Biointerface Research in Applied Chemistry
Platinum Open Access Journal (ISSN: 2069-5837)

Article
Volume 12, Issue 2, 2022, 2422 - 2439
https://doi.org/10.33263/BRIAC122.24222439

A Robust Procedure for Machine Learning Algorithms Using Gene Expression Data

Md. Rabiul Auwul [1], Chongqi Zhang [1,*], Md. Shahjaman [2,*]

[1] School of Economics and Statistics, Guangzhou University, Guangzhou 510006, China; rabiulauwul@gmail.com (M.R.A.); cqzhang@gzhu.edu.cn (C.Z.);
[2] Department of Statistics, Begum Rokeya University, Rangpur-5400, Bangladesh; shahjaman_brur@yahoo.com (M.S.)
* Correspondence: cqzhang@gzhu.edu.cn (C.Z.) and shahjaman_brur@yahoo.com (M.S.)
Scopus Author ID 35212374200 (C.Z.)
57193488285 (M.S.)
Received: 6.04.2021; Revised: 15.05.2021; Accepted: 19.05.2021; Published: 18.06.2021

Abstract: Cancer classification is one of the main objectives for analyzing big biological datasets. Machine learning algorithms (MLAs) have been extensively used to accomplish this task. Several popular MLAs are available in the literature to classify new samples into normal or cancer populations. Nevertheless, most of them often yield lower accuracies in the presence of outliers, which leads to incorrect classification of samples. Hence, in this study, we present a robust approach for the efficient and precise classification of samples using noisy GEDs. We examine the performance of the proposed procedure in a comparison of the five popular traditional MLAs (SVM, LDA, KNN, Naïve Bayes, Random forest) using both simulated and real gene expression data analysis. We also considered several rates of outliers (10%, 20%, and 50%). The results obtained from simulated data confirm that the traditional MLAs produce better results through our proposed procedure in the presence of outliers using the proposed modified datasets. The further transcriptome analysis found the significant involvement of these extra features in cancer diseases. The results indicated the performance improvement of the traditional MLAs with our proposed procedure. Hence, we propose to apply the proposed procedure instead of the traditional procedure for cancer classification.

Keywords: Gene expression data; Classification; Outlier detection and modification; DE gene; MAD and Robustness.

1. Introduction

Big data is becoming the main issue in today's world for its diverse, unstructured and fast-changing behavior. Gene expression datasets (GEDs) are high-dimensional and big datasets. So, analyzing using these datasets becomes complicated day by day [1,2]. For example, a single gene expression dataset consists of thousands of genes/features relative to a small number of sample sizes. Many of the features are correlated [3]. Therefore, the curse of dimensionality problems hampers the downstream analysis using GEDs. One of the main objectives of GEDs is the classification of new samples into one of two or more populations (e.g., normal or cancer) based on training datasets whose category membership is known in advance. Many supervised machine learning algorithms have been developed to extract useful information about the underlying mechanism of gene functions and pathways [4–7]. Classification is a two-step process, first step is the training phase, where the classification model is constructed by using a training set. The second step is the classification phase, where

the model is used to predict class labels and test the built model on test data [8]. MLAs Fisher's linear discriminant analysis (LDA) [9] is the oldest and popular among them. Nearest Neighbor like K-Nearest Neighbor [10] used difference metric to find the nearest neighbor for classifying unlabeled data. It has been used extensively in GED analysis. It explores the k closest features in the training set and allots to the class that seems most regularly. A regression-based classification, namely Multi-nomial Logistic Regression Model [11], using dichotomous dependent variables to predict. The bayesian-based algorithm, namely Naïve Bayes Classifier [12], used Bayes theorem to classify unlabeled data into classes. A support vector machine (SVM) algorithm has been extensively used to analyses data and recognizes the patterns for classification and regression analysis [13]. It is done by creating hyperplanes in a high or infinite-dimensional space, which can then be used for classification, regression, or other tasks. Random forest [14] based decision tree also has been applied in GED analysis and accepted by the research communities its robustness performance for large-sample case.

Notwithstanding the classification algorithms for labeling the class, most of these algorithms are sensitive to outliers and frequently produce misrepresentative results in the presence of outlying observation. Outliers may originate in microarray datasets because there are a number of steps involved in data generating procedures, from hybridization to image analysis [15]. Since the GED sets contain many genes, the researcher used some feature selection techniques and classification methods [16–25]. These feature selection techniques are used to simplify the computational process of grouping disease samples from normal samples by reducing the time and cost and increasing classifiers' accuracy.

The appliance of robust estimating purposes for data analysis and the appliance of classical estimating purpose with the reduced/modified datasets are generally the two types of statistical procedures to overwhelm the outlying difficulties in the GE data analysis [26]. Several research studies have existed to robustify the classifiers by excluding/reducing the outliers from the main dataset [27–33]. In this process, the significant genes might be detached before the analysis or sample size turns smaller, generating computational intricacy by the statistical algorithms. In this case, the process of modification of the dataset is better to analyze than the datasets after reducing/excluding outliers, in this case. Since any gene or sample need not be detached from the dataset. It interchanges the outlying expressions by the feasible values. Hence, in this paper, we propose an outlier modification rule to progress the performance of machine learning algorithms (MLAs). We considered five widespread machine learning algorithms (SVM, LDA, KNN, Naïve Bayes and Random forest) to explore the performance of the proposed procedure.

The remaining part of this paper is prepared as follows: Section 2 briefly describes the formulation of five classification algorithms and the proposed algorithm. After Section 2, a comprehensive simulation study and two real data studies have been carried out with results and discussion.

2. Materials and Methods

2.1. Machine Learning Algorithms to be compared.

In this study, five popular machine learning algorithms (MLAs) are evaluated, namely, Support Vector Machine (SVM), Linear Discriminant Analysis (LDA), K- Nearest Neighbor (KNN), Naïve Bayes (NB) and Random Forest (RF).

