Comparison of Support Vector Machine Recursive Feature Elimination and Kernel Function as feature selection using Support Vector Machine for lung cancer classification

Z Rustam and S A A Kharis
Department of Mathematics, Faculty of Mathematics and Natural Sciences, Universitas Indonesia, Depok 16424, Indonesia
Corresponding author’s e-mail: rustam@ui.ac.id

Abstract. Cancer is the uncontrolled growth of abnormal cell that need a proper treatment. Cancer is second leading cause of death according to the World Health Organization in 2018. There are more than 120 types of cancer, one of them is lung cancer. Cancer classification has been able to maximize diagnosis, treatment, and management of cancer. Many studies have examined the classification of cancer using microarrays data. Microarray data consists of thousands of features (genes) but only has dozens or hundreds of samples. This can reduce the accuracy of classification so that the selection of features is needed before the classification process. In this research, the feature selection methods are Support Vector Machine Recursive Feature Elimination (SVM-RFE) and Kernel Function and the classification method is Support Vector Machine (SVM). The results showed SVM using SVM-RFE as feature selection is better than SVM method without using feature selection and Gaussian Kernel Function.

Keywords: Kernel function, lung cancer, recursive feature elimination, support vector machine.

1. Introduction
Cancer is a term for disease in which cells divide abnormally without control and can invade surrounding tissue. Each year more than 1.1 million of people die cause of lung cancer [1]. Lung cancer can be included to solid tumor categories or abnormal tissues which is solid form [2]. One of the main causes of lung cancer is prolonged exposure or inhalation of a carcinogenic substance [3]. Lung cancer is divided into two types namely, non-small cell lung cancer (NSCLC) and small cell lung cancer (SCLC). The division is based on growth patterns and different medical treatments. NSCLC is subdivided into three cancer subtypes, namely Adenocarcinoma, Squamous cell carcinoma, and Large cell carcinoma. SCLC is the most aggressive and rapidly developing type of lung cancer. SCLC is closely related to smoking, with only 1% of all cases occurring in nonsmokers. The treatment given by the medical of the two types of cancer is different. NSCLC is usually treated surgically while SCLC generally responds better to chemotherapy and radiotherapy.

The detection of lung cancer can be done in various ways, namely history taking, physical examination, anatomic pathology examination, laboratory examination, imaging examination, special examination, and other examinations. In general, detection of lung cancer is carried out using radiological imaging techniques through imaging tests. However, these techniques still result low survival rates cause detect the malignant cells in late stage of lung cancer [4]. Another method for lung classification by using a test image or usually known as image segmentation. Research on cancer classification by using image segmentation previously been done by the method Fuzzy C-Mean Possibilistic; Clustering K-Nearest Neighbor; and Fuzzy Clustering and Support Vector Machine..
Cancer classification has improved for many years to detect cancer at early stage. One of them is using microarray data for lung cancer classification. The advances in molecular biology can be applied to detect the formation of cancer in earlier stages using the information of RNA, DNA and proteins. Several cancer studies using microarray data are also been done by the method Fuzzy C-Means and Possibilistic C-Means [5]. Microarray data shows human genes expressions data on specific part of body numerically. Given some microarray data that already classified manually from patient data, the data will make clues or models to the machine to classify other microarray data on the same issues. An important problem in the classification using microarray data is that there are many features (in this case in the form of genes) but a large number of samples. The number of genes can reach tens of thousands while the number of samples is only in the tens or hundreds. This can add to the data processing costs needed. In addition, irrelevant features can reduce classification accuracy. Therefore, before carrying out the classification process, the selection of features is done first. Other characteristic of the data is it does not have many samples [6].

The feature selection method is divided into two categories, namely independent classifier and dependent classifier. The independent classifier is also called the filter method. The filter method does not depend on any classifier method, that is, the filter method does not use the classifier method to determine the features to be selected and assess the relevance of the features by only looking at the intrinsic nature of the data. The dependent classifier is divided into two methods, namely wrapper and embedded. The basic idea of the wrapper method is to search through a subset of feature spaces with input of all the features of the dataset. The wrapper method uses the overall work evaluation of the classifier or its algorithm to find possible interactions between variables by measuring the accuracy of the classifier's predictions. In this paper, the feature selection using Support Vector Machine Recursive Feature Elimination (SVM-RFE) and Kernel Function and the classification method using Support Vector Machine (SVM).

