International Journal of Electrical and Computer Engineering (IJECE)
Vol. 8, No. 5, October 2018, pp. 3194~3203
ISSN: 2088-8708, DOI: 10.11591/ijece.v8i5.pp3194-3203

Effect of Feature Selection on Gene Expression Datasets Classification Accuracy

Hicham Omara, Mohamed Lazaar, Youness Tabii
University Abdelmalak Essaadi, National School of Applied Sciences Tetuan, Morocco

Article Info

Article history:

Received Dec 5, 2017
Revised Jan 4, 2018
Accepted Sep 17, 2018

Keyword:

Accuracy
Classification
Feature selection
Microarray gene expression

ABSTRACT

Feature selection attracts researchers who deal with machine learning and data mining. It consists of selecting the variables that have the greatest impact on the dataset classification, and discarding the rest. This dimentionality reduction allows classifiers to be fast and more accurate. This paper traits the effect of feature selection on the accuracy of widely used classifiers in literature. These classifiers are compared with three real datasets which are pre-processed with feature selection methods. More than 9% amelioration in classification accuracy is observed, and k-means appears to be the most sensitive classifier to feature selection

Corresponding Author:

Hicham Omara,
National School of Applied Sciences Tetuan,
University Abdelmalak Essaadi, Morocco.
Email: hichamomara@gmail.com

1. INTRODUCTION

In recent years, the data is exponentially expanded, so their characteristics, therefore, reducing the size of the data by removing variables that are irrelevant or that are redundant and selecting only the most significant according to some criterion has become a requirement before any classification, this reducing should give the best performance according to some objective function [1]-[5]. DNA microarray technology has the ability to study thousands of genes simultaneously in a single experiment. This technology provides a large amount of data from which much knowledge can be processed. A set of microarray gene expression data can be represented in tabular form, in which each line represents a particular gene, each column a sample and each entry of the matrix is the measured level of expression gene in a sample. Researchers have a database of more than 40,000 gene sequences that they can use for this purpose. Unfortunately, the enormous size of DNA microarray causes a problem when it treated by clustering or classification algorithms such as SOM, K-means, KNN … or other; so pre-processing the data beforehand by reducing its size becomes a necessity. Feature selection consists of choosing a subset of input variables and deleting redundant or irrelevant entities from the original dataset. Consequently, the execution time for classification the data decreases, and the accuracy increases [6].

Feature selection algorithms are divided into three categories; filters, wrappers and embedded or hybrid selectors [7], [8]. The filters extract features from the data without any learning involved by ranking all features and chosen top ones [9]-[11]. There were several and widely used filter in literature, like: Information Gain (IG) [12] that ranks features based on a relevancy score which is based on each individual attribute. Correlation-based Feature Selection (CFS) algorithms looks for features that are highly correlated with the class which has no or minimal correlation with each other (Hall, 2000). Minimum Redundancy Maximum Relevance (mRMR) [8] that maximizes the relevancy of genes with the class label and minimizes

the redundancy in each class using Mutual Information (MI) mesures. Relief F is also widely used with cancer microarray data [13]; it detects features which are statistically relevant to the target concept.

The wrappers uses classifying algorithm to evaluate which features are useful; it means that the features were selected taking the classification algorithm into account [14]. Many researches have applied wrappers selector, like study of Gheyas and Smith that proposed a new method named simulated annealing generic algorithm (SAGA), which incorporates existing wrapper methods into a single solution [15]. LDA-based Genetic Algorithm (LDA-GA) proposed by Huerta et al in [16]; this method applied t-statistic filter to retain a group of p top ranking genes, and used the LDA-based GA. Leave-one-out calculation sequential forward selection (LOOCSFS) algorithm that combine the leave-one-out calculation measure with the sequential forward selection scheme proposed by Tang et al [17]. Genetic Algorithm-Support Vector Machine (GA-SVM) creates a population of chromosomes as binary strings that represent the subset of features that are evaluated using SVMs developed by Perez and Marwala in [18].

The third field of feature selection approaches is embedded methods. It takes advantage of the two models by using their different evaluation criteria in different search stages [19]. In this case we can cite the most widely applied embedded techniques based on support vector machine based on Recursive Feature Elimination (SVM-RFE) for gene selection and cancer classification proposed by Guyon et al. in [20]. Maldonado et al. proposed an embedded approach called kernel-penalized SVM (KP-SVM) by introducing a penalty factor in the dual formulation of SVM [21]. Mundra et al. hybridized two of the most popular feature selection approaches: SVM-RFE and mRMR [22]. Chuang et al. proposed a hybrid approach that hybridize correlation based feature selection (CFS) and Taguchi-Genetic Algorithm (TGA) and used KNN as the classifier with the leave-one-out cross-validation (LOOCV) [23]. Lee and Liu [24] proposed an approach called Genetic Algorithm Dynamic Parameter (GADP) for producing every possible subset of genes and rank the genes using their occurrence frequency.

