**Indonesian Journal of Electrical Engineering and Computer Science**
Vol. 35, No. 3, September 2024, pp. 1976~1985
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v35.i3.pp1976-1985

# Microarray classification using genetic algorithm and latin hypercube sampling

**Bangun Rizki Awangditama, Nanik Suciati**
Department of Informatics, Faculty of Intelligent Electrical and Informatics Technology,
Institut Teknologi Sepuluh Nopember (ITS), Surabaya, Indonesia

**Article Info**

***Article history:***

Received Feb 23, 2024
Revised May 4, 2024
Accepted Apr 12, 2024

***Keywords:***

Classification
Feature selection
Genetic algorithm
Latin hypercube sampling
Microarray

## ABSTRACT

Cancer, the second leading cause of global death, requires advanced diagnostic technology. Microarray gene expression technology plays an important role in comprehensively analyzing the genetic aspects of cancer. However, challenges such as high-dimensional attributes, limited samples, and varying gene presence rates hinder the accurate classification of microarray data. This study proposes a model that uses latin hypercube sampling (LHS) in genetic algorithms (GA) for Feature Selection in microarray data classification. LHS makes the chromosome samples in the initial population of GAs representative and diverse. The study used three microarray datasets with different numbers of features and classes. The results reveal that first, the use of GA alone tends to limit the exploration of the resulting feature space, while the use of LHS can expand the feature selection possibilities in the context of feature selection. Secondly, this study shows that microarray classification using GA with LHS (GALHS) consistently outperforms other feature selection methods such as based correlation features (BCF), principal component analysis (PCA), relief, and lasso. Thus, this research contributes to feature selection by applying LHS and GA to optimize the performance of microarray data classification models.

***Corresponding Author:***

Nanik Suciati
Department of Informatics, Faculty of Intelligent Electrical and Informatics Technology
Institut Teknologi Sepuluh Nopember (ITS)
Surabaya, Indonesia
Email: nanik@if.its.ac.id

## 1. INTRODUCTION

Cancer is the second leading cause of death worldwide. According to data from the World Health Organization (WHO), cancer causes one in six deaths globally. Amidst the increasing incidence of cancer, there is a need for ways that can help in diagnosing cancer accurately [1]. One approach that has been applied is the use of microarray methods to analyze gene expression in the context of cancer [2]. Microarray methods allow simultaneous monitoring of thousands of genes in a single experiment and generate a matrix of gene expression data from multiple samples. The advantage of this method lies in its ability to provide extensive genetic information, allowing in-depth analysis of gene expression profiles and changes in deoxyribonucleic acid (DNA). Using DNA microarray technology, medical professionals can effectively monitor thousands of DNA sequences in a single experiment [3].

Microarray data typically features high-dimensional data, limited sample sizes, and varying levels of gene presence among different samples [4]. Sorting and categorizing genes based on specific criteria in a dataset containing millions of genes can be a challenging task [5], [6]. In this context, the application of machine learning for classifying microarray data can effectively address this challenge [7], [8].

The process of microarray data classification involves several important stages. In the initial stage, dataset selection is performed, where relevant gene expression data is collected for further analysis. Once the dataset is formed, preprocessing steps are carried out to examine the data, including handling missing values, outlier detection, and normalization. In the context of microarray data, feature selection is a crucial next step. Microarray data often has many features, making it necessary to select the most relevant features from the subset. The difference between using feature selection and not using feature selection lies in computational efficiency and higher accuracy [9]. Feature selection reduces the dimensionality of the data, reduces model complexity, and avoids the risk of overfitting [10].

Several previous studies related to microarray data classification have proposed different methods. In one such study, researchers overcame the inefficiency and low accuracy of DNA microarray data by introducing a novel feature gene selection algorithm, RefFPSO, which combines ReliefF and particle swarm optimization (PSO). The algorithm effectively filtered out irrelevant genes through ReliefF and utilized PSO as the search algorithm, achieving a remarkable classification accuracy ranging from 80% to 100% across four datasets [11]. Furthermore, other innovative approaches, such as DPCAForest [12] and a hybrid LASSO [13] and support vector machine (SVM) model, have been introduced, showcasing superior performance in cancer classification on small-sample gene expression datasets compared to conventional methods. Additionally, a simulated Kalman filter [14], with mutation (SKF-MUT) was proposed for feature selection, achieving a classification accuracy ranging from 95% to 100% across diverse cancer datasets. These advancements underscore the continuous efforts to enhance the accuracy and efficiency of microarray data analysis for cancer classification. One of the commonly used techniques to optimize feature selection is the genetic algorithm (GA), which is inspired by biological evolution. GA use natural selection methods to identify and retain relevant features that significantly impact the performance of classification models [15], [16]. Previous studies used GAs to tackle microarray data classification, emphasizing on feature selection. In one approach, an ensemble method combining extreme gradient boosting (XGBoost) and GA effectively classified cancer types in microarray data, achieving higher accuracy compared to other models proposed in this study [17]. Another study [18], combined GA with a dual filter approach named MF-GARF, using random forest (RF) to assess features through relevance, redundancy and optimization stages. Results on cancer microarray data demonstrated the ability of the MF-GARF approach to achieve high accuracy with a minimal number of features, surpassing other hybrid techniques. Another research proposes a novel GA approach to simultaneously optimize feature selection and kernel parameters for SVM, achieving higher classification accuracy with fewer features compared to traditional methods [19].

