JOIN (Jurnal Online Informatika)
Volume 5 No.1 | June 2020: 9-18
DOI: 10.15575/join.v5i1.569

p-ISSN: 2528-1682
e-ISSN: 2527-9165

Two-stage Gene Selection and Classification for a High-Dimensional Microarray Data

Masithoh Yessi Rochayani[1], Umu Sa'adah[2], Ani Budi Astuti[3]
[1,2,3]Department of Statistics, Universitas Brawijaya, Indonesia

Article Info

Article history:

Received May 12, 2020
Revised June 20, 2020
Accepted June 23, 2020
Published July 15, 2020

Keywords:

Classification and Regression
Feature selection
Gene expression
High-dimensional
Microarray
Tree

ABSTRACT

Microarray technology has provided benefits for cancer diagnosis and classification. However, classifying cancer using microarray data is confronted with difficulty since the dataset has high dimensions. One strategy for dealing with the dimensionality problem is to make a feature selection before modeling. Lasso is a common regularization method to reduce the number of features or predictors. However, Lasso remains too many features at the optimum regularization parameter. Therefore, feature selection can be continued to the second stage. We proposed Classification and Regression Tree (CART) for feature selection on the second stage which can also produce a classification model. We used a dataset which comparing gene expression in breast tumor tissues and other tumor tissues. This dataset has 10,936 predictor variables and 1,545 observations. The results of this study were the proposed method able to produce a few numbers of selected genes but gave high accuracy. The model also acquired in line with the Oncogenomics Theory by the obtained of GATA3 to split the root node of the decision tree model. GATA3 has become an important marker for breast tumors.

Corresponding Author:

Umu Sa'adah,
Department of Statistics,
Universitas Brawijaya,
Jln. Veteran, Malang, Indonesia
Email: u.saadah@ub.ac.id

1. INTRODUCTION

The emergence of microarrays technology has provided benefits for cancer research. A DNA microarray is a technology to help in studying gene expression. Scientists have used microarray data to identify healthy people and patients from various types of cancer. However, analysis of microarray data is a challenging task since it contains thousands even tens of thousands of feature genes with a small number of observations, which is commonly known as high-dimensional data. It is not efficient to involve all genes for classification because of the presence of redundant genes and the model will be complicated.

To deal with this issue, one strategy for classifying high-dimensional data is to select the variables in the preprocessing stage. The existing methods of variable selection can be broken up into filter, wrapper, and embedded [1]. The filter methods reduce the number of variables by rank each feature based on a univariate metric such as information gain or chi-square. These methods have computationally cheaper as compared to the wrapper and embedded methods. But filter methods not include interaction with the classification model to select the best variable subset. Wrapper methods work by evaluating all possible variable subsets using an algorithm. These methods can produce more beneficial variables than filter methods. Yet, wrapper methods are computationally more expensive as compared to filter methods, especially when performed on very high-dimensional data. Embedded methods interact with the classification model to produce the selected variable and have computationally efficient as compared to wrapper [2-3]. Therefore, embedded methods, such as regularization, are preferred over the filter and wrapper methods.

The most common regularization method is the Least Absolute Shrinkage and Selection Operator (Lasso) proposed by Tibshirani [4]. This method works by adding a penalty function, known as the L1 penalty, to the objective function of regression. The number of nonzero coefficients of Lasso depends on the regularization parameter chosen. Various kinds of literature had proposed several approaches to determine the optimum regularization parameter, including cross-validation, information criteria [5-6], and algorithm [7], but the most commonly used is cross-validation. Lasso and other regularization methods tend to produce a large number of predictors at the optimum regularization parameter as in [8] and [9]. Therefore, feature selection can be continued to the second stage.

Classification and Regression Trees (CART), introduced by Breiman et al. [10], is a supervised learning algorithm that makes predictive models based on tree structures. This algorithm can be used to model any kind of data since it does not need any assumptions for the data. This method also has advantages, that is, can be used for variable selection, and the model acquired can be interpreted easily.

For gene selection and classification from high-dimensional microarray data, we proposed the two-stage gene selection, namely Lasso and CART hybrid method (Lasso+CART). At the first stage of gene selection, we used Lasso, and at the second stage, we used CART which also able to construct an easily interpretable classification model. The model compatibility was measured using AUC. In addition to evaluating the goodness of the model, we also interpret and validate it based on theory. The success of this method will be very useful in medical research, especially to discover new knowledge from a disease.

2. METHOD

2.1. Regularized logistic regression

The dataset was first split into training and testing sets. Then, the training data was normalized using z-score normalization and continued with the implementation of Lasso for gene selection. Because the response variable was binary, then the model used was binary logistic regression. Binary logistic regression is a method that can be used to determine the relationship between binary response variables and continuous or categorical predictor variables. Let $\pi(\boldsymbol{x}_i)$ represent the probability of class y=1 that represents the probability of success, a binary logistic regression model with p predictor variables are stated by (1) as follows.

$$\pi(\boldsymbol{x}_i) = \frac{\exp(\boldsymbol{x}_i\boldsymbol{\beta})}{1 + \exp(\boldsymbol{x}_i\boldsymbol{\beta})} \tag{1}$$

where: $\boldsymbol{x}_i$ is a vector of predictors at i$^{\text{th}}$ observation and $\boldsymbol{\beta} = (\beta_1, \beta_2, \dots, \beta_n)$ is a vector containing the logistic regression coefficients. The vector $\boldsymbol{\beta}$ is obtained by maximizing the likelihood function stated by (2).