Therefore, this paper attempts to present a review of widely used feature selection techniques focusing on cancer classification. In addition, other tasks related to microarray data analysis also have been revealed such as missing values, normalization and discretisation. Furthermore, commonly used classification methods were discussed. This study evaluated five different filter algorithms: Random forest, information gain and chi-squared on three cancer datasets; and evaluated their effect on three classification algorithm: SOM, KNN, K-means and Random Forest.

2. METHOD AND MATERIALS

2.1. General Bachground

Analysis of gene expression data is primarily based on comparison of gene expression profiles. To do these, we need a measure to quantify the similarity between genes in expression profiles. A variety of distance measures can be used to compute similarity. In this section, a description of most metrics used is discussed. The gene expression data from microarray experiments is usually in the form of large matrices $G_{(n+1)\times m}$ of expression levels of genes $g_1, g_2, \dots, g_n$ under different experimental conditions $s_1, s_2, \dots s_m$ and the last row contains the label Y of each sample, their values $y_j \in \{-1,1\}$. Each element $G[i,j]$, denoted as g_{ij}, represents the expression level of the gene g_i in the sample s_j (see Table 1). The expression profile of a gene *i* can be represented as a row vector: $gi = (g_{i1}, g_{i2}, .., g_{im})$ as follow:

$$G = \begin{pmatrix} g_1 \\ \vdots \\ g_n \\ \overline{Y} \end{pmatrix} = \begin{pmatrix} g_{11} & \cdots & g_{1m} \\ \vdots & \ddots & \vdots \\ g_{n1} & \cdots & g_{nm} \\ \overline{y_1} & \cdots & \overline{y_m} \end{pmatrix}$$

Table 1. Microarray Dataset Example

Genes\Samples	S_1	S_2	S_3	S_4	...	S_m
g_1	56,23	43,74	4,18	9,5	...	34,18
g_2	33,54	30,5	4,71	32,18	...	43,71
g_3	13	29,09	4,13	2,88	...	49,13
g_4	64,25	70,24	76,1	31,4	...	36,91
⋮	⋮	⋮	⋮	⋮		⋮
g_n	3,54	0,5	40,71	2,99		
Label : Y	Normal	ANormal	Normal	Anormal	...	Normal

Pearson correlation coefficient: **(**represented by the letter ρ), can be obtain by substituting covariances cov and variances σ based on a sample. So, for two genes g_1 and g_2 the formula for ρ is:

$$\rho = \frac{cov(g_1,g_2)}{\sqrt{\sigma^2(g_1)\sigma^2(g_2)}} = \frac{\sum_{i=1}^{n}(g_{1i}-\overline{g_1})(g_{2i}-\overline{g_2})}{\sqrt{\sum_{i=1}^{n}(g_{1i}-\overline{g_1})^2}\sqrt{\sum_{i=1}^{n}(g_{2i}-\overline{g_2})^2}} \quad (2)$$

where $\overline{g_1} = \frac{1}{n}\sum_{i=1}^{n} g_{1i}$ and $\overline{g_2} = \frac{1}{n}\sum_{i=1}^{n} g_{2i}$ are the mean for gene g_1 and g_2 respectively.

Mutual information (MI): It is a distance measure that compares genes whose profiles are discrete. It can be calculated using Shannon's entropy. It has been used to measure the dependency between two random variables based on the probability of them. For two genes g_1 and g_2, the mutual information between theme, $I(g_1, g_2)$, can be calculated as follow: [25], [26]:

$$\begin{aligned} I(g_1,g_2), &= H(g_1) - H(g_1|g_2) \\ &= H(g_2) - H(g_2|g_1) \\ &= H(g_1) + H(g_2) - H(g_1,g_2) \end{aligned} \quad (3)$$

where: $H(g_1), H(g_2)$ are the Shannon's entropies, expressed as follow:

$$H(g_1) = -\sum_{i=1}^{m} P(g_{1i}) \times \log_2\big(P(g_{1i})\big)$$

$H(g_1, g_2)$ is the joint entropy of the g_1 and g_2 defined as follow:

$$H(g_1,g_2) = -\sum_{j=1}^{m}\sum_{i=1}^{m} P\big(g_{1i}, g_{2j}\big) \times \log_2\Big(P\big(g_{1i}, g_{2j}\big)\Big)$$

$H(g_2|g_1)$ is the conditional entropy of g_1 given g_2 . It can be calculated as follow:

$$H(g_2|g_1) = -\sum_{j=1}^{m}\sum_{i=1}^{m} P\big(g_{1i}, g_{2j}\big) \times \log_2\big(P(g_{2i}|g_{1i})\big)$$

Noted that $P(g_{1i})$ represent the probability mass function, it can be calculated, when gene g_1 is discrete, as follow:

$$\mathrm{p}(g_{1i}) = \frac{\text{number of instants with value } g_{1i}}{\text{total number ofinstants (n)}}$$

and $P\big(g_{1i}, g_{2j}\big)$ is the joint probability mass function of the gene g_1 and g_2

2.2. Feature Selection

The goal of the feature selection is to select the smallest subset of features by scoring all features and using a threshold to remove features below the threshold. This process makes a classification problem simpler to interpret and reduces the time for training model. Mathematicly, for a feature set composed by all genes $G_f = \{g_{f1}, g_{f2}, \dots, g_{fn}\}$, the feature selection process identifies a subset of features S_f with dimension k where $k \leq n$, and $S_f \subseteq G_f$. In this study, five features selector algorithm were descussed, includes information gain, mRMR, linear correlation and chi-squared. The choice of filter method instead of a wrapper one due to the huge computational costs when uses wrappers methods [2].