2.1.1. Support Vector Machine (SVM).

SVM is an essential classification algorithm used to search the hyperplane in an N-dimensional space explicitly classifies the data points (N- the number of features). The main objective of SVM is to search this hyperplane that has the maximum margin [34]. For maximizing the margin between the hyperplane and support vectors (data points that are nearer to the hyperplane), "Hinge's loss/cost" function has been used. Suppose X is the features and y is the target value that needs to predict. SVM is to predict y class to the actual y.

Predict y = function (weighted values of X)

The Hinges's loss function added with regularization term as:

$$\text{Total cost} = \frac{\|\omega\|^2}{2} + C * (\text{sum of all losses for each observation});$$

Here, C is the hyper-parameter that reins the amount of regularization. If C is chosen sufficiently small, then we call this hard-margin classifier and if C is chosen sufficiently large, then we call this soft-margin classifier. ω is defined as weight values.

2.1.2. Linear Discriminant Analysis (LDA).

The LDA is a classification algorithm first developed by R.A. Fisher in 1936. LDA is based upon the concept of searching for a linear combination of predictor variables that best separates the target variables in classes [9]. LDA is a general discriminant function with a linear decision boundary. For example, the target dataset y can be classified simply by solving the discriminant function d_i for each class C_i with the rule of classification R_c. Let, the prior probabilities is $p(C_i)$, mean of each class is μ_i and the common covariance matrix is S_w. Then the discriminant function is:

$$d_i(y) = \log(p(C_i)) - \frac{1}{2}\mu_i^T S_w^{-1}\mu_i + y^T S_w^{-1}\mu_i$$

The classification rule for the target dataset given as:

$$R_c(y) = i^{**} :\Leftrightarrow i^{**} = \underbrace{\arg\max}_{i} d_i(y)$$

2.1.3. K-Nearest Neighbors (KNN).

A non-parametric methodology used to discover a group of *k* samples nearest to unknown samples is known as KNN classifier [35], used for supervised learning. For determining the nearest sample, it uses distance metric (for example- Euclidian Distance). The main function of KNN is to determine the class (label) of unknown samples from those k samples by calculating the average of the response variables.

Suppose, X_j be the values of the features and K_j denotes labels of X_j for each j. Let the number of a class is k and x be the points for which label is unknown. The steps of finding a class for unknown labels by KNN are:

Step 1: Determine $d(x, X_j), \mathrm{j} = 1, 2, ..., \mathrm{n}$ for all values of k (d is a distance metric)

Step 2: For n determined distances, arrange the values in increasing order and take D distances from the sorted list ($D \geq 0$).

Step 3: Find the D points corresponding to these D distances.

Step 4: let D_j denotes the number of points belonging to the j^{th} class among the D points.

Step 5: If $D_j > D_i, \forall j \neq i$, then put x in class i.

2.1.4. Naïve Bayes.

The probabilistic machine learning model used to solve classification problems based on *Bayes Theorem* where the features are independent is known as Naïve Bayes Classifier [12].

Let, y is a class variable that needs to predict and x_1, $x_2,...x_n$ are the features; then, according to Bayes Theorem, the probability of getting classes for y based on x's is:

$$P(y \mid x_1, x_2, ..., x_n) = \frac{P(x_1 \mid y)P(x_2 \mid y).....P(x_n \mid y)P(y)}{P(x_1)P(x_2).....P(x_n)}$$

Since the features are independent and the denominator is unchangeable, then by removing the denominator and get result proportionally,

$$P(y|x_1, x_2, ..., x_n) \propto P(y)\prod_{i=1}^{n} P(x_i|y)$$

So, we can obtain the class based on features by finding the maximum probability as:

$$\underbrace{\arg\max}_{K} P(y_K)\prod_{i=1}^{n} P(x_i|y)$$

Where, K is the finite classes of y.

2.1.5. Random Forest.

An ensemble learning proposed by Breiman [36] constructing several decision trees based on a random averaging selection of independent variables of the training set is known as a classification algorithm. For the classification problem, the variables are ranked through their importance. Let the dataset $S_n = \{(x_i, y_i)\}$, $i = 1, 2, ..., n$, we fit a random forest to that data to quantify the importance of a variable. Throughout the fitting procedure, the error for each data point is intended and averaged over the forest. To quantity, the importance of the i^{th} feature after training, the values of the i^{th} feature are permuted among the training data and the error is again calculated on this data set. The importance score for the i^{th} feature is calculated by averaging the difference in error before and after the permutation for all the trees. Standardization of the score is done by the standard deviation of these differences and classifies each group according to the training group importance score. Features which yield large values for this score are more important than features that produce small values. Random forests deliver evidence about the importance of a variable and the closeness of the data points.

2.2. Proposed method.

The simple, robust estimate of location is a median and the simple, robust estimate of scale is a median absolute deviation (MAD). For that reason, an outlier detection and modification technique using median and MAD introduce in this paper. Let us suppose that $\boldsymbol{x}_{ijk}$ stand the i^{th} gene expression for the j^{th} replicates in k^{th} class then median and MAD is defined by $\text{med}_{i,k} = median(\boldsymbol{x}_{ijk}; i = 1, 2,\ldots, G; j = 1,2,\ldots, n_k; k=1,2)$ and $\text{MAD}_{i,k} = \text{med}_j(|\boldsymbol{x}_{ijk} - \text{med}_{i,k}|)$, respectively. The proposed robust technique is as follows:

i. We proclaim an outlying observation within the expressions of a gene-based on training dataset if it does not fall within a certain interval, i.e., $Pr\,(\text{med}i,k - 3\times1.4828\times\text{MAD}_{i,k} \leq \boldsymbol{x}_{ijk} \leq \text{med}i,k + 3\times1.4828\times\text{MAD}_{i,k})=0$. Otherwise, we declare that gene as a non-outlying gene.
ii. If outliers occur, then modify the outlying observation by the median of this gene and obtain modified training gene expression (MTGE) data.
iii. Apply t-test to select the most informative features based on MTGE data and rank these features according to their adjusted p-values.
iv. Select top $k<\max(n_1,n_2)$ features out of G genes using the adjusted p-values and apply them to train the popular classifiers.
v. Calculate different indices of the confusion matrices (accuracy, sensitivity, specificity, FPR, PPV, NPV and detection rate) to investigate the performance of the classifiers using top k-features of the modified training dataset.

