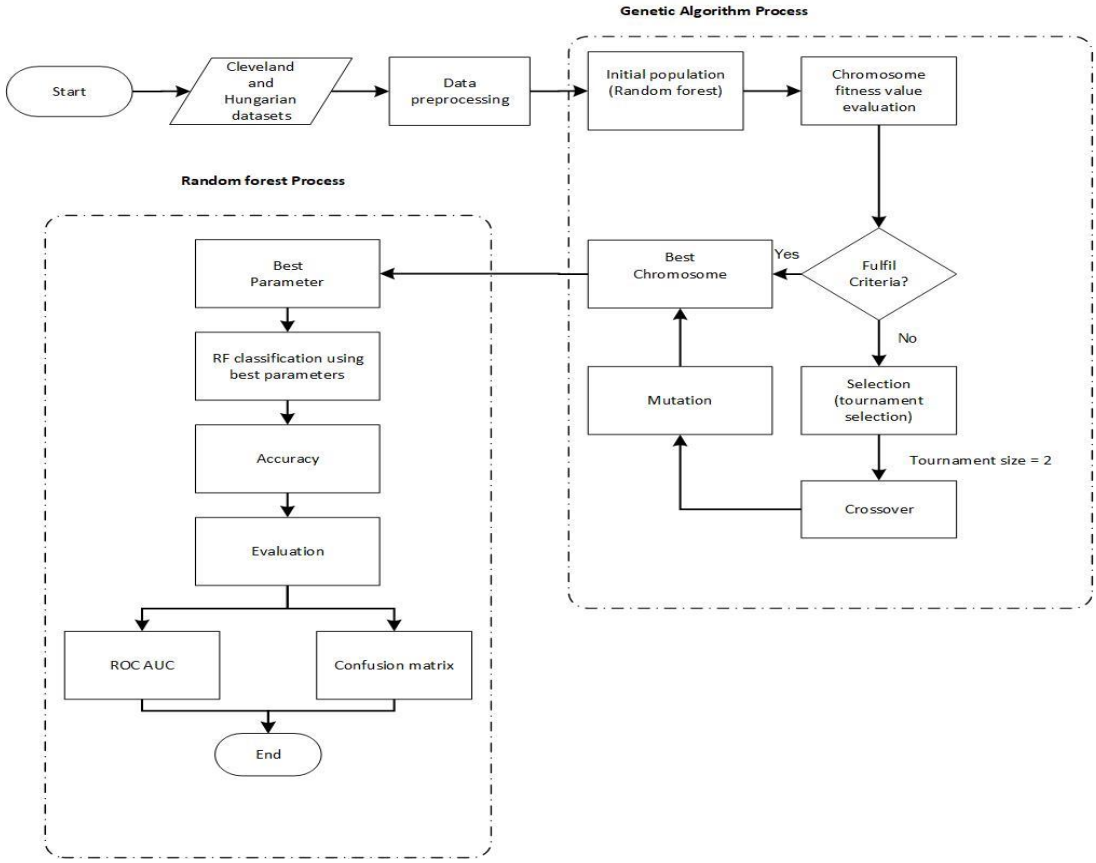

Figure 3. RF-GA Optimization

Genetic algorithms are used before the classification process to improve the results of random forest classification. RF-GA Optimization: Random Forest-Genetic Algorithm Optimization is the proposed optimization method for this research. RF-GA optimization can be seen in Figure 3.

Optimizing RF with GA begins by entering heart disease data and then preprocessing data on the dataset. The Genetic Algorithm performs to initialize the initial population (define chromosomes). Chromosomes are parameters of the machine learning algorithm used in this study, which is Random Forest. Parameters in random forests that are optimized and become population initialized in the genetic Algorithm are max_depth, max features, min_sample_leaf, min_sample_split, and n_estimators. Evaluate the fitness value for each chromosome to ensure that the chromosome criteria are suitable for selection. This study is a classification of heart disease, the purpose of the classification is to predict whether a person has heart disease or not.