Information gain (IG): It is a filter method that ranks features based on high information gain entropy in decreasing order. It ranks features based on the value of their mutual information with the class label using equation 3. Simplicity and low computational costs are the main advantages of this method. However, it does not take into consideration the dependency between the features; rather, it assumes independency, which is not always the case. Therefore some of the selected features may carry redundant information.

Chi-squared **(**Chi^2): is a statistical test to determine the dependency of two events, it characterize by it simplicity to implement (In feature selection, the two events are occurrence of the feature and occurrence of the class). The process consists of calculation of Chi^2 between every feature variable g_{fi} and the label Y. If Y is independent of g_{fi} , this feature variable will be discard. If they are dependent, this feature variable will be present into training model [27]. The initial hypothesis H_0 is the assumption that the two features are uncorelated, and it is tested by Chi^2 formula as follow:

$$Chi^2 = \sum_{i=1}^{r}\sum_{j=1}^{c}\frac{(g_{ij}-E_{ij})^2}{E_{ij}} \quad (4)$$

where g_{ij} is the observed frequency and E_{ij} is the expected frequency under the null hypothesis. E_{ij} can be computed by :

$$E_{ij} = \frac{row\ totla \times column\ total}{sample\ size}$$

The high value of Chi^2 indicates that the hypothesis of independence is incorrect and the feature is correlated with the class, thus it should be selected for model training. Linear correlation (Corr): **(**well-known similarity measure between two random variables) It can be calculated using Pearson correlation coefficient (ρ) as defined in equation 2. The resulting value is in $[-1; 1]$, with -1 meaning perfect negative correlation (as one variable increases, the other decreases), +1 meaning perfect positive correlation and 0 meaning no linear correlation between the two variables [28].

minimum Redundancy-Maximum Relevancy (mRMR): The mRMR filter method selects genes with the highest relevance and minimally redundant with the target class [8], [29]. mRMR of genes are based on mutual information using equation 3. The Maximum Relevance method selects the highest top k genes, which have the highest relevance correlated to the class labels from the descent arranged set of $I(g_i, Y)$, equation 5. Minimum Redundancy criterion is introduced by [14] in order to remove the redundancy features; this criterion defined by Equation 6.

$$max\left(\frac{1}{|G_f|}\sum_{g_i \in G_f} I(g_i; Y)\right) \quad (5)$$

$$min\left(\frac{1}{|G_f|^2}\sum_{g_i, g_j \in G_f} I(g_i; g_j)\right) \quad (6)$$

The (mRMR) filter takes the mutual information between each pair of genes into consideration and combines both optimization criteria of equation 5 and 6.

2.3. Classifiers

In this part, a brief description of commonly classifier algorithms used for classification task. Table 2 shows the parameters used for each classifier.

Table 2. Table Parameters of Classifier

Classifier	Parameter
K-means	K=2:9 Distance = Euclidean distance;
KNN	Distance = Euclidean distance; Number of nearest neighbors = 5 Kernel= rectangular
SOM	Number of input neurons = 10×10 Learning rate = 0.9 Radius = 20 Distance Metric = Euclidean Initialization = Random Number of iteration = 1000
Random Forest	Number of trees: 500 Number of variables tried at each split: 10

K-means: is a clustering algorithm or unsupervised classification which divides observations into k clusters [30]–[32]. It can be adapted for supervised classification case by dividing data into equal to or more than the number of classes. It takes a set S of m samples and the number of clusters K as input, and outputs a set $C = \{c_1, c_2, \dots, c_k\}$ of K centroids. The algorithm starts by initialising randomly all centroids; then, it iterates between two steps until a stopping criteria is done (often, the maximum number of iterations is reached). In the first one, each sample s_i is assigned to its nearest centroid c_k, based on the distance measure between s_i and c_k as follow:

$$\text{argmin}_{c_k \in C} D_{euc}(c_k, s_i)^2 \quad (7)$$

generating a set S_k^c formed by sample assignments for each k^{th} cluster centroid. In the second step, each centroid c_k is updated based on the mean of all samples assigned to their S_i^c as follow:

$$c_i = \frac{1}{|S_k^c|} \sum_{x_j \in S_k^c} s_j \quad (8)$$

Self-organizing maps (SOM): SOM is commonly used for visualizing and clustering of multidimensional data, due to his ability to project high-dimensional data in a lower dimension [33]-[37]. The SOM often consists of a regular grid of map units. Each unit is represented by a vector $W_j = (W_{j1}, W_{j2}, \dots, W_{jm})$, where m is input sample dimension. The units are connected to adjacent ones by neighbourhood relation. The SOM iteratively trained. At each training step, a sample input S is randomly chosen from the input data set, a metric distance is computed for all weight vectors W_j to find the reference vector W_{bmu} (called Best Matching Unit (BMU) that satisfies a minimum distance or maximum similarity criterion following the Equation 9.