2. Materials and methods

2.1. Microarray data

This paper use lung cancer microarray data form of lung cancer. Microarray data is a data containing the value expression gene called with features. The data obtained from Kent Ridge Bio-medical Dataset from Michigan (http://datam.i2r.astar.edu.sg/datasets/krbd/). Data lung cancer will be tested has 7129 features and divided into two classes, namely cancer with a sample of 86 and 10 non-cancer samples.

2.2. Feature selection based on Kernel Function

Kernel function was invented by Vapnik [7], developed by Scholkopf et al. [8] and Christianini and Taylor [9]. Kernel functions using a linear classifier to solve a linear or non-linear problem into a higher dimensional space. The dot product \( x_k^T x_i \) is mapped to \( \Phi(x_k)^T \Phi(x_i) \). Kernel function from Theorem of Mercer defined as follows [10]:

\[
K(x_k^T x_i) = \Phi(x_k)^T \Phi(x_i)
\]  

(1)

Kernel function has several types. There are gaussian, polynomial, and the others. Euclidean distance can be calculated as follows:

\[
\|\Phi(x_k) - \Phi(x_i)\|^2 = (\Phi(x_k) - \Phi(x_i))^T (\Phi(x_k) - \Phi(x_i)) = K(x_k, x_k) - 2K(x_k, x_i) + K(x_i, x_i)
\]  

(2)

To measure the function dissimilarity in feature selection using a Gaussian Kernel, equation (2) was applied as below:

\[
K(x_k^T x_i) = exp\left(-\frac{\|x_k - x_i\|^2}{2\sigma^2}\right)
\]  

(3)

The main idea of kernel function as feature selection is find weight each features to optimize the objective function \( g \). Dataset \( X \in \mathbb{R}^{n \times p} \), \( n \) is number of samples and \( p \) is number of genes can be calculated using \( w_k = \frac{1}{p} \) which \( k = 1, 2, ..., p \) first. The dataset will be used of course should be labeled to be classified, \( \gamma_j \in Y \) which \( Y = 1, 2, ..., C \). The \( C \) class will be treated as a cluster, so the cluster center \( v_i = [v_{i1}, v_{i2}, ..., v_{ip}] \) can be calculated using equation (4):

\[
\]
\[
v_{ik} = \frac{\sum_{x_j \in C_i} x_{jk}}{|C_i|}, \quad i = 1, 2, \ldots, C, j = 1, 2, \ldots, n
\]  

(4)

\(|C_i|\) is the number of samples contained in class \(C_i\). In kernel function, dissimilarity measure of feature selection will be calculated. The function of dissimilarity between the sample and the center of the cluster obtained by using kernel written in the following equation (5):

\[
\Phi^2 \left( x_j, v_i \right) = \sum_{k=1}^{p} ||\Phi \left( x_k \right) - \Phi \left( x_i \right)||^2
\]

\[
= \sum_{k=1}^{p} \left\{ K(x_k, x_k) - 2 K(x_k, x_j) + K(x_j, x_j) \right\}
\]

(5)

Using a Gaussian Kernel in equation (3), then the distance in equation (4) can be written as:

\[
||\Phi \left( x_k \right) - \Phi \left( x_i \right)||^2 = 2 \left( 1 - K(x_k, v_i) \right) = 2 \left( 1 - \exp \left( -\frac{||x_k - x_i||^2}{2\sigma^2} \right) \right)
\]

(6)

The objective function similar with the objective function in SCAD as feature selection algorithm as, shown in equation [8]:

\[
\min J = \sum_{i=1}^{C} \sum_{x_j \in C_i} \Phi^2 \left( x_j, v_i \right) + \delta \sum_{k=1}^{p} w_k^2
\]

\[
= \sum_{i=1}^{C} \sum_{x_j \in C_i} \sum_{k=1}^{p} w_k \left\{ \Phi \left( x_j \right) - \Phi \left( v_{jk} \right) \right\}^2 + \delta \sum_{k=1}^{p} w_k^2
\]