For this reason, the fitness score used in this study is the AUC score. AUC Score is one of the fitness value metrics evaluations. If the fitness value meets the criteria, it is selected to be the best chromosome for the genetic algorithm optimization process. However, if it does not meet, the selection process is carried out using the tournament size two times. Crossover to swap genes from two-parent chromosomes to get a new child to achieve a better solution. Mutations to make small changes in specific gene elements of the population by randomly selecting genes with very low probability and replacing them. This process produces the best chromosome, obtained as the best parameter. When running a genetic algorithm with parameters such as crossover_probabilty, mutation_probability, population_size, and number_of_generations, the algorithm module from DEAP will be used to execute the Evolutionary Algorithm. One of the parameters required from the algorithm module is the HallOfFame() module. The Genetic Algorithm process will be stored in a list by HallOfFame(), which contains the best individuals who survive after going through the evolution process in the form of best_parameters. These best_parameters are random forest parameters that a genetic algorithm has optimized. Furthermore, classification is carried out using optimized parameters (best_parameters).

In the Random Forest Process, the best parameters obtained from the Genetic Algorithm process are classified using Random Forest. Accuracy results are evaluated using the Confusion Matrix to measure the performance of the classification and determine the level of obtaining precision, accuracy, and error values. The ROC-AUC evaluation technique will describe an accuracy improvement curve and obtain a final score of accuracy.

DEAP is built using python and can be used to perform computational calculations for researchers who want to use Genetic Programming. DEAP provides the essentials for assembling advanced Evolutionary Computation (EC) systems. The aim is to provide a practical tool for rapid prototyping of custom evolution algorithms, where every step of the process is as straightforward as possible and easy to read and understand.

DEAP provides basic data structures, genetic operators, and basic examples for users to implement evolutionary loops [26]. DEAP consists of two basic structures: the creator and toolbox modules. The creator module allows the generation of genotypes and populations from any data structure. The creator module is the key to facilitating the implementation of all evolutionary algorithms, including Genetic Algorithms, genetic programming, evolution strategies, and others.

2.6. Evaluation Method

The evaluation methods that will be used to test the performance of the classification model are Confusion Matrix and ROC AUC. The confusion matrix is used to obtain the accuracy of the classification performed on the Algorithm. The classification process's accuracy value is obtained in the confusion matrix. In measuring performance using the confusion matrix, there are four terms used, namely: True Positive (TP), True Negative (TN), False Negative (FN), and False Positive (FP). The confusion matrix results measure performance metrics, often called evaluation matrices. The evaluation metrics used are classification accuracy, classification error, precision, and recall. Classification accuracy is used to display the accuracy obtained from the evaluation results. Classification error is used to display the number of errors or errors in the evaluated data. Precision is used to describe a measure of the accuracy of the evaluation. The recall is used to describe the success of the accuracy obtained.

$$\text{Accuracy} = \frac{(\text{TN}+\text{TP})}{(\text{TN}+\text{FP}+\text{FN}+\text{TP})} \quad (2)$$

$$\text{Error rate} = \frac{(FP+FN)}{(TP+TN+FP+FN)} \quad (3)$$

$$\text{Precision} = \frac{(TP+TN)}{(TP+TN+FP+FN)} \quad (4)$$

$$\text{Recall} = \frac{TP}{TP+FN} \quad (5)$$

A better classification model is a model that has a larger ROC curve. The results of the ROC curve show the visualization of the accuracy of the model and comparison between classification models based on their True Positive Rate(TPR) and False Positive Rate(FPR) [27]. The AUC Score is also used to test the performance of the model.AUC (Area Under the Curve) closer to 1 would be able to ideally differentiate the two classes in the case of binary classification [28].

3. Result and Discussion

3.1. Result

The proposed work performs four experiment models: Random Forest with default Parameter, Grid Search, Random Search, and RF + GA. Performance measures are calculated and compared, as mentioned in the evaluation section.

In RF+GA, we do some research to see the best parameters of GA like generations, population, crossover rate, and mutation rate. This study compares the classification results based on the RF with the default parameter, Random Search, Grid Search, and RF +GA. The result of the experiment shows in these figures.