$$L(\boldsymbol{\beta}) = \prod_{i=1}^{n} \pi(x_i)^{y_i}\big(1 - \pi(x_i)\big)^{1-y_i} \tag{2}$$

Maximizing the likelihood function is similar to minimizing the negative log-likelihood function (3).

$$J(\boldsymbol{\beta}) = -\left[\sum_{i=1}^{n} y_i\,(\boldsymbol{x}_i\boldsymbol{\beta}) - \ln\big(1 + e^{\boldsymbol{x}_i\boldsymbol{\beta}}\big)\right] \tag{3}$$

Therefore,

$$\widehat{\boldsymbol{\beta}} = \arg\min J(\boldsymbol{\beta}) \tag{4}$$

The negative log-likelihood function is continuous, convex, and derivable. Therefore, numerical optimization methods, such as coordinate descent method can be used to obtain $\widehat{\boldsymbol{\beta}}$ [11].

Regularization works by adding constrain (penalty) when minimizing the objective function. The penalty function of Lasso is stated by (5).

$$p(\beta_j, \lambda) = \lambda \sum\nolimits_{j=1}^{p} |\beta_j|, \tag{5}$$

where: $\lambda > 0$ is a regularization parameter that controls the number of nonzero coefficients. Coefficients of Lasso $\left(\widehat{\boldsymbol{\beta}}_{Lasso}\right)$ are obtained by solving the optimization problem stated by (6).

$$\widehat{\boldsymbol{\beta}} = \arg\min\left\{J(\beta_j) + \lambda \sum\nolimits_{j=1}^{p} |\beta_j|\right\} \tag{6}$$

The algorithm used to solve equation (6) was pathwise coordinate optimization [12-15], which contains three nested loops. In the outer loop, the regularization parameter is derived iteratively. In the middle loop, the quadratic approximation of negative log-likelihood is updated using the current parameters. In the inner loop, the coordinate descent algorithm is run to solve (6).

2.2. Choosing optimum regularization parameter

Since the number of nonzero coefficients depends on the regularization parameter chosen, then the optimum regularization parameter has to be determined. According to [16], the method commonly used to select optimum λ is the *K*-fold CV, and using 5 or 10-fold is recommended. *K*-fold CV splits data into subsets with relatively the same size. *K*-1 subsets are used as training data and 1 subset is used as testing data. This process will be carried out *K* times so that each subset of data could become testing data.

The optimum λ is determined based on the average binomial deviance. The model will get better if the value of the binomial deviance gets smaller [17]. The formula to get the binomial deviance is stated in the equation (7).

$$Dev = 2\sum_{i} o_i \log\left(\frac{o_i}{e_i}\right) \tag{7}$$

where o_i denoted observed, e_i denoted expected under the model of interest. The optimum λ chosen is the λ that gives the smallest average binomial deviation results from cross-validation. The *K*-fold CV used to obtain the optimum Lasso regularization parameters is described as follows [18]:

1. Divide data $\{1, 2, \dots, n\}$ into *K* subset (fold) $(\mathrm{F}_1, F_2, \dots F_K)$ with relatively the same size
2. For $k = 1, 2, \dots, K$:
   a. use $(\mathrm{x_i}, y_i), i \notin F_k$ as training data and $(\mathrm{x_i}, y_i), i \in F_k$ as testing
   b. for each regularization parameter $\lambda \in \{\lambda_1, \dots, \lambda_\mathrm{m}\}$, which m denotes the iteration index, calculate the total binomial deviance $Dev_k(\lambda)$ of the testing data.
3. For each regularization parameter λ, calculate the average binomial deviance over all folds using formula (8)

$$\mathrm{CV}(\lambda) = \frac{1}{n}\sum_{k=1}^{K} Dev_k(\lambda) \tag{8}$$

4. The optimum regularization parameter is

$$\hat{\lambda} = \underset{\lambda \in \{\lambda_1, \dots, \lambda_\mathrm{m}\}}{\operatorname{argmin}} \mathrm{CV}(\lambda) \tag{9}$$

The standard deviation of $CV(\lambda)$ could also be estimated. First, average binomial deviance in each fold

$$CV_k(\lambda) = \frac{1}{n_k} Dev_k(\lambda) \tag{10}$$

where: $\mathrm{n_k}$ is the number of points in the *k*th fold and Dev_λ is binomial deviance of Lasso model. The standard deviation of $CV(\lambda)$ is stated by (11)

$$\mathrm{SD}(\lambda) = \sqrt{var\left(\mathrm{CV_1}(\lambda), \dots, \mathrm{CV_K}(\lambda)\right)} \tag{11}$$

While the standard error of $CV(\lambda)$ is given by (2.12).

$$SE(\lambda) \leq \frac{SD(\hat{\lambda})}{\sqrt{K}} \tag{12}$$

2.3. Classification and Regression Tree (CART)

After *k* genes were selected, classification modeling was done using the Classification and Regression Tree (CART) algorithm which is also able to make a selection. CART algorithm constructs a decision tree by splitting the node binary, beginning with split the root node which contains the whole training data. Let *D* be the training set. Step 1, calculate the Gini index from *D* using formula (13).