The R-codes of the proposed algorithm have been implemented in the R package MLOutMod, which can be found in https://github.com/snotjanu/MLOutMod. The workflow of this proposed procedure has been visualized in Figure 1.

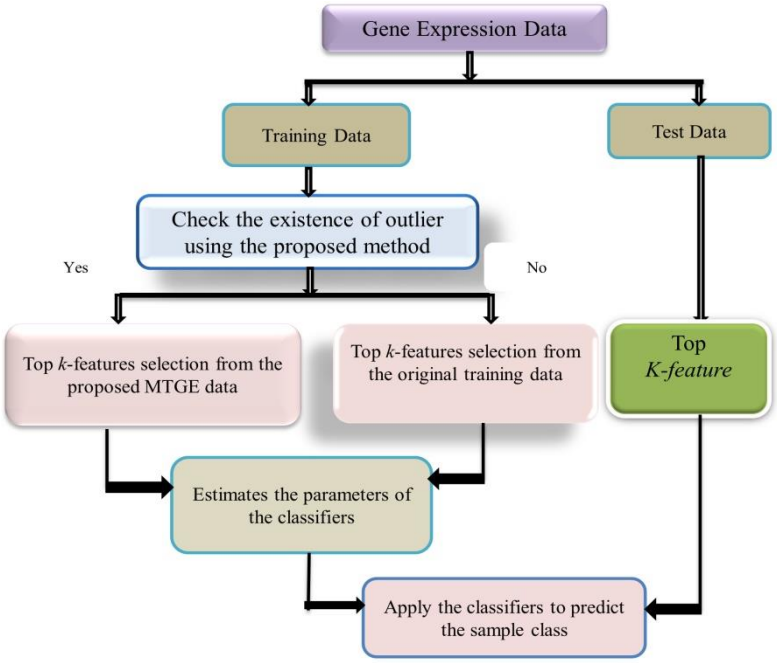

Figure 1. Schematic flowchart of the proposed procedure.

2.3. Performance measure.

In order to measure the performance of different classification algorithms for binary classification tests such as DE genes or EE genes, we use receiving operating characteristics

(ROC) curve, area under the ROC curve (AUC) and all the measures related to this curve. We compute the following measures of performance:

False positive rate (FPR) = $\frac{FP}{FP+TN}$, Accuracy = $\frac{TP+TN}{TP+FP+TN+FN}$

Sensitivity = $\frac{TP}{TP+FN}$, Specificity = $\frac{TN}{TN+FP}$, Positive predicted value (PPV) = $\frac{TP}{TP+FP}$,

Negative predicted value (NPV) = $\frac{TN}{TN+FN}$ and Detection Rate = $\frac{TP}{TP+FP+TN+FN}$

Where, TP, TN, FP and FN denote the number of true positives, number of true negatives, number of false-positive and number of false negatives, respectively. PPV, NPV are positive predicted values and negative predicted values, respectively. Based on these parameters, we announce an algorithm as a good performer if it produces greater values of accuracy, sensitivity, specificity and lesser values of FPR and NPV.

3. Results and Discussion

We examine the performance of the proposed procedure in comparison with the classical procedure through five widely used classification algorithms (SVM, LDA, KNN, Naïve Bayes and RF) using one simulated and two real microarray gene expression datasets, namely and head- -neck cancer [37] and colon cancer [38]. We used R packages *class* for LDA, *knn* for KNN, *caret* for Random Forest, *e1071* for SVM and naïve Bayes. The R package *MASS* was used to evaluate the performance of these algorithms. The comprehensive R archive network (cran) or Bioconductor are the main sources of these packages.

Table 1. Simulated data-generating model.

Group Name	Normal sample (n_1)	Cancer Sample (n_2)
p_1	$N(\mu, \sigma^2)$	$N(+\mu, \sigma^2)$
p_2	$N(-\mu, \sigma^2)$	$N(+\mu, \sigma^2)$
p_3	$N(0, \sigma^2)$	$N(0, \sigma^2)$

3.1. Simulated data analysis.

The simulated data were generated with known characteristics for two (k=2) groups with and without outliers that imitative the behavior of real gene expression data. The model used to generate this simulated data is defined in Table 1, where the row and column represent the genes and sample groups (normal and cancer), respectively. For randomizing, the datasets are contaminated with Gaussian noise. Two categories of datasets were generated from this model: one comprises of G=1,000 genes with small sample size, 10 in each group (n_1=n_2=10), and others comprise G=1,000 genes with a large sample size, 50 in each group (n_1=n_2=50). The gene expression profiles of 1,000 genes with n= (n_1+n_2) samples represent both datasets. We generated 100 differentially expressed (DE) genes and 900 equally expressed (EE) genes (pattern 3) from G=1,000 genes that were represented in each dataset. These 100 DE genes are then divided into P_1 =50, up-regulated (pattern 1) and P_2=50, down-regulated (pattern 2) DE genes of two groups. To generate both of these datasets, we fixed the value of the Gaussian noise parameter, σ^2= 0.05 and the parameter μ as 0.2.

Table 2. Classification performance of five classifiers based on original and modified training dataset for a small sample case.