However, the initialization of the initial population in GA remains a crucial aspect, with traditional random methods potentially leading to suboptimal solutions [20]. In response, the integration of latin hypercube sampling (LHS) with GA presents a promising solution to this challenge [21]. LHS offers a structured approach to population initialization, promoting better variation and convergence in the optimization process [22].

Currently, there are not many implementations of LHS in the context of using GA. Nonetheless, LHS has proven to be very useful in various research contexts. For example, research [21] used LHS to split training and test datasets, comparing it with random sampling. The results showed that LHS had better space-filling properties, which resulted in significant improvements in the accuracy of k-NN, DT, and LR models. On the other hand, [23] applied LHS and simple random sampling for weight and bias initialization of neural networks, and found that LHS resulted in faster and more efficient modeling due to the even distribution of samples. In addition, [22] introduced conditioned LHS (cLHS) as a sampling strategy suitable for area modeling with extensive supporting data, with emphasis on its ability to select samples according to variable distributions, making it versatile for sampling continuous and categorical variables with optimization potential.

Building upon these insights, this research aims to enhance the classification performance of microarray data by proposing a model that combines GAs with LHS for feature selection. By leveraging the benefits of LHS in initializing the GA population, we seek to explore a more diverse and representative feature space, ultimately improving the accuracy and efficiency of microarray data classification models. LHS helps GA to explore a large search space in terms of feature combinations more efficiently. LHS can form a more diverse and representative population compared to random initialization methods, thus increasing the chances of the GA finding the optimal solution. This overcomes the complexity of high-dimensional microarray data and limited number of samples. Through GA iterations, the optimal combination of features will be generated to achieve the highest classification performance. The integration of GA and LHS in feature selection is expected to produce a more accurate classification model compared to other methods.

## 2. METHOD

### 2.1. Proposed model

This section delves into a feature selection method that utilizes a GA with an initial population initialized by LHS, specifically designed for high-dimensional microarray data. The different stages involved are outlined in Figure 1.

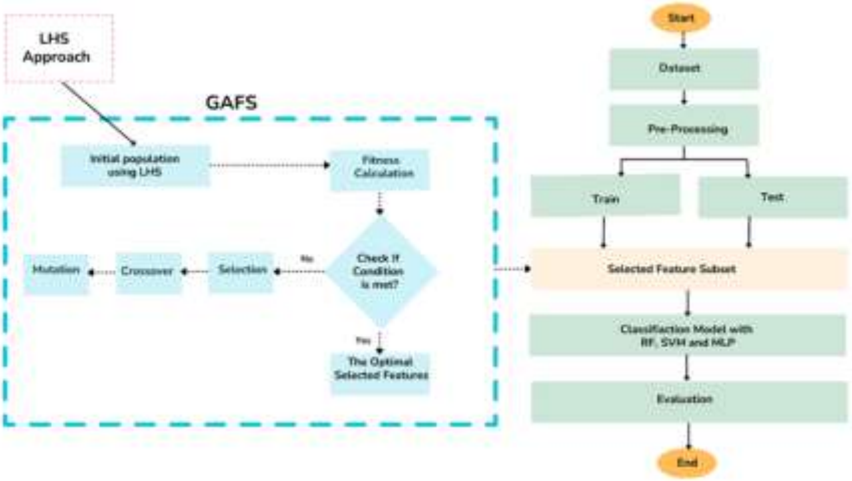

Figure 1. System architecture of the proposed GA-based feature selection using LHS

### 2.2. Dataset

In this paper, we use three high-dimensional cancer microarray datasets to assess the performance of GA-based feature selection method with LHS approach on initialization of initial population. These three datasets are central nervous system (CNS) [24], Sarcoma [25], and small round blue cell tumors (SRBTCs) [26]. The description of the microarray dataset is presented in Table 1.

Table 1. Description of microarray dataset

| Dataset | No. of features | No. of instances | No. of classes |
|---|---|---|---|
| CNS | 7,128 | 60 | 2 |
| Sarcoma | 22,283 | 105 | 10 |
| SRBTCs | 2,308 | 63 | 4 |

The first dataset is the CNS extracted from biopsies of brain cancer patients called medulloblastoma. 39 samples contain the survival class of medulloblastoma, and 21 other samples are treatment failures, which means individuals who experienced treatment failure. The number of features contained in the Colon data is 7,128 features.