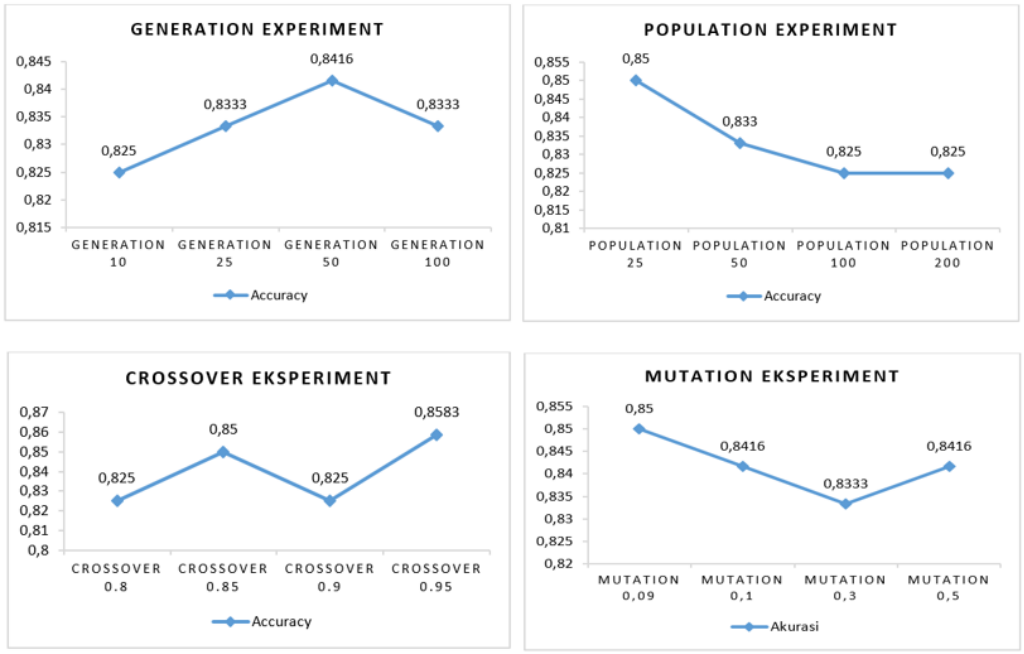

Figure 4. Parameter GA Experiment

From Figure 4, we can state that the best parameters for GA to produce a better result are generation 50, population 25, Crossover 0.95, and Mutation 0.09. In figure 4, the blue line (the value of the axis) is the accuracy value of each experiment.

Table 3. Parameters used in the experiment

Experiment	Max_ depth	Max_ features	Min_ sample_leaf	Min_ sample_split	N_estimators
Default parameter	None	auto	1	2	100
Grid search	5	log2	1	2	300
Random search	2	sqrt	2	5	100
RF+GA	2	sqrt	4	5	100

The four classification methods use the same training and testing samples to maintain the comparability of the result. Table 3 shows the parameters used for each classification. Random Forest parameters max_depth, max features, min_sample_leaf, min_sample_split, and n_estimators will be optimized to achieve optimal results with the Genetic Algorithm. This value is obtained from the literature review of similar research.

Table 4. Experiment Results

Experiment	Accuracy	Error	Precision	Recall	AUC
Default parameter	0.825	0.175	0.8534	0.8919	0.79
Grid search	0.8333	0.1667	0.8661	0.8642	0.82
Random search	0.8167	0.1833	0.8734	0.8519	0.81
RF + GA	0.8583	0.1417	0.8861	0.8974	0.84

The Accuracy of the AUC Score for RF with default Parameter, Grid Search, Random Search, and RF + GA are illustrated in Table 4. It can be observed that the Accuracy and AUC scores of RF + GA come out to be more than Default Parameter, Random Search, and Grid Search. Based on the table, the best evaluation metrics experiment is Grid Search.