(7)

where \(w = (w_1, \ldots, w_p)\) subject to

\[
\begin{align*}
& w_k \in [0,1], k = 1, \ldots, p, \\
& \sum_{k=1}^{p} w_k = 1
\end{align*}
\]

(8)

Minimize the objective function with make the value of \(\delta\) and \(w_k\) lowest. So, the values of \(\delta\) and \(w_k\) need to be update to find the lowest of \(\delta\) and \(w_k\). The objective function for updating shown in equation (9) and equation (10):

\[
w_k = \frac{1}{p} + \frac{1}{2\delta} \sum_{i=1}^{C} \sum_{x_j \in C_i} \left( \frac{\sum_{k=1}^{p} \left\{ \Phi \left( x_j \right) - \Phi \left( v_{jk} \right) \right\}^2}{p} - \left\| \Phi \left( x_j \right) - \Phi \left( v_{jk} \right) \right\|^2 \right)
\]

(9)

\[
\delta^t = \frac{\alpha \sum_{i=1}^{C} \sum_{x_j \in C_i} \sum_{k=1}^{p} \left( w_k^{-t} \right)^{-1} \left\{ \Phi \left( x_j \right) - \Phi \left( v_{jk} \right) \right\}^2}{\sum_{k=1}^{p} \left( w_k^{-t} \right)^{-1}} , \quad \alpha \text{ constant}
\]

(10)

Algorithm 1 show the Kernel function algorithm as feature selection[10],

**Algorithm 1.** Kernel function as feature selection

Input: Dataset \(X\), Label \(y\), \(\alpha = 0.5\), \(\theta = 10^{-6}\)

Output: each gene weight

Step 1. Initialization

- Count \(w_k^0 = \frac{1}{p}\) to find the weight of each genes (k)
- Using equation (4) to find \(v_i\)
- Using equation (5) to find the distance,

Step 2. Using equation (10) to find the value of \(\delta^t\)

Step 3. Update weight of each gene using equation (9)

Step 4. Using equation (7) to find \(J\)

Step 5. Determine iteration stopping for centroid of current iteration \((t)\) and previous iteration \((t-1)\)

\[
\Delta = \left\| J^t - J^{t-1} \right\|
\]

If \(\Delta < \theta\), then iteration stops. If \(\Delta > \theta\), then go to step 2
2.3. Feature selection based on SVM-RFE

SVM-RFE was introduced for feature selection by Guyon in 2002. The SVM-RFE method is a recursive feature elimination application that uses SVM weights as a ranking criterion. SVM-RFE is classified as an embedded method. SVM-RFE widely used for gene selection and several improvements have been recently suggested [11–13]. The algorithm for SVM-RFE is shown in Algorithm 2 [11]. The main principle of SVM-RFE is to eliminate features that have the lowest weight squares in each iteration. The procedure for recursive feature elimination in general is:

a. Perform classifier training to find the weight vector \( w \)

b. Calculate ranking criteria for all features

c. Eliminate features with the smallest ranking criterion value

Features used in the iteration are eliminated with backward feature elimination. The ranking score is given by the components of the weight vector \( w \) of the SVM as follows:

\[
w = \sum_k \alpha_k y_k x_k \tag{11}\]

Algorithm 2. Feature Selection based on SVM-RFE

Input : training data \( \{x_i, y_i\}_{i=1}^N \)

Output : ranked feature list \( R \)

Initialize : \( S = \{1, 2, \ldots, D\} \)

\( R = \emptyset \)

While \( S \) is not empty, do:

1. Restrict the features of \( X_f \) to the remaining \( S \)
2. Train SVM to get weight vectors
3. Calculate the ranking criteria \( c_k = w_k^2, k = 1, \ldots, |S| \)
4. Look for features with the smallest value of \( c_k \), called feature \( p \)
5. Add feature \( p \) into \( R \) (\( R = \{p\} \cup R \))
6. Remove feature \( p \) from \( S \) (\( S = S \setminus p \))