$$bmu(t) = argmin_{1 \le i \le n} \|S(t) - W_i(t)\| \quad (9)$$

Where n is the neurons number in the map. The weights of the *bmu* and its neighbours are then adjusted towards the input pattern, following equation:

$$W_i(t+1) = W_i(t) + \beta_{bmu,i}(t) \|S - W_i\| \quad (10)$$

where $\beta_{bmu,i}$ is the neighbourhood function between the winner neuron *bmu* and neighbour neuron *i*. It is defined by the equation (11).

$$\beta_{bmu,i}(t) = \exp\left(\frac{\|r_{bmu} - r_i\|}{2\sigma_i^2(t)}\right) \quad (11)$$

Where $\|r_{bmu} - r_i\| \cong \left\|W^{bmu} - W^i\right\|$, r_{bmu} and $r_i$are positions of the BMU and neuron i on the Kohonen topological map. The σ(t) decreases monotonically with time.

K nearst neighbours (k-NN): is a non-parametric method used for classification [38]–[40]. The process begins by calculating similarity distance $D_{euc}(s_j, s_i)$ between test sample s_j and a set of training samples s_i and it sorts the distances in ascending (or descending) order. Then, it selects k closest neighbours to the sample s_j, and it gathers them together. To predict the class of this sample, it uses the majority voting: the class that occurs the most frequently in the nearest neighbors wins.

Random forest (RF): can be supposed of as a form of nearest neighbor predictor. It creates a set of decision trees from randomly selected subset of original training set; and sums the votes from different decision trees to decide the final class of the test object. It is considered well suited to situations characterized by a large number of features [41]-[43].

2.4. Datasets Descreption

In this study, the following published datasets was used (a brief description exists in Table 2). The first one is ALL/AML leukemia proposed by Golub et al in 1999 [3]; these data contains 7129 genes and 72 samples splits in two classes. It was used to classify patients with acute myeloid leukemia (labelled as AML) 25 examples (34.7%) and acute lymphoblastic leukemia (labelled as ALL) 47 examples (65.3%). The second dataset is Colon cancer dataset [44] that contains 62 samples. Among them, 40 tumor biopsies are from tumors (labeled as “N”) and 22 normal (labeled as "P") biopsies are from healthy parts of the colons of the same patients. The total number of genes to be tested is 2000. The third dataset is Lymphoma Cancer Data Classification [45]; it includes 45 tissues and 4026 genes. The first category, Germinal Centre B-Like (labelled as GCL) has 23 patients, and the second type Activated B-Like (labelled as ACL) has 22. The problem is to distinguish the GCL samples from the ACL samples. This data contains about 3.28% missing values.

Before applying any learning algorithm, the data must be pre-processed by several processes as missing values imputation, noisy data elimination, and normalizing data. Missing values: In general, dataset contains missing values occuring due to a variety of reasons including hybridization failures, artifacts on the microarray, insufficient resolution, image noise and corruption, or they may occur systematically as a result

of the spotting process. There are many techniques to handle these missing values such as omitting the entire record which contains the missing value or impute themes by Median, Mean, K-NN [46]. Data Normalization: Some algorithms, such as K-means and K-NN, may require that the data be normalized to increase the efficacy as well as efficiency of the algorithm. The normalization will prevent any variation in distance measures where the data may not been normalized. Normalizing the attribute will place all attribute within a similar range, usually [0, 1] [47]. Data Discretization: Discretization is the process of converting continuous variables into nominal ones. Studies have shown that discretization makes learning algorithms more accurate and faster [48]. The process can be done manually or by predefining thresholds on which to divide the data [49]–[51]. In this study, the percentage of missing values in our data set is less than 5%, which leads us to impute the missing values by the mean; and all data were normalized to zero. Then, gene expression values were directly used as input characteristics for classifiers. The framework of our process is described in Figure 1.