The second dataset is microarray data from Sarcoma data [22]. Sarcoma data is from patients with malignant tumors (cancer) that start in soft tissue. There are 105 samples divided into 10 classes. The number of each class is as follows: synovial sarcoma (16), myxoid/round cell liposarcoma (19), lipoma (3), well-differentiated liposarcoma (3), dedifferentiated liposarcoma (15), myxofibrosarcoma (15), leiomyosarcoma (6), MPNST (3), fibrosarcoma (4), and MFH (21). The number of features contained in the Sarcoma data is 22,283.

The third dataset is microarray data from SRBTCs data [23]. These tumors have a similar appearance to light microscopy and are often indistinguishable by common immunocytochemical markers. There were 63 samples divided into 4 classes. The number of each class includes neuroblastoma (12), rhabdomyosarcoma (20), non-Hodgkin lymphoma (8), and Ewing family of tumors (23). The number of features contained in the Sarcoma data is 2,308.

### 2.3. Preprocessing

The pre-processing stage of microarray datasets involves understanding their characteristics, which consist of float-formatted data with a wide range of values. Checking for missing values and outliers is essential to obtain accurate performance results, and addressing them needs to be done before proceeding. The dataset was divided into training set (75%) and testing set (25%) for classification modeling. The training data undergoes min-max normalization, where the scale of the data is transformed from one range to another [27].

### 2.4. LHS

LHS is a sampling method widely employed in statistical analysis and numerical experiments to generate more homogeneous and evenly distributed samples from the sample space compared to regular random sampling methods. Initially developed for Monte Carlo simulations, particularly to efficiently select input variables for computer mode [28], LHS has found applications in diverse fields such as soil science, environmental studies for uncertainty assessment in predictive models, and geostatistics for simulating gaussian random fields. LHS adopts the concept of a Latin square, ensuring that each row and column contains only one sample. This concept has been extended to be applicable in various dimensions.

In the implementation of LHS for multivariate distributions, a sample size (n) from multiple variables is chosen such that the samples are evenly distributed for each variable. This is achieved by dividing the sample space into n strata and selecting one sample from each stratum. In the initial stage of LHS algorithm [22], the distribution of each variable is systematically divided into n intervals, each with an equal probability of selection. This division guarantees that every interval or stratum holds an identical likelihood of being chosen during the subsequent sampling process. For a given interval, denoted as the ith interval, the cumulative probability (Probi) is determined using:

$$\mathrm{Prob}_i = \frac{1}{n} \times r_u + \frac{i-1}{n} \quad (1)$$

Here, $r_u$ represents a uniformly distributed random number ranging from 0 to 1. This calculation ensures that the cumulative probability for each interval is accurately derived. Following this, the sampled x value for each interval is obtained by applying the inverse of the distribution function:

$$x = F^{-1}(\mathrm{Prob_i}) \quad (2)$$

This transformation allows the generation of sampled values that adhere to the specified distribution, ensuring the representation of each interval in the variable’s range. In the final step of the LHS algorithm, the values obtained for each variable are systematically paired, either in a random manner or following a prescribed order, with corresponding values of the other variables.

### 2.5. Genetic algorithm

In 1975, a researcher named John Holland developed the concept of GA, which is inspired by Charles Darwin’s theory of evolution. GA is used as a technique to identify optimal solutions to optimization problems by using the principle of natural selection. Optimal solutions are obtained through a process of selection, mutation, and crossover that is repeated repeatedly in a population. A chromosome is a representation of a solution in GA, while a population consists of a collection of chromosomes that are used to form a new population. GA strives to produce the best and optimal population for the problem at hand. Some biological terms in the GA are initial population, chromosome, gene, fitness, selection, crossover, and mutation [15].

### 2.6. GA based feature selection using LHS

In Figure 1, the block diagram showing the GAFS part using LHS consists of the following steps: generation of initial population using LHS, calculation of fitness function and selection strategy for selection of parents that produce offspring for the next generation, crossover, mutation, and production of the next generation.

#### 2.6.1. Initial population using LHS

The first step in executing the GA is to create the initial population. The population consists of a number of chromosomes, representing all possible combinations of features. Each chromosome signifies a set of selected features and is encoded with a series of 0s and 1s (where 1 indicates a chosen feature and 0 indicates a feature that is not selected). In this phase, chromosomes are randomly created through simple random sampling. In this study, the researcher attempts a different approach by replacing the random method with LHS. The code for this approach is as follows:

```
begin
def latin_hypercube_sampling(bounds, population_size):
    num_dimensions = len(bounds)
    interval_width = 1.0 / population_size
    lhs_points = np.empty((population_size, num_dimensions))
    for dim in range(num_dimensions):
        points = np.arange(0, 1, interval_width) + np.random.uniform(0,
        interval_width, population_size)
        np.random.shuffle(points)
        lhs_points[:, dim] = points
    scaled_lhs_points = np.empty_like(lhs_points)
    for dim in range(num_dimensions):
        min_val, max_val = bounds[dim]
        scaled_lhs_points[:, dim] = min_val + lhs_points[:, dim] * (max_val
        - min_val)
    return scaled_lhs_points
end
```

The LHS method works by first dividing the parameter space into d intervals, where each interval corresponds to one dimension. Then, a list of n points is created within each interval, where each point is randomly assigned an offset. The list of points is then shuffled to ensure that the points are randomly distributed across the parameter space. Finally, the scaled LHS points are returned, where they are scaled so that they fall within the specified range for each dimension. Figure 2, illustrates a comparison between the initial population samples generated using the LHS and SRS methods, with a sample population of 6 and a feature count of 6. The blue dots represent true-valued features, while the red dots represent false-valued features.