Algorithms	% of Outliers	Accuracy	95% CI of Accuracy	Sensitivity	Specificity	PPV	NPV	Detection Rate
SVM	No Outlier	(0.996) [0.995]	(0.686,0.999) [0.684,0.999]	(0.996) [0.996]	(0.996) [0.996]	(0.997) [0.996]	(0.997) [0.996]	(0.996) [0.995]
	10% outliers	(0.834) [0.994]	(0.494,0.970) [0.683,0.999]	(0.832) [0.992]	(0.836) [0.996]	(0.867) [0.979]	(0.862) [0.993]	(0.832) [0.992]
	20% outliers	(0.673) [0.995]	(0.336,0.905) [0.685,0.999]	(0.664) [0.996]	(0.682) [0.994]	(0.705) [0.995]	(0.680) [0.997]	(0.664) [0.996]
	50% outliers	(0.509) [0.993]	(0.205,0.807) [0.682,0.999]	(0.488) [0.990]	(0.530) [0.996]	(0.494) [0.997]	(0.507) [0.992]	(0.488) [0.990]
LDA	No Outlier	(0.934) [0.934]	(0.609,0.991) [0.607,0.999]	(0.928) [0.926]	(0.940) [0.938]	(0.945) [0.943]	(0.939) [0.937]	(0.928) [0.927]
	10% outliers	(0.835) [0.935]	(0.498,0.966) [0.622,0.994]	(0.826) [0.642]	(0.844) [0.998]	(0.857) [0.955]	(0.846) [0.948]	(0.826) [0.942]
	20% outliers	(0.698) [0.946]	(0.366,0.910) [0.625,0.992]	(0.706) [0.934]	(0.690) [0.958]	(0.723) [0.961]	(0.713) [0.945]	(0.923) [0.999]
	50% outliers	(0.511) [0.926]	(0.205,0.810) [0.602,0.988]	(0.496) [0.904]	(0.526) [0.948]	(0.419) [0.951]	(0.503) [0.922]	(0.496) [0.904]
KNN	No Outlier	(0.995) [0.995]	(0.685,0.999) [0.685,0.999]	(0.992) [0.992]	(0.998) [0.997]	(0.998) [0.997]	(0.993) [0.992]	(0.992) [0.992]
	10% outliers	(0.838) [0.995]	(0.496,0.973) [0.685,0.999]	(0.838) [0.992]	(0.828) [0.998]	(0.866) [0.998]	(0.866) [0.993]	(0.838) [0.992]
	20% outliers	(0.706) [0.993]	(0.366,0.922) [0.682,0.999]	(0.712) [0.996]	(0.700) [0.990]	(0.729) [0.992]	(0.726) [0.997]	(0.712) [0.996]
	50% outliers	(0.502) [0.986]	(0.201,0.802) [0.672,0.999]	(0.474) [0.986]	(0.530) [0.996]	(0.485) [0.988]	(0.524) [0.988]	(0.474) [0.987]
NB	No Outlier	(0.997) [0.995]	(0.688,0.999) [0.684,0.999]	(0.998) [0.996]	(0.996) [0.994]	(0.996) [0.994]	(0.998) [0.996]	(0.998) [0.998]
	10% outliers	(0.932) [0.987]	(0.688,0.990) [0.688,0.999]	(0.936) [0.978]	(0.928) [0.996]	(0.945) [0.996]	(0.952) [0.983]	(0.936) [0.978]
	20% outliers	(0.817) [0.981]	(0.677,0.961) [0.677,0.999]	(0.858) [0.976]	(0.776) [0.986]	(0.842) [0.989]	(0.870) [0.982]	(0.858) [0.976]
	50% outliers	(0.498) [0.980]	(0.390,0.802) [0.690,0.997]	(0.408) [0.968]	(0.488) [0.992]	(0.510) [0.993]	(0.492) [0.978]	(0.508) [0.968]
RF	No Outlier	(0.999) [0.999]	(0.692,0.999) [0.929,0.999]	(0.999) [0.999]	(0.999) [0.999]	(0.999) [0.999]	(0.999) [0.999]	(0.999) [0.999]
	10% outliers	(0.950) [0.999]	(0.626,0.997) [0.692,0.999]	(0.646) [0.999]	(0.954) [0.999]	(0.961) [0.999]	(0.956) [0.999]	(0.936) [0.999]
	20% outliers	(0.817) [0.998]	(0.465,0.961) [0.692,0.999]	(0.822) [0.999]	(0.792) [0.999]	(0.810) [0.999]	(0.839) [0.999]	(0.822) [0.999]
	50% outliers	(0.513) [0.999]	(0.208,0.810) [0.692,0.999]	(0.488) [0.999]	(0.538) [0.999]	(0.513) [0.999]	(0.517) [0.999]	(0.488) [0.999]

[1]The brackets types () and [ ] indicate the result obtained from the original training data and proposed modified training data, respectively.

Table 3. Classification performance of five classifiers based on original and modified training dataset for a large-sample case.