$$Gini\ (D) = 1 - \sum_{i=1}^{m} p_i^2 \tag{13}$$

where p_i is the probability of a tuple in *D* belongs to class C_i. Step 2, determine the candidate of split-point. To obtain the candidate of split-point for continuous-valued variables, first the observation values of the j$^{\mathrm{th}}$ predictor, i.e. $\{x_{ij} | \, i = 1, \dots, n\}$, are sorted in increasing order and ordered observation values $\{x_{(i)j}\}$ are obtained. Then, the midpoint of two adjacent values is made as a candidate of split-point [19]. For example, the midpoint between the values $x_{(c)j}$ and $x_{(c+1)j}$ is $u_{cj} = \frac{x_{(c)j} + x_{(c+1)j}}{2}$. Step 3, for each possible split-point u_{cj}, calculate the Gini index of *D* if *D* is partitioned binary by t_{cj} using formula (14).

$$Gini_{u_{cj}}(D) = \frac{|D_1|}{|D|} Gini(D_1) + \frac{|D_2|}{|D|} Gini(D_2) \tag{14}$$

Where D_1 and D_2 are the subsets of the training data after being split by u_{jc}. Step 4, calculate the goodness of split $\Delta\, Gini(u_{cj})$ using formula (15).

$$\Delta\, Gini(u_{cj}) = Gini\ (D) - Gini_{u_{cj}}(D) \tag{15}$$

Step 5, repeat step 2 to step 4 for $j = 1, \dots, k$, then find the maximum $\Delta\, Gini(u_{cj})$. The candidate that has a maximum $\Delta\, Gini(u_{cj})$ is used as a split-point of the node. Finally, repeat step 1 to step 5 to split the child nodes until the maximum tree is obtained.

The splitting process that is carried out continuously will produce a complex tree and has many terminal nodes. Therefore, pruning the maximum tree is needed to get a simpler model and to avoid overfitting.

The parameter to measure the complexity of the tree is called the complexity parameter (CP). The smaller the CP, the more complex a tree. The CP value of the maximum tree is equal to zero.

To get the optimum tree, the 1-SE rule is used [10]. Let $\widehat{T}_t$ be the subtree that has minimum cross-validation error. The model chosen is the model that has cross-validation error less than or equal to $CV(\widehat{T}_t)$ plus one standard error $(SE(\widehat{T}_t))$ as stated by (16).

$$CV(T_t) \leq CV(\widehat{T}_t) + SE(\widehat{T}_t) \tag{16}$$

Where:

$\widehat{T}_t = \underset{T_t \in \{T_t, \ldots T_{max}\}}{\arg\min} \; CV_k(T_t)$,

$CV(\widehat{T}_t)$: minimum cross-validation error,

$SE(\widehat{T}_t)$: cross-validation standard error of $\widehat{T}_t$ and

$CV(T_t)$: cross-validation error of the tree selected.

2.4. Model Evaluation

The model compatibility was measured using the area under the ROC curve (AUC) and accuracy. ROC is a curve that measures classification performance with various thresholds. AUC value ranges between 0 and 1. The higher the AUC of a model, the better the model can predict. To create this curve, first, a scoring is made at each observation using probability which represents the degree to which an observation is a member of a class [20]. After obtaining the probabilities, various thresholds are then used. Threshold values range between 0 and 1. An observation that has a probability value of less than the threshold is given a value of 0 and if more than the threshold is given a value of 1. Then, the False Positive Rate (FPR) and True Positive Rate (TPR) are calculated. ROC curve is then created by plotting the FPR as the value of x and the TPR as the value of y. After the ROC curve is obtained, then AUC is calculated using the formula for the area of a trapezoid. Meanwhile, accuracy is calculated using formula (2.17).

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{17}$$

Where TP stands for true positive, TN stands for true negative, FP stands for false positive and FN stands for false negative.

The proposed framework is shown in Figure 1. We did all the processes using R version 3.6.1 with several packages, including glmnet for gene selection, rpart, and rpart.plot for constructing and plotting the CART model and ROCR for model evaluation.

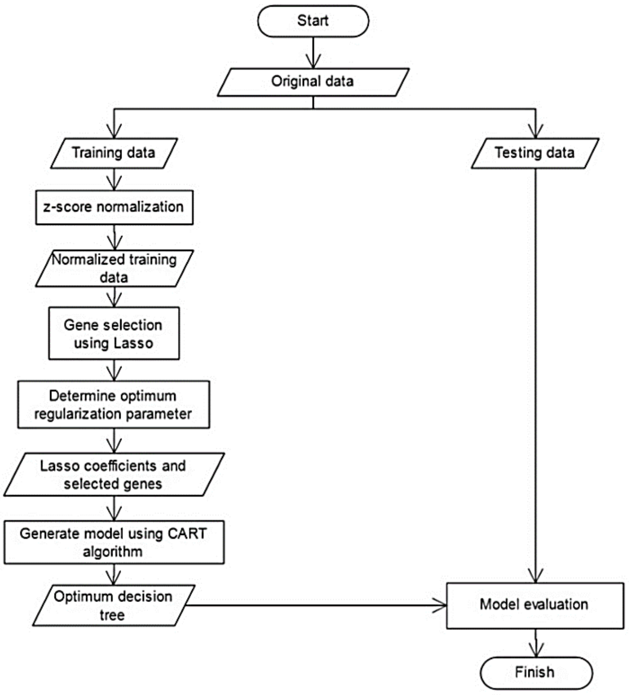

Start
Original data
Training data
Testing data
z-score normalization
Normalized training data
Gene selection using Lasso
Determine optimum regularization parameter
Lasso coefficients and selected genes
Generate model using CART algorithm
Optimum decision tree
Model evaluation
Finish

Figure 1. Flowchart of Two-stage Gene Selection using Lasso+CART

3. RESULTS AND DISCUSSION

To demonstrate the performance of the proposed method, we used the OVA_Breast dataset from GEMLeR. This dataset compares the gene expression of breast tumor tissues and other tumor tissues. The OVA_Breast dataset has 1,545 observations, where the "Breast" class has 344 instances and the "Other" class has 1201 instances. The number of predictor variables in this dataset is 10,936 which is genes.