The initial selection using the simple random method is done randomly in the range of 0 to 1 for each feature, resulting in a diverse and random initial population with no particular structure in the feature distribution. In some cases, this can lead to some individuals in the population having the same or very similar combinations of features, as seen in chromosomes 4, and 5, where 3 true-valued features and 3 false-valued features are in the same position.

On the other hand, when using LHS, the initial population is distributed more evenly across the feature space. Point selection minimizes the probability of selecting the same feature between true and false features in all chromosomes. The number of feature selections is also evenly distributed between 0 and 6 correct and incorrect feature selections. Finally, in Figure 3, a population is created with a chromosome combination between feature 0 which was not selected and feature 1 which was selected by applying LHS.

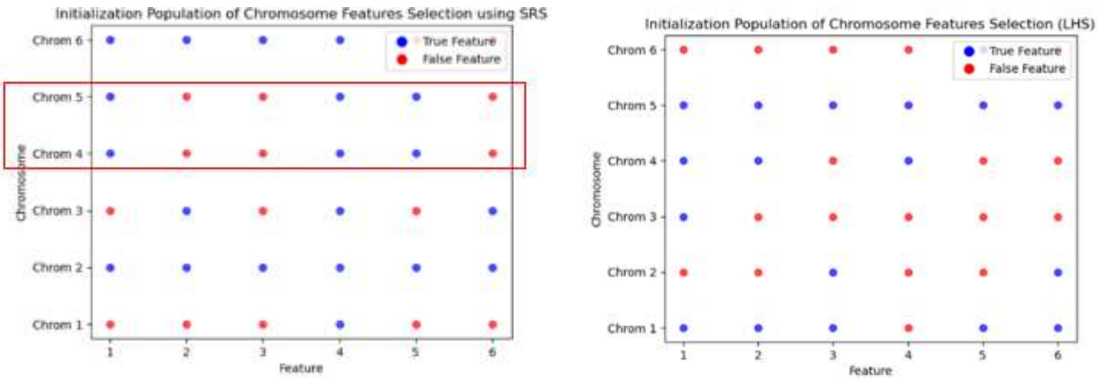

Figure 2. Comparison of initial population sampling using LHS with SRS

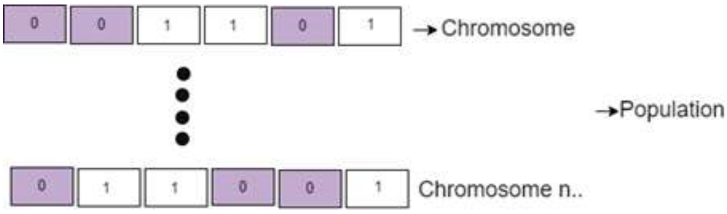

Figure 3. Initial population

### 2.6.2. Fitness and selection

The fitness function is used to assess the performance of each chromosome, by training a model to classify a subset of features corresponding to the genes in the chromosome. The measured model performance will be used to predict the accuracy of test data. Individuals with higher fitness function values will have a greater probability of becoming the next generation of their parent individuals.

Selection is the selection of the best-performing chromosomes that will be chosen as parents to produce the next generation. The number of parent chromosomes selected is determined by the n_parents parameter. If n_parents is set to two, then two chromosomes are selected as parents to proceed to the next stage. Chromosome selection can be done in various ways, one of which is by using feature ranking. Feature ranking is the process of ranking features based on their influence on model performance.

### 2.6.3. Crossover

Crossover is an important stage in GAs that is performed after selection. In this research, the crossover method used is the one-point crossover. Figure 4 illustrates the one-point crossover method, in which a cut point is selected on the parent chromosomes as a separation boundary to exchange the two chromosome parts of the parents.

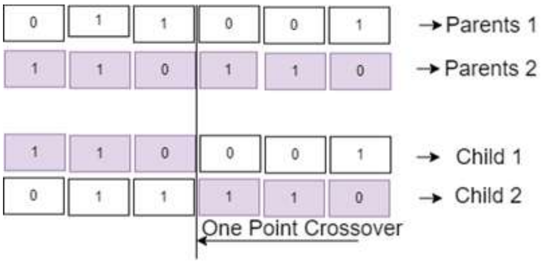

Figure 4. One point crossover

### 2.6.4. Mutation

Figure 5 shows the mutation step in GA that introduces random changes to offspring chromosomes, aiming to promote genetic diversity within the population and prevent convergence to suboptimal solutions. Its role is to randomly alter a few genes in offspring chromosomes, generating new variations and facilitating an exploration of a broader search space, thus avoiding getting trapped in undesired local minima. The mutation rate, determined by an input parameter, dictates how often mutations occur, with, for instance, a 10% mutation rate meaning that around 10% of genes in each offspring chromosome will be randomly altered.