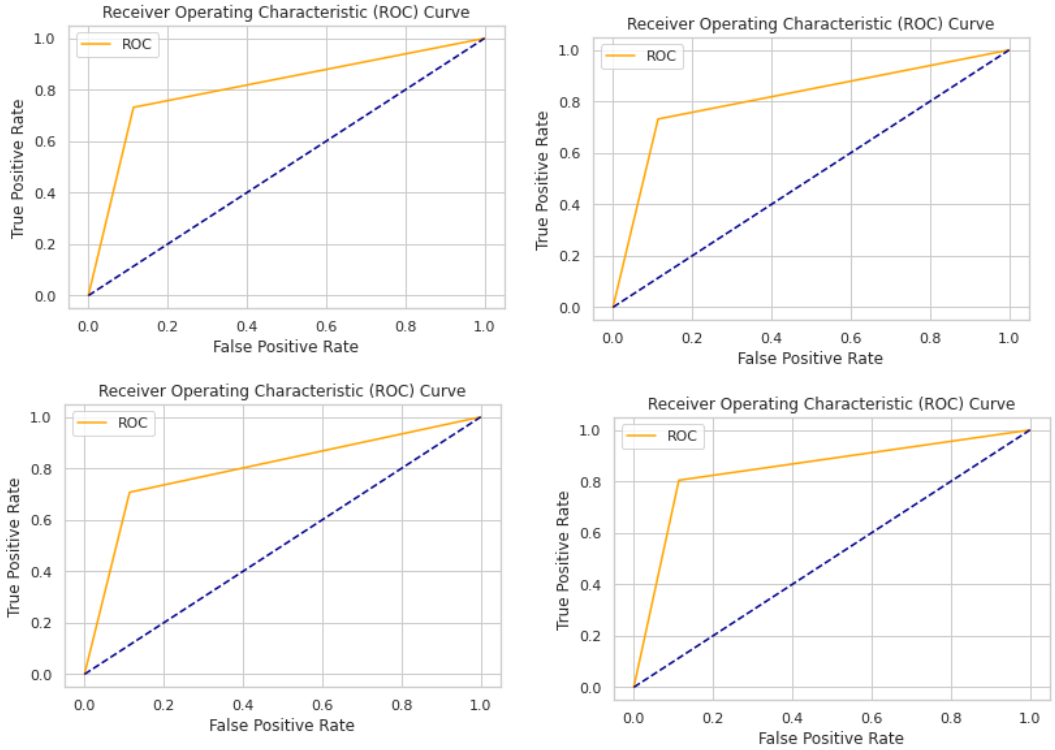

Figure 5. ROC Curve of (a) Default Parameter (b) Random Search (c) Grid Search (d) RF+GA

Figures 5 compares the ROC curves for RF with Default Parameter, Grid Search, Random Search, and RF + GA. The curve observation states that RF + GA is more suitable for the prediction model since the AUC and the graphic are closer to 1.

Table 4 shows the result considering different performance measures such as Accuracy, Error, Precision, and Recall. From the performance measure, we can state that RF + GA outperforms the other Algorithm to predict heart disease.

3.2. Discussion

In the Random Forest optimization experiment using Genetic Algorithm (RF + GA), the authors conclude that GA can be used to optimize the parameters of Random Forest and produce better accuracy than Grid Search. The search space used by GA and Grid Search is also the same through the initial population input in GA. The performance of the GA is also influenced by the parameters that exist in the GA, including Generation, Population, Crossover Rate, and Mutation Rate. Accordingly, the experimental results can be analyzed as follows:

a. The number of Generations is not directly proportional to accuracy. We conclude that the generation parameter will provide the optimum solution for a particular generation so that the GA will stop searching when it has obtained the optimal solution, which can be referred to as termination criteria.
b. The number of small populations produces better accuracy than large populations, and we conclude that this is influenced by the dataset and search space performed by GA, the search space that is not too large makes GA not need a larger population to search. However, if the search space is large, we assume that GA will require a larger population to produce a more optimum solution
c. The experimental results show that the crossover with the highest value and the mutation with the lowest value provides better accuracy and obtains the optimum solution.

In the Random Forest Experiment, experiments have been carried out using default parameters, Random Search, and Grid Search. The experimental results show that parameter optimization using Grid Search can increase accuracy, while experiments using Random Search experience a decrease compared to the default parameters.
The result of the analysis of the relationship between input parameters and RF classification accuracies are as follows:

a. In some cases, a high number of n_estimators can produce good accuracy, but using the default value=100 can also produce more optimal accuracy.
b. The higher the max_depth value, the higher the observation probability so that it can improve the model's capabilities.
c. Using max_features = sqrt(n) tends to produce a better model than auto and sqrt. But it is possible to use max_features = log2(n) to produce a good solution as in Grid Search.
d. Using min_sample_split and min_sample_leaf with higher values tends to produce a better result.

4. Conclusion

Random Forest is one of the classifying algorithms of machine learning. One application of the classification algorithm is Heart Disease Classification. There are several classification algorithms, including Random Forest. Random Forest is an algorithm that produces good results when classifying. Random Forest has parameters that are used to build a classification model. This research focuses on GA, which is used to optimize five parameters on RF, namely n_estimator, max_depth, max_feature, min_sample_split, and min_sample_leaf, to produce optimal heart disease classification accuracy. Optimization of the Random Forest parameter using a genetic algorithm is carried out by using the Random Forest parameter as input for the initial population in the Genetic Algorithm. The Random Forest parameter undergoes a series of processes from the Genetic Algorithm: Selection, Crossover Rate, and Mutation Rate.