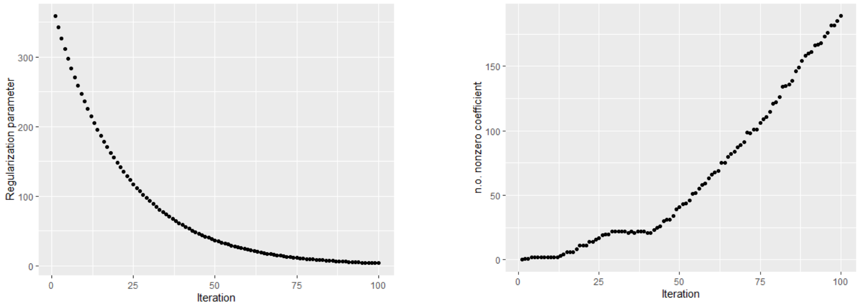
The dataset was divided into training and testing data with a ratio of 80:20 for training and testing data, respectively. So that the number of objects used for gene selection was 1236. For the training data, there were 274 instances in the Breast class and 962 instances in the Other class. While for the testing data, there were 70 instances in the Breast class and 239 instances in the Other class as described in Table 1.

Table 1. The Division of Dataset

	Breast	Other
Training data	274	962
Testing data	70	239

3.1. Gene Selection

The number of selected genes obtained depends on the regularization parameter. If the regularization parameter decreases, the number of selected genes will increase. The initial regularization parameter is that which makes all regression coefficients equal to zero. Meanwhile, the regularization parameter which is equal to zero produces regression coefficients from the full model. Figure 1(a) presents the plot of the regularization parameter for each iteration. The regularization parameter decreases with a ratio (r) of 0.955. This ratio was obtained from $r = \sqrt[M]{\frac{\lambda_M}{\lambda_0}}$, where $m = 1, 2, \ldots, M$ represented the iteration index (we used $M = 100$) and the glmnet specified $\lambda_M = 0{,}01 \times \lambda_0$. Figure 1(b) shows the number of nonzero coefficients for each iteration. It is seen that the more iteration indexes, the number of nonzero coefficients increases. In other words, the number of selected genes also increases.



(a) (b)

Figure 1. Regularization Parameters and the Number of Nonzero Coefficients for Each Iteration

3.2. Choosing optimum regularization parameter

Since the number of nonzero coefficients depends on the chosen regularization parameter, the optimum regularization parameter must be determined. In this study, we used 10-fold CV and the result is presented in Figure 2.

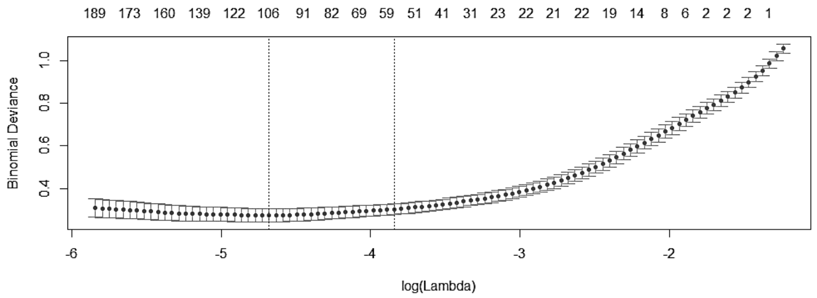


Figure 2. Cross-validation for Determining Optimum Regularization Parameter

In Figure 2, the left vertical line shows the smallest average binomial deviance which also indicates the optimum regularization parameter obtained. Meanwhile, the right vertical line shows the 1-SE [16]. The number above the picture represents the number of nonzero coefficients. Based on the result of cross-validation, it can be seen that the optimum regularization parameter produces a model with 106 nonzero coefficients with average binomial deviance and a standard deviation of 0.2748 ± 0.0306. The regularization parameter which produced 106 nonzero coefficients was 0.009284. Meanwhile, the regularization parameter in 1-SE was 0.021447 which produced 58 genes and can be used as an alternative limit.

3.3. CART Modeling

The first step in CART modeling is determining the Gini index from the training data. Let the training data is denoted by (D). Based on Table 1, the training data contains 274 observations in the "Breast" class and 962 observations in the "Other" class. The Gini Index for D is calculated using the formula (2.13) as follows.

$$Gini\ (D) = 1 - \left(\frac{274}{1236}\right)^2 - \left(\frac{962}{1236}\right)^2 = 0.3451$$

Because all predictor variables had continuous value, the candidate of split-point was obtained by sorting the value of each predictor variable from the smallest, then used the midpoint of two adjacent points as the candidate of split-point. Let A was a candidate of split-point, then $Gini_A(D)$ was calculated using the formula (2.14). Furthermore, the goodness of split of all split-point candidates was calculated using the formula (2.15). The point that had the greatest goodness of split was used as the point to split the root node. This splitting process was carried out until it was no longer possible to be split. The maximum classification tree from OVA_Breast training data is presented in Figure 3.