Algorithms	% of Outliers	Accuracy	95% CI of Accuracy	Sensitivity	Specificity	PPV	NPV	Detection Rate
SVM	No Outlier	(0.999) [0.999]	(0.929,0.999) [0.929,0.999]	(0.999) [0.999]	(0.999) [0.999]	(0.999) [0.999]	(0.999) [0.999]	(0.999) [0.999]
	10% outliers	(0.986) [0.999]	(0.905,0.998) [0.929,0.999]	(0.979) [0.999]	(0.992) [0.999]	(0.992) [0.999]	(0.980) [0.999]	(0.979) [0.999]
	20% outliers	(0.821) [0.999]	(0.691,0.910) [0.929,0.999]	(0.761) [0.999]	(0.880) [0.999]	(0.889) [0.999]	(0.813) [0.999]	(0.761) [0.999]
	50% outliers	(0.495) [0.999]	(0.351,0.640) [0.929,0.999]	(0.341) [0.999]	(0.649) [0.999]	(0.480) [0.999]	(0.474) [0.999]	(0.341) [0.999]
LDA	No Outlier	(0.999) [0.999]	(0.929,0.999) [0.929,0.999]	(0.999) [0.999]	(0.999) [0.999]	(0.999) [0.999]	(0.999) [0.999]	(0.999) [0.999]
	10% outliers	(0.995) [0.999]	(0.921,0.998) [0.926,0.998]	(0.996) [0.999]	(0.995) [0.998]	(0.996) [0.999]	(0.996) [0.999]	(0.996) [0.999]
	20% outliers	(0.934) [0.999]	(0.833,0.977) [0.929,0.998]	(0.923) [0.999]	(0.946) [0.998]	(0.950) [0.999]	(0.934) [0.999]	(0.923) [0.999]
	50% outliers	(0.495) [0.999]	(0.352,0.640) [0.929,0.998]	(0.525) [0.999]	(0.466) [0.998]	(0.487) [0.999]	(0.503) [0.999]	(0.525) [0.999]

Algorithms	% of Outliers	Accuracy	95% CI of Accuracy	Sensitivity	Specificity	PPV	NPV	Detection Rate
KNN	No Outlier	(0.999) [0.999]	(0.929,0.999) [0.929,0.999]	(0.999) [0.999]	(0.999) [0.999]	(0.999) [0.999]	(0.999) [0.999]	(0.999) [0.999]
	10% outliers	(0.835) [0.999]	(0.709,0.917) [0.929,0.999]	(0.741) [0.999]	(0.929) [0.999]	(0.936) [0.999]	(0.817) [0.999]	(0.741) [0.999]
	20% outliers	(0.610) [0.999]	(0.465,0.742) [0.929,0.999]	(0.356) [0.999]	(0.864) [0.999]	(0.815) [0.999]	(0.615) [0.999]	(0.356) [0.999]
	50% outliers	(0.495) [0.999]	(0.351,0.640) [0.929,0.999]	(0.264) [0.999]	(0.726) [0.999]	(0.472) [0.999]	(0.492) [0.999]	(0.264) [0.999]
NB	No Outlier	(0.999) [0.999]	(0.929,0.999) [0.929,0.999]	(0.999) [0.999]	(0.999) [0.999]	(0.999) [0.999]	(0.999) [0.999]	(0.999) [0.999]
	10% outliers	(0.977) [0.999]	(0.929,0.990) [0.929,0.999]	(0.968) [0.999]	(0.986) [0.999]	(0.989) [0.999]	(0.976) [0.999]	(0.968) [0.999]
	20% outliers	(0.854) [0.999]	(0.629,0.910) [0.929,0.999]	(0.835) [0.999]	(0.874) [0.999]	(0.924) [0.999]	(0.897) [0.999]	(0.835) [0.999]
	50% outliers	(0.612) [0.999]	(0.429,0.737) [0.929,0.999]	(0.556) [0.999]	(0.668) [0.999]	(0.757) [0.999]	(0.685) [0.999]	(0.556) [0.999]
RF	No Outlier	(0.999) [0.999]	(0.929,0.999) [0.929,0.999]	(0.999) [0.999]	(0.999) [0.999]	(0.999) [0.999]	(0.999) [0.999]	(0.999) [0.999]
	10% outliers	(0.999) [0.999]	(0.929,0.999) [0.929,0.999]	(0.999) [0.999]	(0.999) [0.999]	(0.999) [0.999]	(0.999) [0.999]	(0.999) [0.999]
	20% outliers	(0.999) [0.999]	(0.929,0.999) [0.929,0.999]	(0.999) [0.999]	(0.999) [0.999]	(0.999) [0.999]	(0.999) [0.999]	(0.999) [0.999]
	50% outliers	(0.999) [0.999]	(0.927,0.999) [0.929,0.999]	(0.999) [0.999]	(0.998) [0.999]	(0.998) [0.999]	(0.999) [0.999]	(0.999) [0.999]

[1]The brackets types () and [ ] indicate the result obtained from the original training data and proposed modified training data, respectively.

The performance of the proposed procedure compared with the classical procedure for sample classification as normal or cancer groups, we employed five popular classifiers such as SVM, LDA, KNN, naive Bayes and random forest. We generated 100 simulated datasets from Table 1 for each of small (n=20, n_1=10, n_2=10) and large (n=100, n_1=50, n_2=50) sample cases, respectively. To demonstrate the classification performance of these methods, each of the 100 simulated datasets was randomly divided into two independent datasets for constructing the training and test dataset, where these training and test datasets comprised of a same number of samples. The performance of these algorithms is also estimated with outlier observations. We multiply a constant, z term with the maximum value of the gene expressions within the groups to generate outlying datasets using $x_{ijk}^{*} = v + z^{*}(x_{ijk}; k = 1,2; i = 1,2,...,G; j = 1,2,...n_k)$. Here, x_{ijk} it symbolizes the i^{th} gene expression of j^{th} samples in k^{th} a group, $v \in (5, 10)$ and $z \in (2, 4)$. We considered different outlying percentages of genes (10%, 20% and 50%) with one or two randomly selected samples.