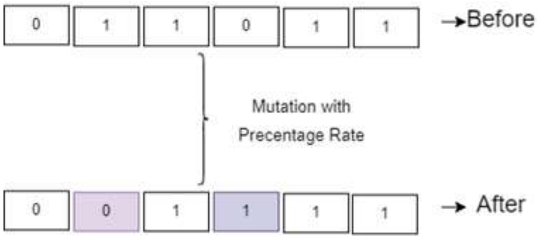

Figure 5. Mutation

### 2.6.5. Production of the next generation

The final stage is a loop or literacy of literacy from the fitness, selection, crossover, and mutation process steps. A new population for each generation will be formed. All these steps are repeated in n_generation iterations. The result is a continuous evolution of the chromosome population, where each generation strives to improve the combination of features and the performance of the classification model. The goal of generation is to improve the combination of features the classification model uses. A new chromosome population is always formed in the hope of producing a better combination of features than the previous generation.

## 3. RESULTS AND DISCUSSION

The effect of using LHS in GA population initialization can affect the results of feature selection in microarray classification. Before observing the results, it is necessary to explain the experimental setup of this study. The classification used the K-nearest neighbors (KNN) model, and feature selection was performed using GA with LHS compared to GA using the random method for chromosome formation in the initial population. The experiment lasted for 10 generations with an initial population of 500 chromosomes, using a crossover of 250 children and a mutation rate of 0.01.

In Table 2, the results obtained on the CNS dataset are described, revealing that LHS-based feature selection resulted in feature selection that ranged from 3527 to 3626 with corresponding accuracy between 66% and 73%. In contrast, without LHS, the feature set ranged from 682 to 1647 with accuracies between 73% and 80%. Notably, the average accuracy across generations in the CNS dataset shows the superiority of LHS-based selection compared to the non-LHS approach.

In the Sarcoma dataset, where LHS-based feature selection resulted in feature selection ranging from 11097 to 1220 with accuracies between 74% and 77%. Without LHS, the feature set ranged from 4604 to 9026 with accuracy between 74% and 80%. Remarkably, LHS achieved an accuracy of 81% in the 10th generation, outperforming the non-LHS approach which failed to achieve the same accuracy in the same timeframe.

The last dataset, SRBTCs, LHS-based feature selection resulted in feature selections that ranged from 1,117 to 1,186 with accuracies between 87.5% and 100%. In contrast, without LHS, the feature sets ranged from 259 to 1,171 with accuracies between 87.5% and 100%. It is important to note that LHS accelerates the achievement of perfect accuracy 100% from the 3rd generation, whereas the non-LHS approach achieves the same accuracy only at the 8th generation.

Table 2. Comparison of selected feature distribution between GA without LHS and GA with LHS

| Dataset / Num of generations | Number of selected features without LHS | Accuracy without LHS | Number of selected features with LHS | Accuracy with LHS |
|---|---|---|---|---|
| CNS | | | | |
| 1 | 3550 | 0.6 | 682 | 0.73 |
| 2 | 3527 | 0.66 | 739 | 0.73 |
| 3 | 3536 | 0.66 | 800 | **0.80** |
| 4 | 3584 | 0.66 | 847 | 0.73 |
| 5 | 3585 | 0.66 | 907 | **0.80** |
| 6 | 3629 | 0.66 | 960 | **0.80** |
| 7 | 3626 | 0.66 | 1001 | **0.80** |
| 8 | 3570 | 0.73 | 1441 | 0.73 |
| 9 | 3571 | 0.66 | 1598 | **0.80** |
| 10 | 3588 | 0.66 | 1647 | **0.80** |
| Sarcoma | | | | |
| 1 | 11178 | 0.74 | 9026 | 0.74 |
| 2 | 11119 | 0.74 | 4604 | 0.77 |
| 3 | 11097 | 0.74 | 4744 | 0.77 |
| 4 | 11175 | 0.77 | 4860 | 0.77 |
| 5 | 11220 | 0.77 | 6048 | 0.77 |
| 6 | 11214 | 0.77 | 6132 | 0.77 |
| 7 | 11206 | 0.77 | 6220 | 0.77 |
| 8 | 11183 | 0.77 | 5076 | 0.77 |
| 9 | 11183 | 0.77 | 8418 | 0.77 |
| 10 | 11183 | 0.77 | 6118 | **0.81** |
| SRBTCs | | | | |
| 1 | 1164 | 0.875 | 1171 | 0.875 |
| 2 | 1117 | 0.875 | 497 | 0.9375 |
| 3 | 1163 | 0.875 | 259 | **1** |
| 4 | 1181 | 0.9375 | 278 | 0.9375 |
| 5 | 1182 | 0.875 | 515 | **1** |
| 6 | 1185 | 0.9375 | 526 | **1** |
| 7 | 1186 | 0.9375 | 337 | **1** |
| 8 | 1138 | **1** | 404 | **1** |
| 9 | 1139 | **1** | 436 | **1** |
| 10 | 1164 | **1** | 455 | **1** |

To evaluate the model accuracy performance and the impact of LHS, it is necessary to compare it with other feature selection methods. In previous research [20], experiments were conducted by applying the Relief and Lasso feature selection methods on the same dataset. The three classification models used were SVM, multi-layer perceptron (MLP), and RF. We also experimented with feature selection based on correlation feature (BCF) [29] and principal component analysis (PCA) [30] to increase the variety of feature selection models. Furthermore, we propose a new approach by GA and LHS to see the accuracy of the model proposed in this study.