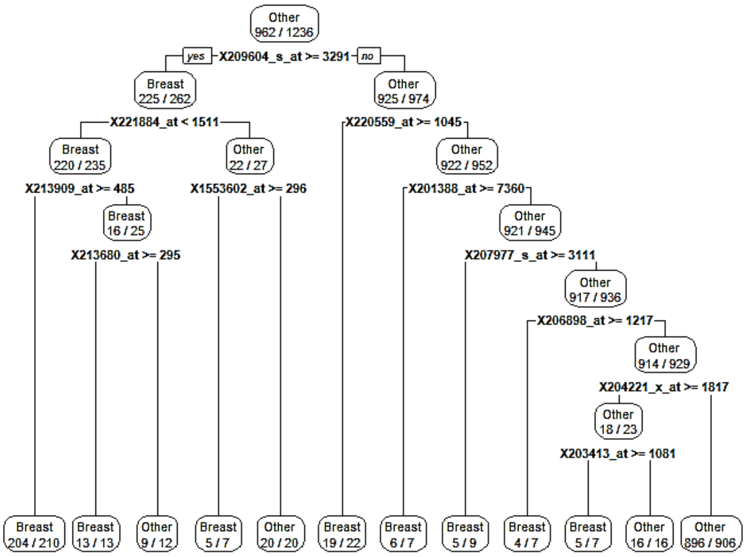


Figure 3. The Maximum Tree

Splitting the tree continuously would produce a complex tree as shown in Figure 3, which is called the maximum tree. It can be seen that the maximum tree has 12 terminal nodes. This model will be very good if used to re-model the training data. But when new data is used, this model will give low accuracy. This condition is called overfitting. To avoid overfitting, to simplify the tree and to interpret the model easily, pruning is needed. By pruning the maximum tree, an optimum subtree could be obtained. The result of cross-validation to get the optimum subtree is presented in Table 2.

Table 2. Complexity Parameter and the Result of Cross-validation

subtree	CP	nsplit	rel error	xerror	xstd
1	0.68613	0	1.00000	1.00000	0.05330
2	0.06204	1	0.31387	0.35401	0.03451
3	0.05839	2	0.25182	0.32847	0.03334
4	0.01825	3	0.19343	0.29927	0.03193
5	0.01095	4	0.17518	0.26277	0.03005
6	0.00456	7	0.14234	**0.25912**	0.02986
7	0.00000	11	0.12409	0.27372	0.03063

Notes:

nsplit	:	number of splits
CP	:	complexity parameter
rel error	:	relative error
xerror	:	cross-validation error
xstd	:	cross-validation standard error

The selection of the optimum subtree was based on the 1-SE rule. Based on Table 2, the 6th subtree has the smallest $\mathrm{CV}(\widehat{\mathrm{T}}_\mathrm{t})$ with a value of 0.25912. Using equation (2.16), the optimum subtree is determined as follows:

$$\mathrm{CV}(\mathrm{T_t}) \leq \mathrm{CV}(\widehat{\mathrm{T}}_\mathrm{t}) + \mathrm{SE}(\widehat{\mathrm{T}}_\mathrm{t})$$
$$\mathrm{CV}(\mathrm{T_t}) \leq 0.25912 + 0.02986$$
$$\mathrm{CV}(\mathrm{T_t}) \leq 0.28898$$

Based on Table 2, the 5th, 6th, and 7th subtree have a $\mathrm{CV}(\mathrm{T_t}) \leq 0.28898$. Therefore, these three subtrees can be selected as the optimum subtree. Determining the optimum subtree can also be done by looking at the complexity parameter plot showed in Figure 4.

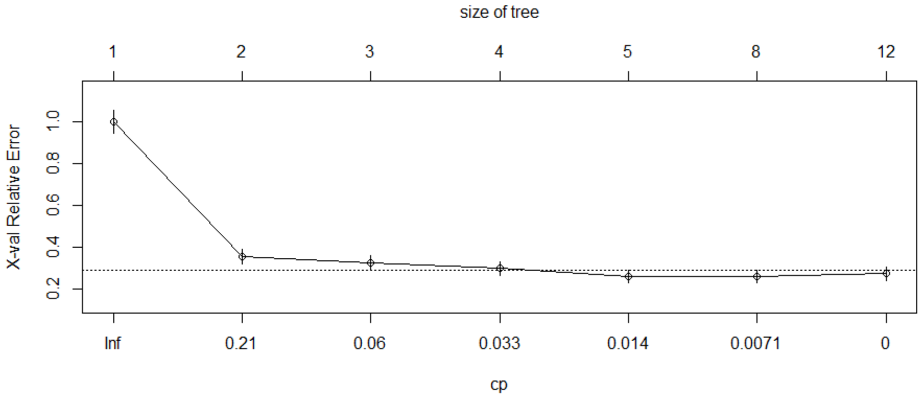


Figure 4. The Plot of Complexity Parameter for Determining Optimum Subtrees

The dashed horizontal line in Figure 4 shows the $\mathrm{CV}(\mathrm{T_t})$ with a value of 0.28898. The optimum subtree is a subtree that has a $\mathrm{CV}(\mathrm{T_t})$ below the dashed line, i.e. the 5th, 6th, and 7th subtree which produce 5, 8, and 12 terminal nodes respectively. To avoid overfitting, a subtree with 5 terminal nodes was chosen as the optimum tree which corresponding to CP = 0.01095.