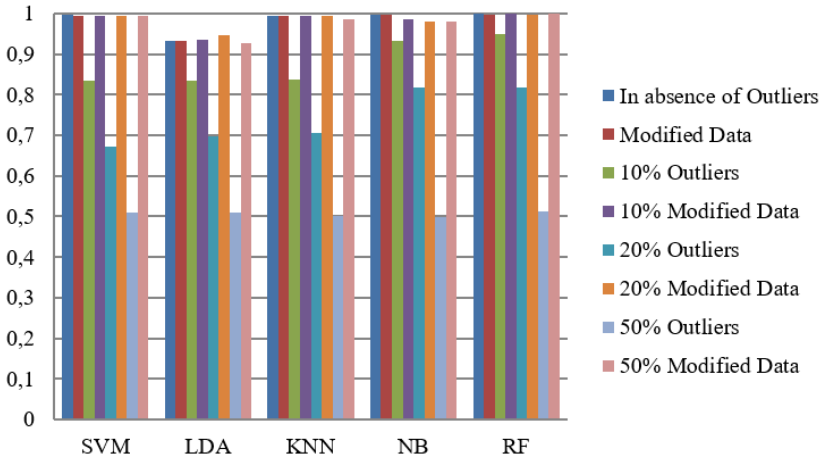

Figure 2. Performance evaluation using the average value of accuracy for small-sample case

First, we apply the proposed outlier modification rule in the 100 training datasets to reconstruct 100 modified training gene expression (MTGE) datasets. We computed average values of different performance measurements of these five classifiers: sensitivity, specificity, positive predictive value (PPV), negative predictive value (NPV), accuracy, and detection rate based on the estimated 100 DE genes. Table 2 and Table 3 summarized these performance measures by averaging these 100 values for small and large sample cases, respectively. We noticed in Table 2 that for small-sample cases, all five classifiers (SVM, LDA, KNN, NB and RF) produce the same results using original data and proposed modified training datasets with the absence of outlier. The values within the brackets ( ) and [ ] indicate the results obtained from the original training data and proposed modified training data, respectively. However, in the presence of 10%, 20% and 50% outliers, in this case, the five classifiers produce much better results using the proposed modified training dataset than the original training dataset. For example, the average accuracies 0.994, 0.935, 0.995, 0.987 and 0.999 produced by SVM, LDA, KNN, NB and RF, respectively, with the presence of one outlier in each of 10% genes that are larger than 0.834, 0.835, 0.838, 0.932 and 0.950, those were produced by the aforesaid classifiers with the same condition using proposed modified training dataset (see the second row of each classifier in Table 2). Contrariwise, for large sample cases (see Table 3) with the presence of outliers, the five classifiers' performance deteriorated with the original training dataset except for RF. This is because for large sample cases, the random forest (RF) is robust with outliers. Notwithstanding, the performance of all the classifiers improved while using the modified training dataset.

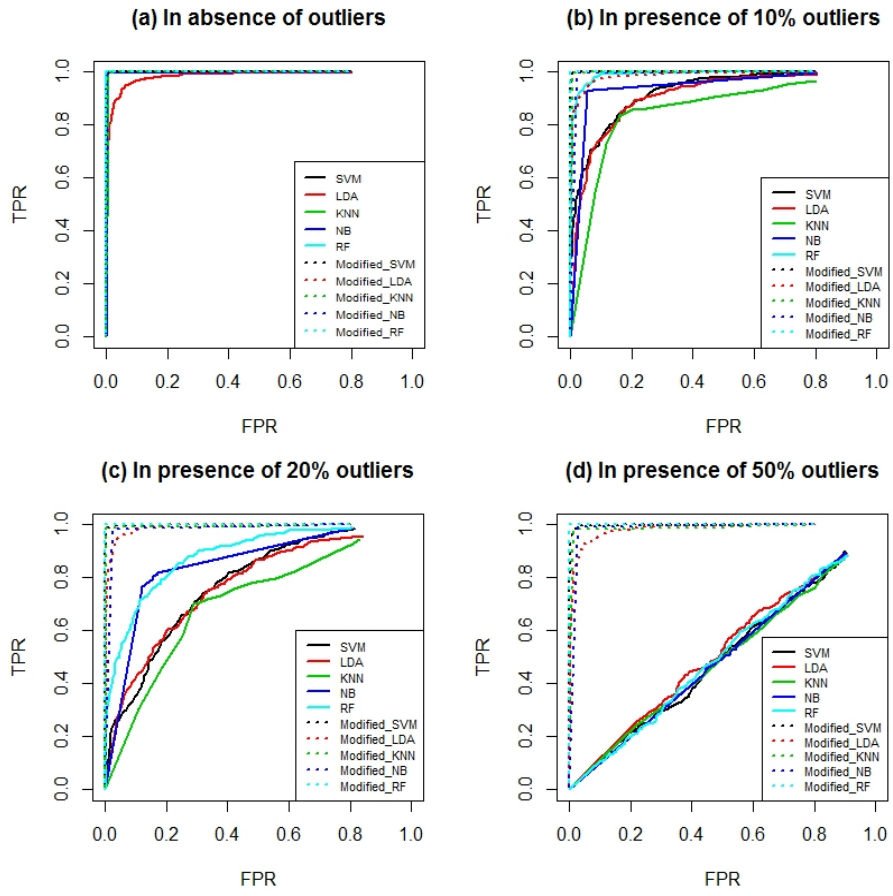

Figure 3. Performance evaluation of five classifiers using ROC curve for a small-sample case. (a) In the absence of outliers (b) in the presence of 10% outliers (c) in the presence of 20% outliers and (d) in the presence of 50% outliers.

Figures 2 & Figure S1 represent the barplot of average accuracies for small sample and large sample cases, respectively. Figures 3 & Figure S2 represent the ROC curve formed by five algorithms for small sample and large sample cases, respectively. In Figure 3 & Figure S2 the solid and dash lines indicate the performance of five classifiers in original and modified datasets, respectively. These barplots and ROC curves also describe the same results as drawn from Table 2 and Table 3. Therefore, from this simulation experiment, we may decide that the performance of the popular classifiers is improved by using the proposed method through modifying the training datasets in the presence of outliers. Otherwise, these classifiers produce the same results using original and modified datasets.