2.4. Support Vector Machine (SVM)

Support Vector Machine (SVM) is developed by Boser, Guyon, Vapnik in 1992 [7]. The purpose of SVM is to find an optimal hyperplane, which is a hyperplane that has a maximum margin [14]. There will be more than one hyperplane that can divide data into two classes. SVM will choose hyperplane with maximum margin. The primal optimization problem can be written as:

\[
\min \frac{1}{2} \|w\|^2 + C \sum_{i=1}^{N} \xi_i \tag{12}
\]

\[
s.t. \ y_i (w \cdot x_i + b) \geq 1 - \xi_i, i = 1, 2, \ldots, N \\
\xi_i \geq 0, i = 1, 2, \ldots, N
\]

where the parameter \( C > 0 \) will control the trade-off between the amount of data in the wrong class (narrowed margin) and generalization capabilities (margin widened). \( C \) values are not obtained in the learning process but must be determined before learning. \( \xi_i \) is the slack variable. The slack variable enables misclassification at some distances. The slack variable is formulated as \( \xi_i = |t_i - f(x_i)| \). The slack variable is formulated as \( \xi_i = |t_i - f(x_i)| \). If \( \xi_i = 0 \) then \( x_i \) lies in the margin and is classified correctly, and if \( 0 < \xi_i \leq 1 \) then \( x_i \) lies in the margin but is still classified in the correct class, whereas if \( \xi_i > 1 \) then \( x_i \) is classified in the wrong class.

\[
\min \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j (x_i \cdot x_j) - \sum_{i=1}^{N} \alpha_i \tag{13}
\]

\[
s.t. \sum_{i=1}^{N} y_i \alpha_i = 0, 0 \leq \alpha_i \leq C, i = 1, 2, \ldots, N
\]

\( \alpha_i \) is the Lagrange multiplier. The weight vector is expressed as \( w = \sum_{i=1}^{N} \alpha_i y_i x_i \).
Table 1. Confusion matrix

| Actual Value | Predicted Class |       |       |
|--------------|----------------|-------|-------|
| Class 1      | TP             | FN    |       |
| Class 2      | FP             | TN    |       |

*) TP (true positive): the amount of class 1 test data predicted as class 1,
TN (true negative): the amount of class 2 test data predicted as class 2,
FP (false positive): the amount of class 2 test data predicted as class 1,
FN (false negative): the number of class 1 test data predicted as class 2.

Table 2. Example of lung cancer data

|          | A28102_at (nucleotide) | AB000114_at (osteonomalgin) | ... | Z97074_at (RABEPK) |
|----------|------------------------|-----------------------------|-----|------------------|
| no       |                        |                             |     | 7129             |
| X1       | 100                    | 69.4                        | ... | 276              |
| X2       | 59.7                   | 18.1                        | ... | 134.7            |
| ...      | ...                    | ...                         | ... | ...              |
| X86      | 57                     | 7.9                         | ... | 50.7             |
| ...      | ...                    | ...                         | ... | ...              |
| X96      | 106.2                  | 164                         | ... | 140.5            |

Note: A28102, AB000114_at, and Z97074_at show features in dataset.
Numbers in each row represents the expression of a gene across all experiments.
X1, X2, ..., X96 show samples.

2.5. Performance measures
Measurement of the ability of a classification model is based on the amount of data that is classified correctly or incorrectly. The amount of data classified correctly or incorrectly can be presented in more detail using the confusion matrix. Confusion matrix for this research is shown in table 1 [15].

3. Experimental results

3.1. Data overview
The data consists of 7129 genes and each gene has 96 samples where 86 are cancer and 10 are non-cancer. The column of data represents features or genes associated with lung cancer. The row of data represents the number of patient samples. The sample data of cancer patients in 1–96 line and remain lines is non-cancer patients. The table below is a form of data that will be used in the process of cancer classification as shown in table 2.