Table 3, In the first model using SVM classification, the results show performance variation among feature selection methods. The CNS dataset achieved 100% accuracy when using the Relief and GALHS methods, while the Lasso method was 85%, GA 80%, BCF 73%, and PCA 66%. For the Sarcoma dataset, GALHS gave the highest accuracy of 88%, while the Lasso, GA, BCF, and PCA methods gave lower results. Meanwhile, on the SRBTCs dataset, all feature selection methods achieved 100% accuracy except BCF and PCA which scored 81%.

Table 4, in the second model using MLP classification, it can be observed that the Relief and GALHS methods consistently provide performance across all datasets. In the CNS dataset, the highest accuracy is achieved by the Relief model with 100%, while GALHS obtains an accuracy of 86%. For the Sarcoma dataset, the GALHS method emerges as the highest accuracy compared to other models, reaching an accuracy level of 85%. The SRBTCs dataset shows the same results as the SVM model, with an accuracy rate of 100% for all models, except for the BCF and PCA models. Table 5, in the third model with RF classification, the results indicate that the GA and GALHS methods perform better on the CNS and SRBTC datasets compared to the Relief, Lasso, BCF, and PCA methods. However, for the Sarcoma dataset, the GALHS method still outperforms other methods, achieving an accuracy of 69%.

Overall, the experiments consistently demonstrate that the combination of GA with LHS (GALHS) proves superior across all trials, outperforming other feature selection methods. This superiority is particularly evident in MLP and RF classifications, showcasing GALHS as a robust and effective strategy for feature selection and hyperparameter tuning across diverse datasets. Notably, on the Sarcoma dataset, GALHS consistently achieves high levels of accuracy, further highlighting its overall excellence in comparison to the previously employed methods.

Table 3. Accuracy model feature selection comparison using SVM classification

| Dataset | BCF | PCA | Relief | Lasso | GA | GALHS |
|---|---|---|---|---|---|---|
| CNS | 0.73 | 0.66 | 1 | 0.85 | 0.80 | 1 |
| Sarcoma | 0.59 | 0.62 | 0.69 | 0.64 | 0.78 | 0.88 |
| SRBTCs | 0.81 | 0.81 | 1 | 1 | 1 | 1 |

Table 4. Accuracy model feature selection comparison using MLP classification

| Dataset | BCF | PCA | Relief | Lasso | GA | GALHS |
|---|---|---|---|---|---|---|
| CNS | 0.66 | 0.73 | 1 | 0.85 | 0.80 | 0.86 |
| Sarcoma | 0.66 | 0.59 | 0.68 | 0.74 | 0.81 | 0.85 |
| SRBTCs | 0.81 | 0.81 | 1 | 1 | 1 | 1 |

Table 5. Accuracy model feature selection comparison using RF classification

| Dataset | BCF | PCA | Relief | Lasso | GA | GALHS |
|---|---|---|---|---|---|---|
| CNS | 0.80 | 0.73 | 0.66 | 0.75 | 0.93 | 0.93 |
| Sarcoma | 0.74 | 0.74 | 0.58 | 0.53 | 0.68 | **0.69** |
| SRBTCs | **1** | 0.93 | **1** | 0.98 | **1** | 1 |

## 4. CONCLUSION

This research addresses the important problem of optimizing feature selection for microarray data classification. This research incorporates LHS within a GA framework. Our analysis, using three different data sets, revealed an important finding. GA alone showed limited variation in the selected features, indicating a potential limitation in exploring the feature space. In contrast, integrating GALHS results in a much wider distribution of selected features. The wider variety of features in GALHS promises to improve model performance. Furthermore, this study comprehensively evaluates various feature selection methods and classification models. In particular, GALHS consistently outperformed all other approaches, especially in SVM, MLP, and RF classification. These findings demonstrate the good accuracy and superior stability of GALHS across various datasets. Our study underscores the contribution of LHS in optimizing GA-based feature selection for microarray data classification. Moreover, this research can be extended in the future with the application of GALHS to feature fusion and feature extraction.

## ACKNOWLEDGEMENTS

This research is supported by a program of master’s thesis research, Institut Teknologi Sepuluh Nopember, 2024.