3.2. Head and neck cancer data analysis.

The gene expression profiles of head-and-neck cancer (HNC) dataset (GSE6631) were acquired from GEO database (http://www.ncbi.nlm.nih.gov/geo/). It was also used in the previous study [16]. In this work, a total of 22 paired samples were studied. As a result, the gene expression profiles of 12625 genes were obtained from 22 individuals in both normal and cancer tissues.

Table 4. Performance evaluation of five classifiers using average values of accuracies based on head-and neck cancer data.

Datasets	SVM	LDA	KNN	NB	RF
HNC	0.918 {0.930}	0.697 {0.738 }	0.908 {0.923 }	0.915 {0.924 }	0.919 {0.936 }
Colon	0.813 {0.825}	0.693 {0.746}	0.809 {0.825}	0.817 {0.825}	0.780 {0.800}

To explore the classification performance of the widespread five machine learning algorithms (SVM, LDA, KNN, NBC and RF); we constructed training and test datasets by randomly partitioning the whole HNC dataset into two independent datasets. To remove the unusual or extreme values in this dataset, the log-transformed HNC dataset was considered in this study. To make the computational simplicity, top twenty genes were selected as top twenty features using the paired sample t-test to train the five MLAs. Firstly, the training HNC dataset was employed in the proposed procedure of outlier modification to get a modified training dataset as described in section 2.2. Then the classical MLAs were applied to train their classifiers after selecting the top twenty features in both original and modified HNC datasets. Thereafter, accuracies (ACC) were measured using test HNC datasets. In Table 4, the average value of accuracies using 100 simulations was summarized. In this table, the without-parenthesis and parenthesis {} indicates that the average value of accuracies using the original HNC dataset and proposed modified HNC dataset, respectively. From this table, we noticed that three classifiers (SVM, NB and RF) produce almost similar results in comparing LDA and KNN using the original HNC training dataset.

On the other hand, these classifiers acquire better estimates using the modified HNC dataset based on the top 20 features. For example, SVM produces ACC=0.930 using the modified training HNC dataset, which is greater than ACC=0.918 using the original training HNC dataset. The boxplot of test accuracies (ACC) values is shown in Figure 4(a). This figure also supports the results of Table 4.

Table 5. KEGG pathways for top twenty features identified by t-test using modified head-and-neck cancer dataset.

KEGG ID	Pathway	No. of genes	Adj. *p*-value
hsa04512	ECM-receptor interaction	2	0.003
hsa04926	Relaxin signaling	2	0.007
hsa04510	Focal adhesion	2	0.012
hsa00514	Other types of O-glycan biosynthesis	1	0.015
hsa00534	Glycosaminoglycan biosynthesis	1	0.018
hsa05206	MicroRNAs in cancer	1	0.021
hsa05219	Bladder cancer	1	0.027
hsa05165	Human papillomavirus infection	1	0.030
hsa04151	PI3K-Akt signaling pathway	1	0.041
hsa00310	Lysine degradation	1	0.045

To reveal the KEGG pathway enrichment analysis and gene ontology (GO) of 20 features obtained from the modified HNC dataset, the functional enrichment analysis web-based tool WebGestalt was performed [39]. The GO analysis results confirmed the involvement of these 20 genes in different biological processes such as extracellular matrix organization, collagen metabolic process, extracellular structure organization, negative regulation of response to external stimulus and so on (see Table S1). The KEGG analysis shown that these genes are meaningfully enriched in ECM-receptor interaction, relaxin signaling, focal adhesion, MicroRNAs in cancer, bladder cancer, PI3K-Akt signaling pathway etc. (see Table 5). In this table, the p-values were adjusted using the Benjamini-Hochberg method [40]. Moreover, a protein-protein interaction (PPI) network of 20 features was made using the GeneMANIA database and visualize via Cytoscape [41,42] was shown in Figure 5. In this figure, the yellow and red color circle indicates the top 20 features identified by the proposed procedure, among which 12 genes (yellow color) are common between original and modified HNC data and 8 genes only identified by the proposed procedure using t-test (red color)).

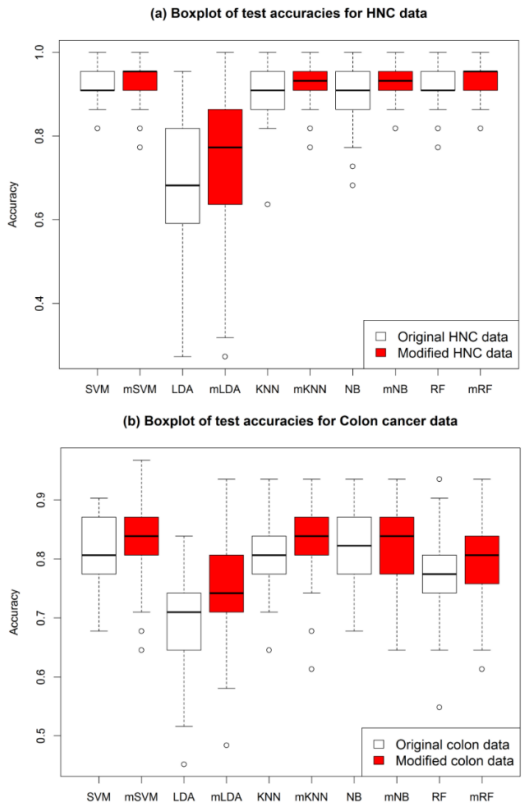

Figure 4. Performance evaluation of five classifiers using boxplot of accuracies. (**a**) for head-and-neck cancer (HNC) data; (**b**) for colon cancer data.

3.3. Colon cancer data analysis.

The expression profile of 22 healthy normal and 40 tumor tissue samples contains 6,500 transcripts. Affymetrix technology was used to generate this dataset. Among 6,500, the gene expression profiles of 2000 genes were screen out by choosing the maximum minimal intensity through the samples. This dataset can be downloaded from R package *plsgenomics* [43] and also from http://microarray.princeton.edu/oncology.