Based on the experiments conducted, the performance of the Random Forest classification with Default Parameters 82.5%, Random Search 82%, and Grid Search 83% shows that parameter optimization using Grid Search can improve accuracy, while experiments using random search experience problems. The performance of RF + GA classification reaches 85.83%; this is influenced by the parameters in the Genetic Algorithm, including Generation, Population,

Crossover Rate, and Mutation Rate. Therefore, it can be concluded that Genetic Algorithms can be used to optimize the parameters of Random Forest and increase the accuracy of Random Forest results.

Further, as an extension of this work, a bigger dataset is required to obtain a better training model, using other optimization algorithms to see the difference in the performance of the Genetic Algorithm with other algorithms for Heart Disease Classification.

References

[1] L. Anderson *et al.*, "Patient education in the management of coronary heart disease," *Cochrane Database Syst Rev.*, vol. 2017, no. 6, 2017, doi: 10.1002/14651858.CD008895.pub3.
[2] K. H. Miao, J. H. Miao, and G. J. Miao, "Diagnosing Coronary Heart Disease using Ensemble Machine Learning," *International Journal of Advanced Computer Science and Applications(IJACSA)*, vol. 7, no. 10, pp. 30–39, 2016, doi: 10.14569/ijacsa.2016.071004.
[3] I. Tougui, A. Jilbab, and J. El Mhamdi, "Heart disease classification using data mining tools and machine learning techniques," *Health and Technology*, vol. 10, no. 5, pp. 1137–1144, 2020, doi: 10.1007/s12553-020-00438-1.
[4] N. B. Muppalaneni, M. Ma, and S. Gurumoorthy, *Soft Computing and Medical Bioinformatics*. Springer Singapore, 2019. doi: 10.1007/978-981-13-0059-2.
[5] H. Kaur and D. Gupta, "Human Heart Disease Prediction System Using Random Forest Technique," *International Journal of Computer Science and Engineering*, vol. 6, no. 7, pp. 634–640, 2018.
[6] P. V. S. N. Sravanthi and P. Rajesh, "An exploration of prediction of heart disease using machine learning classification," *International Journal Scientific & Technology Research*, vol. 9, no. 3, pp. 6817–6824, 2020.
[7] R. R. Waliyansyah and N. D. Saputro, "Forecasting New Student Candidates Using the Random Forest Method," *Lontar Komputer Jurnal Ilmiah Teknologi Informasi*, vol. 11, no. 1, p. 44, 2020, doi: 10.24843/lkjiti.2020.v11.i01.p05.
[8] I. Syarif, A. Prugel-Bennett, and G. Wills, "SVM Parameter Optimization using Grid Search and Genetic Algorithm to Improve Classification Performance," *TELKOMNIKA (Telecommunication Computing Electronics and Control*, vol. 14, no. 4, p. 1502, 2016, doi: 10.12928/telkomnika.v14i4.3956.
[9] A. S. Wicaksono and A. A. Supianto, "Hyperparameter optimization using genetic algorithm on machine learning methods for online news popularity prediction," *International Journal of Advanced Computing Science and Application*, vol. 9, no. 12, pp. 263–267, 2018, doi: 10.14569/IJACSA.2018.091238.
[10] P. Probst, M. N. Wright, and A. L. Boulesteix, "Hyperparameters and tuning strategies for random forest," *Wiley Interdisciplinary Reviews Data Mining and Knowledge Discovery*, vol. 9, no. 3, 2019, doi: 10.1002/widm.1301.
[11] R. Schaer, H. Müller, and A. Depeursinge, "Optimized distributed hyperparameter search and simulation for lung texture classification in CT using Hadoop," *Journal of Imaging*, vol. 2, no. 2, 2016, doi: 10.3390/jimaging2020019.
[12] D. Ming, T. Zhou, M. Wang, and T. Tan, "Land cover classification using random forest with genetic algorithm-based parameter optimization," *Journal of Applied Remote Sensing*, vol. 10, no. 3, p. 035021, 2016, doi: 10.1117/1.jrs.10.035021.
[13] G. Rivera, L. Cisneros, P. Sánchez-Solís, N. Rangel-Valdez, and J. Rodas-Osollo, "Genetic algorithm for scheduling optimization considering heterogeneous containers: A real-world case study," *Axioms*, vol. 9, no. 1, 2020, doi: 10.3390/axioms9010027.
[14] N. K. Kumar, D. Vigneswari, M. V. Krishna, and G. V. P. Reddy, "An Optimized Random Forest Classifier for Diabetes Mellitus", *Emerging Technologies in Data Mining and Information Security*, doi: 10.1007/978-981-13-1498-8.
[15] S. S. Shah and M. A. Pradhan, "R-Ga: an Efficient Method for Predictive Modeling of Medical Data Using a Combined Approach of Random Forests and Genetic Algorithm," *ICTACT Journal on Soft Computing*, vol. 06, no. 02, pp. 1153–1156, 2016, doi: 10.21917/ijsc.2016.0160.
[16] M. D. Yudianto, T. M. Fahrudin, and A. Nugroho, "A Feature-Driven Decision Support System