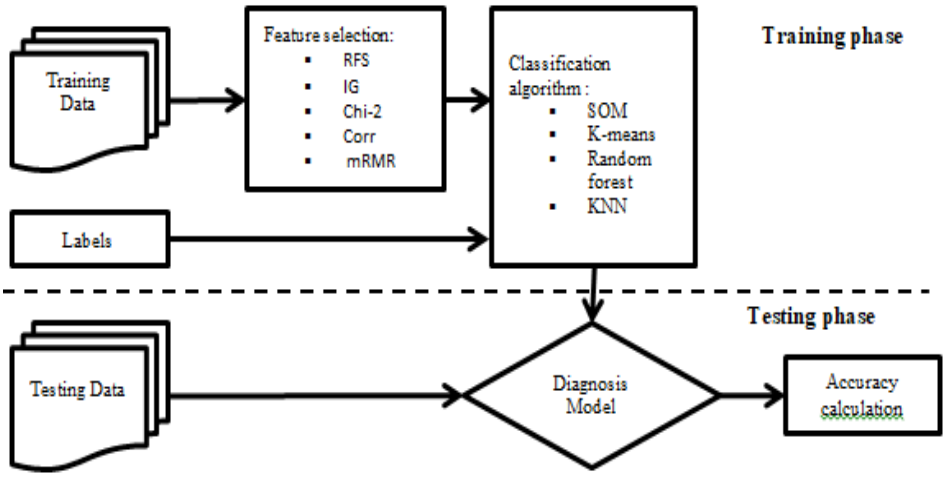

Figure 1. Framework used in this research

Before applying any learning algorithm, the data must be pre-processed by several processes as missing values imputation, noisy data elimination, and normalizing data. Missing values: In general, dataset contains missing values occuring due to a variety of reasons including hybridization failures, artifacts on the microarray, insufficient resolution, image noise and corruption, or they may occur systematically as a result of the spotting process. There are many techniques to handle these missing values such as omitting the entire record which contains the missing value or impute themes by Median, Mean, K-NN [46]. Data Normalization: Some algorithms, such as K-means and K-NN, may require that the data be normalized to increase the efficacy as well as efficiency of the algorithm. The normalization will prevent any variation in distance measures where the data may not been normalized. Normalizing the attribute will place all attribute within a similar range, usually [0, 1] [47]. Data Discretization: Discretization is the process of converting continuous variables into nominal ones. Studies have shown that discretization makes learning algorithms more accurate and faster [48]. The process can be done manually or by predefining thresholds on which to divide the data [49]-[51]. In this study, the percentage of missing values in our data set is less than 5%, which leads us to impute the missing values by the mean; and all data were normalized to zero. Then, gene expression values were directly used as input characteristics for classifiers. The framework of our process is described in Figure 1

3. RESULTS AND ANALYSIS

In this study, five features selector were tested on four different classifiers using three gene expression datasets labeled Leukeimia, Colon and Lymphoma short description in Table 3. Classification accuracies are presented before and after the feature selection in Table 4.The columns named ALL, RFS, IG, Chi-2, Corr, mRMR present the accuracy values of classification using all features, Random Forest Selector, Information Gain, Chi-square, linear Correlation and Minimum Redundancy Maximum Relevance filters.

Table 3. A Brief Summary of Datasets Used

Dataset	No. of examples	No. of features	No. of classes	
			Class 1	Class 2
Leukeimia	72	7129	47(ALL)	25(AML)
Colon	62	2000	40(P)	22 (N)
Lymphoma	47	4026	22(ACL)	23(GCL)

Table 4. Effects of Feature Selection on Classifiers Using 100 Important Features

Classifier	Dataset name	Classification Accuracy %					
		ALL	RFS	IG	Chi-2	Corr	mRMR
K-NN	Leukeimia	89.28	96.43	92.86	95.24	94.29	100
	Colon	78.01	87.22	86.82	87.77	88.88	85.63
	Lymphoma	93.33	100	100	100	100	100
K-means	Leukeimia	84.72	98.61	98.61	97.22	97.22	98.61
	Colon	79.03	87.09	88.70	88.70	88.70	90.32
	Lymphoma	82.22	93.33	97.7	100	100	100
SOM	Leukeimia	93.05	91.66	94.44	87.5	95.83	94.44
	Colon	88.70	98.38	96.77	93.54	96.77	95.16
	Lymphoma	87.68	88.88	95.55	93.33	95.55	97.77
Random Forest	Leukeimia	97.11	98.55	97.76	98.63	97.13	97.13
	Colon	83.39	88.40	86.50	88.73	86.82	85.06
	Lymphoma	90.74	98.33	95.16	93.05	93.16	100

The k-Nearest Neighbor (k-NN), Self-organizing maps (SOM), K-means and Random Forest were used as classifiers in the experiments, and the accuracy of five filters: Random Forest Selector, Information Gain, chi-square, linear correlation and Minimum Redundancy Maximum Relevance, when the top 100 features are selected are compared between them.

The choice of filters is due to the enormous size of the datasets used which increases the calculation time. For the k-NN classifier, we used the Euclidean distance as the distance metric, and the best k between 2 and 9; the same thing for K-means. For SOM, we used the parameters as follow: (10×10) input neurons, 0.9 Learning rate, Euclidean Distance Metric, all the neuron were initialized in random and 1000 as Number of iteration. For Random Forest, we used number of trees equal to 500 and the number of variables tried at each split is 10. The summarized description is in Table 2.

The result of this work in Table 4 and in Figures 2 ,3, 4 and 5 shows a very important effect of the selection of variables on the classification rate (the top 100 features in this experiment). From the table we can observe that mRMR and FRS are a little better on the Leukeimia dataset than other methods if used with K-NN and K-means, with a great improvement over the use of all variables in classification. For Lymphoma dataset, all the selectors work very well with all classifiers, with the exception of SOM which is suitable with mRMR, and FRS, and still, there is an improvement over the use of ALL features. For the colon dataset, the classification rate is always low in all cases, with an improvement when using SOM as classifiers and RFS as filter.