## REFERENCES

[1] W. Ali and F. Saeed, “Hybrid filter and genetic algorithm-based feature selection for improving cancer classification in high-dimensional microarray data,” *Processes*, vol. 11, no. 2, Feb. 2023, doi: 10.3390/pr11020562.

[2] V. Bolón-Canedo, N. Sánchez-Maroño, A. Alonso-Betanzos, J. M. Benítez, and F. Herrera, “A review of microarray datasets and applied feature selection methods,” *Information Sciences*, vol. 282, pp. 111–135, Oct. 2014, doi: 10.1016/j.ins.2014.05.042.

[3] R. Govindarajan, J. Duraiyan, K. Kaliyappan, and M. Palanisamy, “Microarray and its applications,” *Journal of Pharmacy and Bioallied Sciences*, vol. 4, no. 6, p. 310, 2012, doi: 10.4103/0975-7406.100283.

[4] K. Cahyaningrum, Adiwijaya, and W. Astuti, “Microarray gene expression classification for cancer detection using artificial neural networks and genetic algorithm hybrid intelligence,” *2020 International Conference on Data Science and Its Applications (ICoDSA)*, 2020, doi: 10.1109/ICoDSA50139.2020.9213051.

[5] E. Bair, “Identification of significant features in DNA microarray data,” *Wiley Interdiscip Rev Comput Stat*, vol. 5, no. 4, pp. 309–325, Jul. 2013, doi: 10.1002/wics.1260.

[6] N. Almugren and H. Alshamlan, “FF-SVM: new firefly-based gene selection algorithm for microarray cancer classification,” *IEEE Computational Intelligence Society Institute of Electrical and Electronics Engineers*, 2019, doi: 10.1109/CIBCB.2019.8791236.

[7] S. Turgut, M. Dagtekin, and T. Ensari, “Microarray breast cancer data classification using machine learning methods,” *IEEE Engineering in Medicine and Biology Society*, 2018, doi: 10.1109/EBBT.2018.8391468.

[8] C. Yan, J. Zhang, X. Kang, Z. Gong, J. Wang and G. Zhang, “Comparison and evaluation of the combinations of feature selection and classifier on microarray data,” *2021 IEEE 6th International Conference on Big Data Analytics (ICBDA)*, Xiamen, China, 2021, pp. 133-137, doi: 10.1109/ICBDA51983.2021.9403151.

[9] R. K. Singh and M. Sivabalakrishnan, “Feature selection of gene expression data for cancer classification: a review,” in *Procedia Computer Science*, Elsevier B.V., 2015, pp. 52–57. doi: 10.1016/j.procs.2015.04.060.

[10] G. Chandrashekar and F. Sahin, “A survey on feature selection methods,” *Computers and Electrical Engineering*, vol. 40, no. 1, pp. 16–28, Jan. 2014, doi: 10.1016/j.compeleceng.2013.11.024.

[11] M. Liu, L. Xu, J. Yi, and J. Huang, “A feature gene selection method based on ReliefF and PSO,” *2018 10th International Conference on Measuring Technology and Mechatronics Automation (ICMTMA)*, Changsha, China, 2018, pp. 298-301, doi: 10.1109/ICMTMA.2018.00079.

[12] X. Deng and Y. Xu, “Cancer classification using microarray data by DPCAForest,” in *Proceedings - International Conference on Tools with Artificial Intelligence, ICTAI*, IEEE Computer Society, Nov. 2019, pp. 1081–1087. doi: 10.1109/ICTAI.2019.00151.

[13] K. Güçkıran, İ. Cantürk, and L. Özyılmaz, “DNA microarray gene expression data classification using SVM, MLP, and RF with feature selection methods relief and lasso,” *Süleyman Demirel Üniversitesi Fen Bilimleri Enstitüsü Dergisi*, pp. 115–121, Apr. 2019, doi: 10.19113/sdufenbed.453462.

[14] N. A. Zamri, N. A. Nor, T. Bhuvaneswari, N. H. A.Aziz, and A. K. Ghazali, “Feature selection of microarray data using simulated kalman filter with mutation,” *Processes*, vol. 11, no. 8, Aug. 2023, doi: 10.3390/pr11082409.

[15] S. Prajapati, H. Das, and M. K. Gourisaria, “Feature selection using genetic algorithm for microarray data classification,” 2022 *OPJU International Technology Conference on Emerging Technologies for Sustainable Development (OTCON)*, Raigarh, Chhattisgarh, India, 2023, pp. 1-6, doi: 10.1109/OTCON56053.2023.10113937.

[16] Z. Wang, Y. Zhou, T. Takagi, J. Song, Y. S. Tian, and T. Shibuya, “Genetic algorithm-based feature selection with manifold learning for cancer classification using microarray data,” *BMC Bioinformatics*, vol. 24, no. 1, Dec. 2023, doi: 10.1186/s12859-023-05267-3.

[17] M. T. Ashraf, I. Hamid, Q. Nawaz, and H. Ali, “Hybrid approach using extreme gradient boosting (xgboost) and evolutionary algorithm for cancer classification,” in *2023 International Multi-Disciplinary Conference in Emerging Research Trends*, Institute of Electrical and Electronics Engineers Inc., 2023. doi: 10.1109/IMCERT57083.2023.10075236.