Figure 5 presents the optimum subtree from the OVA_Breast dataset with the selected genes from Lasso as the predictor variables. In Figure 5, the genes are expressed in gene symbols, i.e. GATA3 (*GATA binding protein 3),* MECOM *(MDS1 and EVI1 complex locus),* EN1 *(engrailed homeobox 1),* and PSMD3 *(proteasome 26S subunit, non-ATPase 3).*

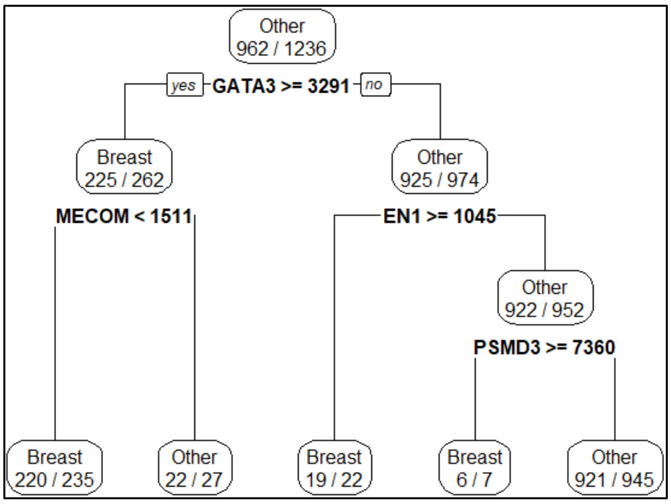


Figure 5. The Optimum Subtree Based on the Results of Cross-validation

In Figure 5, GATA3 split the root node indicates that the split-point of GATA3 maximizes the reduction in impurity (heterogeneity). Based on the model obtained, to determine whether a tumor tissue is a breast tumor or other tumor, first look at the expression of the GATA3 gene. If the expression of GATA3 is more than or equal to 3291, then the expression of MECOM is seen. Tumor tissues that have an expression of MECOM less than 1511 are predicted to be breast tumor tissues with a 93.6% probability. Whereas, if the MECOM expression is more than or equal to 1511, it will be predicted as other tumor tissues. When tissue has GATA3 expression more than 3291, EN1 gene expression is then seen. If the EN1 expression value is more than or equal to 1045, the tissue is predicted as breast tumor tissues. But if EN1 expression is less than 1045, PSMD3 expression is also seen. If the PSMD3 expression is more than or equal to 7360, the tissue will be predicted as a breast tumor tissue. Meanwhile, if the PSMD3 expression is less than 7360, the tissue will be predicted as other tumor tissue with a 97.5% probability.

The expression of GATA3 in breast cancer had been studied in numerous research, including [21], [22]. A high level of GATA3 expression indicates a slow rate of cell proliferation and low grade in breast tumors. The expression decreases with increasing tumor grade [21]. Low expression of GATA3 predicts a poor survival outcome in breast tumor patients [22]. The model obtained in this study was GATA3 has high expression in breast tumor patients, which means that the tumor tissue samples are taken from early-stage breast tumor patients.

Research [23] conducted pairwise comparisons to compare MECOM expression in various tissues under healthy conditions and tumors. MECOM expression decreased significantly in several carcinomas including lung, prostate, and breast carcinoma compared with healthy tissue. The results of these studies support the results of CART modeling in this study, where breast tumor patients have low MECOM expression.

The results of the study [24] showed that EN1 had very high expression in TNBC but not in other breast cancer subtypes. Meanwhile, PSMD3 expression in breast tumors has not been widely studied by scientists. PSMD3 might not be an important biomarker of breast tumors since based on the model obtained, PSMD3 expression will only be seen if GATA3 is low in expression and EN1 is also low in expression.

3.4. Model Evaluation

After obtaining the model as shown in Figure 5, the model was evaluated. Evaluation of the model is conducted using training data and new (testing) data to find out whether overfitting occurs or not. The metric to measure the goodness of the model was the area under the ROC curve which is often called the Area Under Curve (AUC). To make ROC curves we used the probabilities at the terminal node as the scores [20]. From these probabilities, then (FPR, TPR) was plotted with various thresholds. Table 3 and Table 4 show cutoffs (optimum thresholds), FPR, and TPR of training data and testing data respectively. And the ROC curves are displayed in Figure 6.

Table 3. Cutoff, FPR, and TPR of Training Data

Cutoff	FPR	TPR
Inf	0.0000	0.0000
0.9746	0.0876	0.9574
0.8148	0.1058	0.9802
0.1429	**0.1277**	**0.9813**
0.1364	0.1971	0.9844
0.0638	1.0000	1.0000

Table 4. Cutoff, FPR, and TPR of Testing Data

Cutoff	FPR	TPR
Inf	0.0000	0.0000
0.9746	0.0429	0.9540
0.8148	0.0429	0.9707
0.1364	**0.1000**	**0.9791**
0.0638	1.0000	1.0000