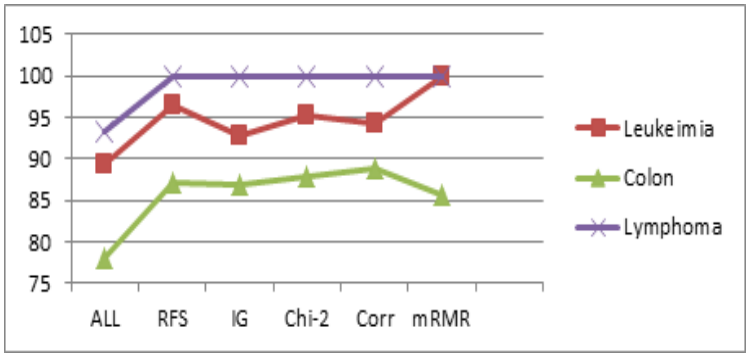

Figure 2. Effects of feature selection on KNN using 100 important features

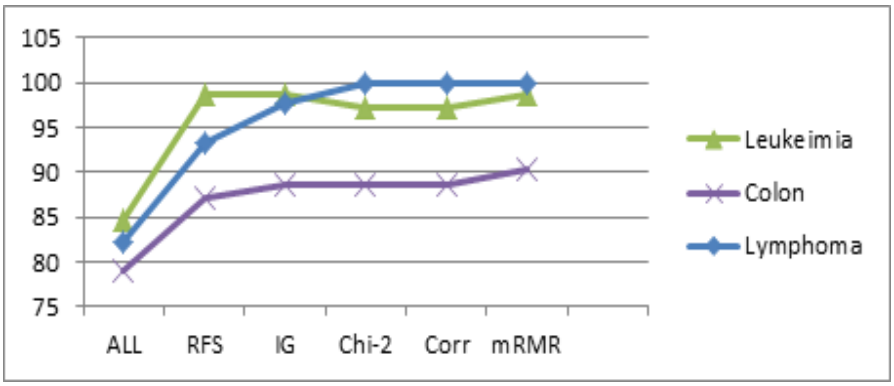

Figure 3. Effects of feature selection on K-means using 100 important features

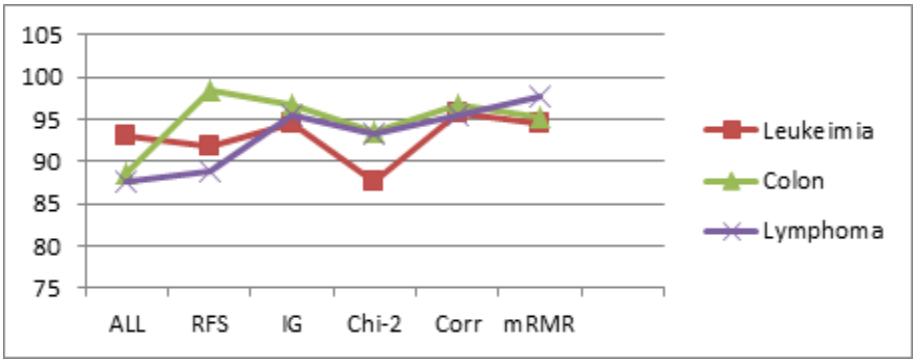

Figure 4. Effects of feature selection on SOM using 100 important features

4. CONCLUSION

Feature selection is an important issue in classification, because it may have a considerable effect on accuracy of the classifier. It reduces the number of dimensions of the dataset, so the processor and memory usage reduce; the data becomes more comprehensible and easier to study on. In this study we have investigated the influence of feature selection on four classifiers SOM, K-NN, K-means and Random Forest using five datasets. So by just using 100 top features, the classification accuracy is improved up to 9% comparing to all feature, and the complexity and the training time were reduced.

REFERENCES

[1] D. Devaraj, B. Yegnanarayana, and K. Ramar, "Radial basis function networks for fast contingency ranking," Int. J. Electr. Power Energy Syst., vol. 24, pp. 387–393, Jun. 2002.
[2] [S. Dudoit, J. Fridlyand, and T. P. Speed, "Comparison of Discrimination Methods for the Classification of Tumors Using Gene Expression Data," J. Am. Stat. Assoc., vol. 97, no. 457, pp. 77–87, Mar. 2002.
[3] T. R. Golub et al., "Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring," Science, vol. 286, no. 5439, pp. 531–537, Oct. 1999.
[4] L. Li, C. R. Weinberg, T. A. Darden, and L. G. Pedersen, "Gene selection for sample classification based on gene expression data: study of sensitivity to choice of parameters of the GA/KNN method," Bioinforma. Oxf. Engl., vol. 17, no. 12, pp. 1131–1142, Dec. 2001.
[5] A. Narayanan, E. C. Keedwell, J. Gamalielsson, and S. Tatineni, "Single-layer Artificial Neural Networks for Gene Expression Analysis," Neurocomput., vol. 61, no. C, pp. 217–240, Oct. 2004.
[6] A. Arauzo-Azofra, J. L. Aznarte, and J. M. Benítez, "Empirical Study of Feature Selection Methods Based on Individual Feature Evaluation for Classification Problems," Expert Syst Appl, vol. 38, no. 7, pp. 8170–8177, Jul. 2011.
[7] A. L. Blum and P. Langley, "Selection of relevant features and examples in machine learning," Artif. Intell., vol. 97, no. 1, pp. 245–271, Dec. 1997.
[8] C. Ding and H. Peng, "Minimum redundancy feature selection from microarray gene expression data," J. Bioinform. Comput. Biol., vol. 3, no. 2, pp. 185–205, Apr. 2005.
[9] I. Guyon and A. Elisseeff, "An Introduction to Variable and Feature Selection," J. Mach. Learn. Res., vol. 3, no. Mar, pp. 1157–1182, 2003.