[18] P. Saqib, U. Qamar, R. A. Khan, and A. Aslam, “MF-GARF: hybridizing multiple filters and GA wrapper for feature selection of microarray cancer datasets,” *2020 22nd International Conference on Advanced Communication Technology (ICACT)*, Phoenix Park, Korea (South), 2020, pp. 517-524, doi: 10.23919/ICACT48636.2020.9061234.

[19] C. L. Huang and C. J. Wang, “A GA-based feature selection and parameters optimizationfor support vector machines,” *Expert Syst Appl*, vol. 31, no. 2, pp. 231–240, Aug. 2006, doi: 10.1016/j.eswa.2005.09.024.

[20] H. M. Pandey, A. Chaudhary, and D. Mehrotra, “A comparative review of approaches to prevent premature convergence in GA,” *Applied Soft Computing Journal*, vol. 24. Elsevier Ltd, pp. 1047–1077, 2014. doi: 10.1016/j.asoc.2014.08.025.

[21] I. Iordanis, C. Koukouvinos, and I. Silou, “Classification accuracy improvement using conditioned latin hypercube sampling in supervised machine learning,” in *Proceedings of the 2022 IEEE 12th International Conference on Dependable Systems, Services and Technologies, DESSERT 2022*, Institute of Electrical and Electronics Engineers Inc., 2022. doi: 10.1109/DESSERT58054.2022.10018677.

[22] B. Minasny and A. B. McBratney, “A conditioned Latin hypercube method for sampling in the presence of ancillary information,” *Computers & Geosciences*, vol. 32, no. 9, pp. 1378–1388, Nov. 2006, doi: 10.1016/j.cageo.2005.12.009.

[23] J. H. Leea, Y.-D. Ko, I. Yun, and K. Han, “Comparison of latin hypercube sampling and simple random sampling applied to neural network modeling of HfOz thin film fabrication,” *Transactionson Electrical and Electronic Material*, vol. 7, no. 4, 2006, doi: https://doi.org/10.4313/TEEM.2006.7.4.210.

[24] L. P. Scott *et al.*, “Prediction of central nervous system embryonal tumour outcome based on gene expression,” *Macmillan Magazines*, 2002.

[25] R. Nakayama *et al.*, “Gene expression analysis of soft tissue sarcomas: characterization and reclassification of malignant fibrous histiocytoma,” *Modern Pathology*, vol. 20, no. 7, pp. 749–759, Jul. 2007, doi: 10.1038/modpathol.3800794.

[26] K. Javed *et al.*, “Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks,” *Nature Publishing Group*, 2001.

[27] H. Henderi, T. Wahyuningsih, and E. Rahwanto, “Comparison of min-max normalization and z-score normalization in the k-nearest neighbor (KNN) algorithm to test the accuracy of types of breast cancer,” *International Journal of Informatics and Information System*, vol. 4, pp. 13–20, 2021.

[28] M. D. McKay, R. J. Beckman, and W. J. Conover, “Comparison of three methods for selecting values of input variables in the analysis of output from a computer code,” *Technometrics*, vol. 21, no. 2, pp. 239–245, 1979, doi: 10.1080/00401706.1979.10489755.

[29] M. Al-Batah, B. Zaqaibeh, S. A. Alomari, and M. S. Alzboon, “Gene microarray cancer classification using correlation based feature selection algorithm and rules classifiers,” *International journal of online and biomedical engineering*, vol. 15, no. 8, pp. 62–73, 2019, doi: 10.3991/ijoe.v15i08.10617.

[30] J. Liu, W. Cai, and X. Shao, “Cancer classification based on microarray gene expression data using a principal component accumulation method,” *Science China Chemistry*, vol. 54, no. 5, pp. 802–811, May 2011, doi: 10.1007/s11426-011-4263-5.

## BIOGRAPHIES OF AUTHORS

**Bangun Rizki Awangditama** received his Bachelor’s degree from the State University of Jember in 2014. He is currently pursuing his Master’s degree at Institut Teknologi Sepuluh November (ITS). His research interests are in Programming, Machine learning, and Deep Learning. Currently, he also works at a Telkom Surabaya University institution as IT and Application Staff. He can be contacted via email: 6025221044@student.its.ac.id.

**Nanik Suciati** held a doctorate in Information Engineering from Hiroshima University, Japan in 2010. She received her Bachelor’s degree from Institut Teknologi Sepuluh Nopember (ITS) in 1994 and her Master’s degree from the University of Indonesia (UI) in 1998. She is been a Professor in the Department of Informatics, ITS, Indonesia since 2023. She also served as Head of the Intelligent Computing and Vision Laboratory since 2015-present. Her research interests are computer vision, computer graphics, computational intelligence, and machine learning. She has published over 120 papers in national journals, international journals, and conferences. She can be contacted at email: nanik@if.its.ac.id.