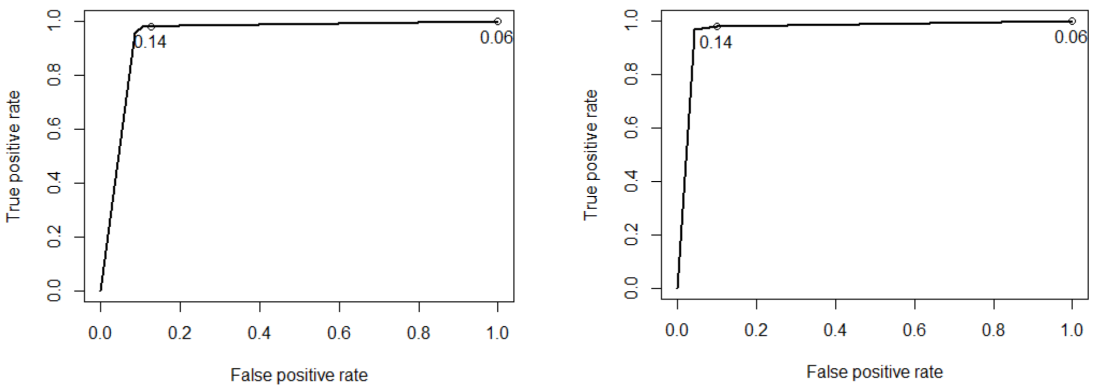


Figure 6. ROC Curve of the Model Evaluation on (a) Training Data and (b) Testing Data

Based on Figure 6, a good threshold value is around 0.1429 which has a True Positive Rate of around 98%, which means that 98% of breast tumor patients are correctly predicted and False Positive Rate of around 13%, which means that 13% of patients predicted to be breast tumors are another tumor patient. On the other hand, in the test data, when using a threshold of 0.1364, the True Positive Rate is around 98%, which means that 98% of breast tumor patients in the test data are correctly predicted. Meanwhile, the False Positive Rate is obtained around 10%, which means that 10% of patients predicted to be breast tumors are other tumor patients. However, the threshold is not a measure to know the goodness of the model. The measure to determine the goodness of the model is the AUC. AUC values for training data and test data are presented in Table 5.

Table 5. Result of Model Evaluation

Dataset	AUC
Training data	0,946
Testing data	0,967

Referring to Table 5, it can be said that the Lasso+CART method gave good results because the AUC value was very high close to 1, although it includes only four genes to construct the model. The model obtained was said to not overfit because it was not only good on training data but also gave a high AUC on the testing data. To compare the result obtained in this study to that of a previous study conducted by Assawamakin et.al. [25] which also used the OVA_Breast dataset, we also calculated the model accuracy using formula (2.17). The comparison of model accuracy is presented in Table 6.

Table 6. Comparison of Model Accuracy

Method	Number of selected features (genes)	Accuracy
NB-HNB [25]	15	0.94
SVM-RFE [25]	32	0.96
Lasso+CART	4	0.968

Based on Table 6, the accuracy of the model for testing data was 0.968. The accuracy from the model obtained by Lasso+CART is slightly greater than that of the model obtained in previous studies. But the number of selected genes is fewer than that of previous studies. Besides being able to produce a simple model with high accuracy, the two-stage gene selection method is also able to produce a model that is in line with the theory.

4. CONCLUSION

In this paper, a two-stage gene selection method has been proposed for dealing with the problem of classification in high-dimensional microarray data. In the OVA_Breast dataset, the two-stage gene selection could produce few numbers of selected genes but has high accuracy. The model acquired is also true according to theory. GATA3 which split root node is an important marker for breast cancer. GATA3 has high expression in breast tissue but low or even undetected in other tissues. High level of GATA3 expression related to early tumor grade. The expression of GATA3 decreases, as the tumor grade increases.

Many things can be done in further research. In this paper, Lasso is used as a regularization method for gene selection. However, Lasso produces a biased estimator. The use of the regularization method with other penalty functions such as elastic-net, adaptive Lasso, or relaxed Lasso can be applied instead of Lasso. This study is limited to the use of data with binary class. In future studies, multiclass data can be used. A simulation study can be carried out to further investigate how this proposed method performs on various data patterns.

5. ACKNOWLEDGMENTS

The authors would like to thank Prof. Widodo, S.Si., M.Si., Ph.D.Med.Sc., the head of Laboratorium Sentra Ilmu Hayati Universitas Brawijaya who has spared his time to discuss biomolecular with us.

6. REFERENCES

[1] I. Guyon and A. Elisseeff, "An Introduction to Variable and Feature Selection," *J. Mach. Learn. Res.*, vol. 3, pp. 1157–1182, 2003.

[2] S. Biswas, M. Bordoloi, and B. Purkayastha, "Review on Feature Selection and Classification using Neuro-Fuzzy Approaches," *Int. J. Appl. Evol. Comput.*, vol. 7, no. 4, pp. 28–44, 2016, doi: 10.4018/IJAEC.2016100102.

[3] H. Zhang, J. Wang, Z. Sun, J. M. Zurada, and N. R. Pal, "Feature Selection for Neural Networks Using Group Lasso Regularization," *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 4, pp. 659–673, 2020, doi:10.1109/TKDE.2019.2893266

[4] R. Tibshirani, "Regression Shrinkage and Selection via the Lasso," *J. R. Stat. Soc. Ser. B*, vol. 58, no. 1, pp. 267–288, 1996.

[5] S. Tateishi, H. Matsui, and S. Konishi, “Nonlinear regression modeling via the lasso-type regularization,” *J. Stat. Plan. Inference*, vol. 140, no. 5, pp. 1125–1134, 2010, doi: 10.1016/j.jspi.2009.10.015.