[10] R. Ruiz, J. Riquelme, J. Aguilar-Ruiz, and M. Garcia Torres, "Fast feature selection aimed at high–dimensional data via hybrid–sequential–ranked searches," Expert Syst. Appl., vol. 39, pp. 11094–11102, Sep. 2012.
[11] A. Sharma, S. Imoto, and S. Miyano, "A top-r feature selection algorithm for microarray gene expression data," IEEE/ACM Trans. Comput. Biol. Bioinform., vol. 9, no. 3, pp. 754–764, Jun. 2012.
[12] Z. Wang, V. Palade, and Y. Xu, "Neuro-Fuzzy Ensemble Approach for Microarray Cancer Gene Expression Data Analysis," in 2006 International Symposium on Evolving Fuzzy Systems, 2006, pp. 241–246.
[13] K. Kira and L. A. Rendell, "A Practical Approach to Feature Selection," in Proceedings of the Ninth International Workshop on Machine Learning, San Francisco, CA, USA, 1992, pp. 249–256.
[14] R. Kohavi and G. H. John, "Wrappers for feature subset selection," Artif. Intell., vol. 97, no. 1, pp. 273–324, Dec. 1997.
[15] I. Gheyas and L. Smith, "Feature subset selection in large dimensionality domains," Pattern Recognit., vol. 43, pp. 5–13, Jan. 2010.
[16] E. B. Huerta, B. Duval, and J.-K. Hao, "Gene Selection for Microarray Data by a LDA-Based Genetic Algorithm," in Pattern Recognition in Bioinformatics, 2008, pp. 250–261.
[17] E. K. Tang, P. Suganthan, and X. Yao, "Gene selection algorithms for microarray data based on least squares support vector machine," BMC Bioinformatics, vol. 7, p. 95, Feb. 2006.
[18] M. Perez and T. Marwala, "Microarray data feature selection using hybrid genetic algorithm simulated annealing," in 2012 IEEE 27th Convention of Electrical and Electronics Engineers in Israel, 2012, pp. 1–5.
[19] J. Canul-Reich, L. O. Hall, D. B. Goldgof, J. N. Korecki, and S. Eschrich, "Iterative feature perturbation as a gene selector for microarray data," Int. J. Pattern Recognit. Artif. Intell., vol. 26, no. 05, p. 1260003, Aug. 2012.
[20] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, "Gene Selection for Cancer Classification using Support Vector Machines," Mach. Learn., vol. 46, no. 1–3, pp. 389–422, Jan. 2002.
[21] S. Maldonado, R. Weber, and J. Basak, "Simultaneous Feature Selection and Classification Using Kernel-penalized Support Vector Machines," Inf Sci, vol. 181, no. 1, pp. 115–128, Jan. 2011.
[22] P. A. Mundra and J. C. Rajapakse, "SVM-RFE With MRMR Filter for Gene Selection," IEEE Trans. NanoBioscience, vol. 9, no. 1, pp. 31–37, Mar. 2010.
[23] L.-Y. Chuang, C.-H. Yang, K.-C. Wu, and C.-H. Yang, "A Hybrid Feature Selection Method for DNA Microarray Data," Comput Biol Med, vol. 41, no. 4, pp. 228–237, Apr. 2011.
[24] C.-P. Lee and Y. Leu, "A novel hybrid feature selection method for microarray data analysis," Appl. Soft Comput., vol. 11, no. 1, pp. 208–213, Jan. 2011.
[25] M. Bennasar, Y. Hicks, and R. Setchi, "Feature selection using Joint Mutual Information Maximisation," Expert Syst. Appl., vol. 42, no. 22, pp. 8520–8532, Dec. 2015.
[26] C. E. Shannon, "A Mathematical Theory of Communication," SIGMOBILE Mob Comput Commun Rev, vol. 5, no. 1, pp. 3–55, Jan. 2001.
[27] H. Zhang et al., "Informative Gene Selection and Direct Classification of Tumor Based on Chi-Square Test of Pairwise Gene Interactions," BioMed Res. Int., vol. 2014, p. e589290, Jul. 2014.
[28] H. F. Eid, A. E. Hassanien, T. Kim, and S. Banerjee, "Linear Correlation-Based Feature Selection for Network Intrusion Detection Model," in Advances in Security of Information and Communication Networks, Springer, Berlin, Heidelberg, 2013, pp. 240–248.
[29] S. S. Shreem, S. Abdullah, M. Z. A. Nazri, and M. Alzaqebah, "Hybridizing relieff, mRMR filters and GA wrapper approaches for gene selection," J. Theor. Appl. Inf. Technol., vol. 46, no. 2, pp. 1034–1039, 2012.
[30] F.-X. Wu, W. J. Zhang, and A. J. Kusalik, "A Genetic K-means Clustering Algorithm Applied to Gene Expression Data," in Advances in Artificial Intelligence, 2003, pp. 520–526.
[31] K. R. Nirmal and K. V. V. Satyanarayana, "Issues of K Means Clustering While Migrating to Map Reduce Paradigm with Big Data: A Survey," Int. J. Electr. Comput. Eng. IJECE, vol. 6, no. 6, pp. 3047–3051, Dec. 2016.
[32] W. K. Oleiwi, "Using the Fuzzy Logic to Find Optimal Centers of Clusters of K-means," Int. J. Electr. Comput. Eng. IJECE, vol. 6, no. 6, pp. 3068–3072, Dec. 2016.
[33] C. Budayan, I. Dikmen, and M. T. Birgonul, "Comparing the performance of traditional cluster analysis, self-organizing maps and fuzzy C-means method for strategic grouping," Expert Syst. Appl., vol. 36, no. 9, pp. 11772–11781, Nov. 2009.
[34] I. Valova, G. Georgiev, N. Gueorguieva, and J. Olson, "Initialization Issues in Self-organizing Maps," Procedia Comput. Sci., vol. 20, pp. 52–57, Jan. 2013.
[35] M. Ettaouil and M. Lazaar, "Vector Quantization by Improved Kohonen Algorithm," J. Comput., vol. 4, no. 6, Jun. 2012.
[36] T. Kohonen, "Essentials of the self-organizing map," Neural Netw., vol. 37, pp. 52–65, Jan. 2013.
[37] S. Pavel and K. Olga, "Visual analysis of self-organizing maps," Nonlinear Anal Model Control, vol. 16, no. 4, pp. 488–504, Dec. 2011.
[38] T. Cover and P. Hart, "Nearest Neighbor Pattern Classification," IEEE Trans Inf Theor, vol. 13, no. 1, pp. 21–27, Sep. 2006.
[39] R. M. Parry et al., "k-Nearest neighbor models for microarray gene expression analysis and clinical outcome prediction," Pharmacogenomics J., vol. 10, no. 4, pp. 292–309, Aug. 2010.
[40] A. Alalousi, R. Razif, M. AbuAlhaj, M. Anbar, and S. Nizam, "A Preliminary Performance Evaluation of K-means, KNN and EM Unsupervised Machine Learning Methods for Network Flow Classification," Int. J. Electr. Comput. Eng. IJECE, vol. 6, no. 2, pp. 778–784, Apr. 2016.