for Heart Disease Prediction Based on Fisher's Discriminant Ratio and Backpropagation Algorithm," *Lontar Komputer Journal Ilmiah Teknologi Informasi*, vol. 11, no. 2, p. 65, 2020, doi: 10.24843/lkjiti.2020.v11.i02.p01.

[17] "Heart Disease Data Set." https://archive.ics.uci.edu/ml/datasets/heart+disease (accessed Apr. 01, 2021).

[18] A. Syukron and A. Subekti, "Penerapan Metode Random Over-Under Sampling dan Random Forest Untuk Klasifikasi Penilaian Kredit," *Jurnal Informatika*, vol. 5, no. 2, pp. 175–185, 2018, doi: 10.31311/ji.v5i2.4158.

[19] E. Goel and E. Abhilasha, "Random Forest: A Review," *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 7, no. 1, pp. 251–257, 2017, doi: 10.23956/ijarcsse/v7i1/01113.

[20] S. Kumar and G. Sahoo, "A random forest classifier based on genetic algorithm for cardiovascular diseases diagnosis," *International Journal of Engineering Transaction B: Application*, vol. 30, no. 11, pp. 1723–1729, 2017, doi: 10.5829/ije.2017.30.11b.13.

[21] S. M. Elsayed, R. A. Sarker, and D. L. Essam, "A new genetic algorithm for solving optimization problems," *Engineering Application of Artificial Intelligence*, vol. 27, pp. 57–69, 2014, doi: 10.1016/j.engappai.2013.09.013.

[22] K. Kim, K. Lee, and H. Ahn, "Predicting corporate financial sustainability using Novel Business Analytics," *Sustainability*, vol. 11, no. 1, pp. 1–17, 2018, doi: 10.3390/su11010064.

[23] J. Emakhu, S. Shrestha, and S. Arslanturk, "Prediction system for heart disease based on ensemble classifiers," *Proceedings of the 5th International Conference on Industrial Engineering and Operations Management*, no. August, pp. 2337–2347, 2020.

[24] C. G. Siji George and B. Sumathi, "Grid search tuning of hyperparameters in random forest classifier for customer feedback sentiment prediction," *International Journal of Advanced Computer Science and Applications(IJACSA)*, vol. 11, no. 9, pp. 173–178, 2020, doi: 10.14569/IJACSA.2020.0110920.

[25] P. Liashchynskyi and P. Liashchynskyi, "Grid Search, Random Search, Genetic Algorithm: A Big Comparison for NAS," no. 2017, pp. 1–11, 2019.

[26] J. Kim and S. Yoo, "Software review: DEAP (Distributed Evolutionary Algorithm in Python) library," *Genetic Programming and Evolvable Machines*, vol. 20, no. 1, pp. 139–142, 2019, doi: 10.1007/s10710-018-9341-4.

[27] D. Krishnani, A. Kumari, A. Dewangan, A. Singh, and N. S. Naik, "Prediction of Coronary Heart Disease using Supervised Machine Learning Algorithms," *IEEE Region 10 Annual International Conference Proceedings/TENCON*, vol. 2019-Octob, pp. 367–372, 2019, doi: 10.1109/TENCON.2019.8929434.

[28] E. K. Hashi and Md. Shahid Uz Zaman, "Developing a Hyperparameter Tuning Based Machine Learning Approach of Heart Disease Prediction," *Journal of Applied Science & Process Engineering*, vol. 7, no. 2, pp. 631–647, 2020, doi: 10.33736/jaspe.2639.2020.

[29] P. T. Nguyen, N. B. Vu, L. Van Nguyen, L. P. Le, and K. D. Vo, "The Application of Fuzzy Analytic Hierarchy Process (F-AHP) in Engineering Project Management," *2018 IEEE 5th International Conference Engineering Technologies Applied Science (ICETAS) 2018*, pp. 1–4, 2019, doi: 10.1109/ICETAS.2018.8629217.