[6] Y. Fan and C. Y. Tang, “Tuning parameter selection in high dimensional penalized likelihood,” *J. R. Stat. Soc. Ser. B (Statistical Methodol.*, vol. 75, pp. 531–552, 2013.

[7] K. Hirose, S. Tateishi, and S. Konishi, “Tuning parameter selection in sparse regression modeling,” *Comput. Stat. Data Anal.*, vol. 59, pp. 28–40, 2013, doi: 10.1016/j.csda.2012.10.005.

[8] Z. Y. Algamal and M. H. Lee, “Penalized Logistic Regression with the Adaptive LASSO for Gene Selection in High-Dimensional Cancer Classification,” *Expert Syst. Appl.*, vol. 42, no. 23, pp. 9326–9332, 2015.

[9] C. Kang, Y. Huo, L. Xin, B. Tian, and B. Yu, “Feature Selection and Tumor Classification for Microarray Data Using Relaxed Lasso and Generalized Multi-class Support Vector Machine,” *J. Theor. Biol.*, 2018, doi: 10.1016/j.jtbi.2018.12.010.

[10] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, *Classification and Regression Trees*. Chapman and Hall, 1984.

[11] H. Jiang, W. Zheng, L. Luo, and Y. Dong, “A two-stage minimax concave penalty based method in pruned AdaBoost ensemble,” *Appl. Soft Comput. J.*, vol. 83, 2019, doi: 10.1016/j.asoc.2019.105674.

[12] B. J. Friedman, T. Hastie, and H. Holger, “Pathwise Coordinate Optimization,” *Ann. Appl. Stat.*, vol. 1, no. 2, pp. 302–332, 2007, doi: 10.1214/07-AOAS131.

[13] J. Friedman, T. Hastie, and R. Tibshirani, “Regularization Paths for Generalized Linear Models via Coordinate Descent,” *J. Stat. Softw.*, vol. 33, no. 1, 2010.

[14] R. Mazumder, J. H. Friedman, and T. Hastie, “SparseNet: Coordinate Descent With Nonconvex Penalties,” *J. Am. Stat. Assoc.*, vol. 106, no. 495, pp. 1125–1138, 2011, doi: 10.1198/jasa.2011.tm09738.

[15] R. Tibshirani, J. Bien, J. Friedman, T. Hastie, N. Simon, J. Taylor, and R. J. Tibshirani, “Strong Rules for Discarding Predictors in Lasso-type Problems,” *J. R. Stat. Soc. Ser. B*, vol. 74, pp. 245–266, 2012.

[16] T. Hastie, R. Tibshirani, and M. Wainwright, *Statistical Learning with Sparsity: The Lasso and Generalizations*. Chapman and Hall, 2015.

[17] A. Agresti, *Categorical Data Analysis*, Second Edi. Wiley-Interscience, 2002.

[18] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning DataMining, Inference, and Prediction*, Second Edi. California: Springer, 2009.

[19] J. Han, M. Kamber, and J. Pei, *Data Mining Concepts and Techniques Third Edition*. Elsevier, 2012.

[20] T. Fawcett, “An Introduction to ROC Analysis,” *Pattern Recognit. Lett.*, vol. 27, pp. 861–874, 2006.

[21] T. Shaoxian, Y. Baohua, X. Xiaoli, C. Yufan, T. Xiaoyu, L. Hongfen, B Rui, S. Xiangjie, S. Ruohong, and Y. Wentao, “Characterisation of GATA3 expression in invasive breast cancer : differences in histological subtypes and immunohistochemically defined molecular subtypes,” *J Clin Pathol*, vol. 15, pp. 1–9, 2017.

[22] H. Liu, J. Shi, M. L. Wilkerson, and F. Lin, “Immunohistochemical Evaluation of GATA3 Expression in Tumors and Normal Tissues: A Useful Immunomarker for Breast and Urothelial Carcinomas,” *Am J Clin Pathol*, vol. 138, pp. 57–64, 2012.

[23] D. Ivanochko, L. Halabelian, E. Henderson, P. Savitsky, H. Jain, E. Marcon, S. Duan, A. Hutchinson, A. Seitova, D. Barsyte-Lovejoy, P. Filippakopoulos, J. Greenblatt, E. Lima-Fernandes, and C. H. Arrowsmith, “Direct interaction between the PRDM3 and PRDM16 tumor suppressors and the NuRD chromatin remodeling complex,” *Nucleic Acids Res.*, vol. 47, no. 3, pp. 1225–1238, 2019, doi: 10.1093/nar/gky1192.

[24] Y. J. Kim, M. Sung, E. Oh, M. Van Vranckena, J. Song, K. Jung, and Y. Choi, “Engrailed 1 overexpression as a potential prognostic marker in quintuple-negative breast cancer,” *Cancer Biol. Ther.*, vol. 19, no. 4, pp. 335–345, 2018, doi: 10.1080/15384047.2018.1423913.

[25] A. Assawamakin, S. Prueksaaroon, S. Kulawonganunchai, P. J. Shaw, Vara, Varavithya, T. Ruangrajitpakorn, and S. Tongsima, “Biomarker Selection and Classification of “‘ - Omics ”’ Data Using a Two-Step Bayes Classification Framework,” *Biomed Res. Int.*, 2013, doi: 10.1155/2013/148014.