[41] D. Amaratunga, J. Cabrera, and Y.-S. Lee, "Enriched random forests," Bioinformatics, vol. 24, no. 18, pp. 2010–2014, Sep. 2008.
[42] L. Breiman, "Random Forests," Mach. Learn., vol. 45, no. 1, pp. 5–32, Oct. 2001.
[43] X. Chen and H. Ishwaran, "Random Forests for Genomic Data Analysis," Genomics, vol. 99, no. 6, pp. 323–329, Jun. 2012.
[44] U. Alon et al., "Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays," Proc. Natl. Acad. Sci. U. S. A., vol. 96, no. 12, pp. 6745–6750, Jun. 1999.
[45] A. A. Alizadeh et al., "Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling," Nature, vol. 403, no. 6769, pp. 503–511, Feb. 2000.
[46] E. Acuna and C. Rodriguez, "The Treatment of Missing Values and its Effect on Classifier Accuracy," in Journal of Classification, 2004, pp. 639–647.
[47] Y. K. Jain and S. K. Bhandare, "Min Max Normalization based data Perturbation Method for Privacy Protection," International Journal of Computer & Communication Technology (IJCCT), vol. 2, no. 8, pp. 45–50, 2011.
[48] J. Dougherty, R. Kohavi, and M. Sahami, "Supervised and Unsupervised Discretization of Continuous Features," in Machine Learning: Proceedings of the Twelfth International Conference, 1995, pp. 194–202.
[49] H. Liu, F. Hussain, C. L. Tan, and M. Dash, "Discretization: An Enabling Technique," Data Min. Knowl. Discov., vol. 6, no. 4, pp. 393–423, Oct. 2002.
[50] P. E. Meyer, F. Lafitte, and G. Bontempi, "minet: A R/Bioconductor package for inferring large transcriptional networks using mutual information," BMC Bioinformatics, vol. 9, p. 461, Oct. 2008.
[51] Y. Yang and G. I. Webb, "On Why Discretization Works for Naive-Bayes Classifiers," in AI 2003: Advances in Artificial Intelligence, 2003, pp. 440–452.