3.2. Results
In this research, first, we make a new data with select 10, 15, 25, 50, 75, 100, 150, 200, 500, 1000 features selected and classify each data. After that, we do classification without feature selection or using all genes in the data was done. SVM will be evaluate by k-fold cross validation with k = 6. The results are shown in table 3.
4. Conclusions
From the result and testing, it can be seen that the accuracy of using a feature selection based SVM-RFE method reached 96.7899% accuracy in usage 5 best features, this percentage slightly more superior than SVM method without using feature selection and using a feature selection base Kernel Function especially Gaussian. Thus, it can be said SVM by using feature selection based SVM-RFE reached 96.7899% accuracy in usage 5 best features, this percentage slightly more superior than SVM method without using feature selection and using a feature selection base Kernel Function especially Gaussian. Except that, average of the accuracy using Gaussian Kernel below the average accuracy using SVM-RFE. Suggestion from the authors can also be discussed using other classification and other kernel types.

Acknowledgements
This research is supported by Universitas Indonesia under PITTA UI 2017 research grant.

References
[1] America Cancer Society 2011 Cancer Facts & Figures (Atlanta: American Cancer Society)
[2] Cornfield J, Haenszel W, Hammond E C, Lilienfield A M, Shimkin M B and Wynder E L 2009 *Int. J. Epidemiol.* **38** 1175–91
[3] Mountain C F and Dresler C M 1997 *Chest* **111** 1718–23
[4] Jeffrey A T, Hagen J A, Carpenter C L and Laird-Offringa I A 2002 *Oncogene* **21** 5450–61
[5] Wibowo A P *Classification of Brain Cancer (Astrocytoma) Using Independent Component Analysis dan Possibilistic c-Means* (Depok: Universitas Indonesia) Undergraduate Thesis
[6] Golub T R et al. 1999 *Science* **286** 531–7
[7] Vapnik V N 1998 *Statistical Learning Theory* (New York: John Wiley & Sons Inc)
[8] Schölkopf B, Smola A, and Müller K R 1998 *Neural Comput.* **10** 1299–1319
[9] Cristianini N and Shawe-Taylor J 2000 *An Introduction to Support Vector Machines and Other Kernel Based Learning Methods* (Cambridge: Cambridge University Press) pp 1299–1319
[10] Rustam Z and Talita A S 2015 *J. Theor. Appl. Inf. Technol.* **80** 147–51
[11] Kai-Bo D, Rajapakse J, Wang H and Azuaje F 2005 *IEEE Trans. Nanobiosci.* **4** 228–34
[12] Tang Y, Zhang Y Q and Huang Z 2007 *IEEE Trans. Comput. Biol. Bioinformatics* **4** 365–381
[13] Tang Y, Zhang Y Q, Huang Z, Hu X and Zhao Y 2008 *IEEE Trans. Inf. Technol. Biomed.* **12** 723–30
[14] Guyon I, Weston J, Barnhill S and Vapnik V 2002 *Mach. Learn.* **46** 389–422
[15] Mert A, Kılıc N and Akan A 2011 *Proc. 53rd International Symposium ELMAR 2011* Zadar, Croatia (Piscataway: IEEE) pp 37–40
[16] Rahideh A and Shaheed M H 2011 *Proc. International Conference on Control, Instrumentation and Automation (ICCIA)* 27–29 December 2011 Shiraz Iran (Piscataway: IEEE) pp 1175–80

### Table 3. Comparison accuracy Kernel Function and SVM-RFE as feature selection

| Number of features | Gaussian Kernel (in %) | SVM-RFE (in %) |
|--------------------|-----------------------|---------------|
| 0                  | 89.58                 | 89.58         |
| 5                  | 74.3682               | 96.7899       |
| 10                 | 89.6364               | 91.3800       |
| 15                 | 89.6541               | 89.610        |
| 25                 | 89.6453               | 89.6541       |
| 50                 | 89.6364               | 89.6364       |
| 75                 | 89.6895               | 89.6453       |
| 100                | 89.6541               | 89.6541       |
| 150                | 89.6364               | 89.6453       |
| 200                | 89.6541               | 89.6623       |
| 500                | 89.6630               | 89.6718       |
| 1000               | 89.6801               | 89.6276       |