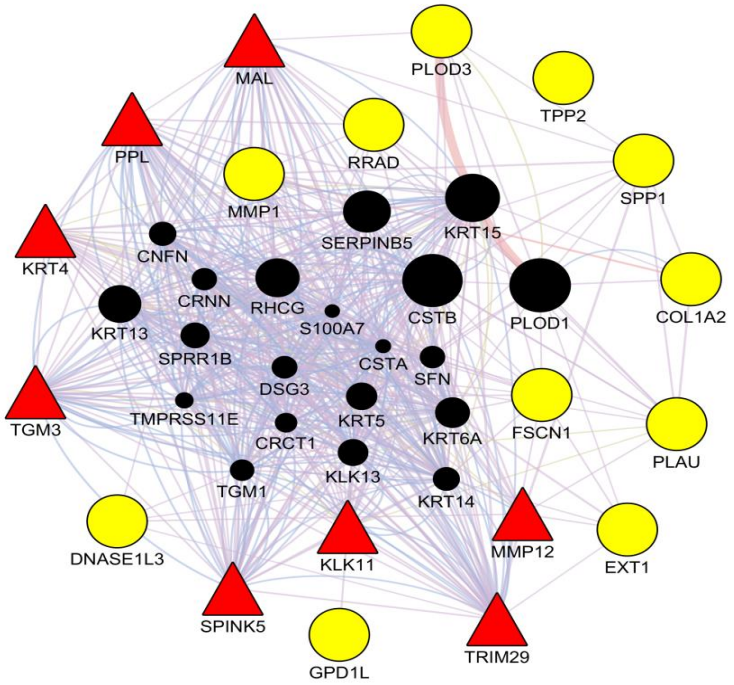

Figure 5. PPI network of top 20 features identified by a proposed procedure for HNC data.

To demonstrate the performance of the proposed procedure in a comparison of the classical procedure for classification of normal and colon samples, the entire dataset was arbitrarily divided into two independent datasets to produce a training and test dataset. The dataset was divided so that the number of training samples and a number of the test sample was the same in training and test datasets (11 normal and 22 cancer in each dataset). At first, applied the proposed outlier modification rule as described in section 2.2 in the training dataset to reconstruct the modified colon cancer training dataset. Then we selected the top 20 features using a t-test from both the original training dataset and the modified training dataset by ranking the adjusted p-values. The Benjamini-Hochberg method [19] was used to adjusted p-values. After that, the classical MLAs were employed to learn the classifiers based on the top 20 features. This process was continued by 100 times and accuracy measures were recorded. The average values of these measures were summarized in table 4. This table clarifies that SVM, KNN, NB, and RF produce better results than LDA in the original colon cancer dataset. However, these machine learning algorithms produce improved results while using the modified colon cancer dataset (bracketed values in Table 4). For example, SVM, KNN and RF produce accuracies 0.825, 0.825 and 0.800 using the proposed modified colon cancer dataset, which is larger than 0.813, 0.809 and 0.780 that are produced by the same classifiers using the original colon cancer dataset. Figure 4(b) displays the boxplot of test ACC using 100 times simulation and supported Table 4.

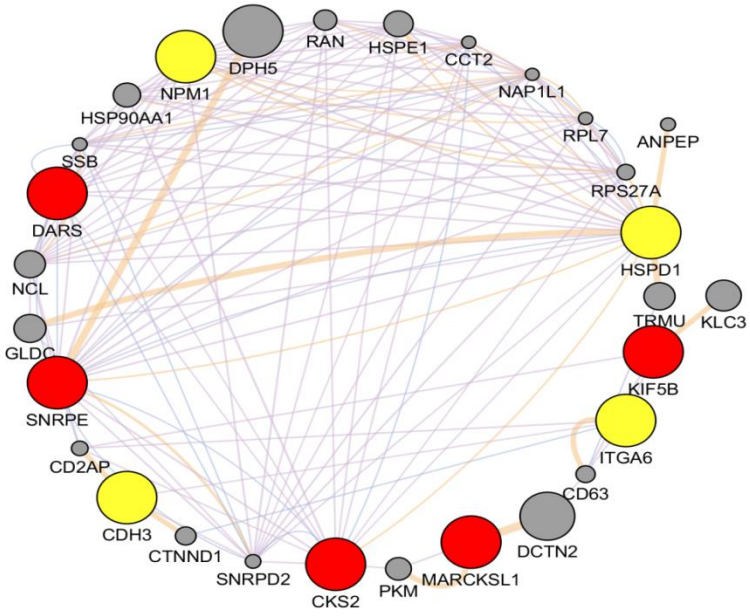

Figure 6. PPI network of top 20 features identified by a proposed procedure for colon cancer data.

To elucidate the biological functions and pathways of top 20 features obtained from the modified colon cancer dataset, we performed GO (gene ontology) and KEGG pathway enrichment analysis using webgestalt software packages [18]. Among the 20 gene bank IDs this database unambiguously mapped to 10 unique entrezgene IDs. Out of 10 IDs, 9 IDs (corresponding to genes SNRPE, HSPD1, NPM1, CKS2, CDH3, ITGA6, MARCKSL1, DARS and KIF5B) are used to annotate the functional categories. From the GO analysis, we revealed that these genes are involved in different biological processes like the molting cycle, hair cycle, response to extracellular stimulus, positive regulation of molecular function and so on (see Table S2). From the KEGG pathway analysis, we explored different pathways such as Type I diabetes mellitus, Small cell lung cancer, Cell adhesion molecules (CAMs), ECM-receptor interaction etc. (see Table 6). In addition, we also constructed a PPI–network using Cytoscape software via GeneMANIA plug-in [20], which was shown in Figure 6. Among the 9 genes, 4 genes (HSPD1, NPM1, CDH3, ITGA6) are common in the original colon dataset and 5 genes (SNRPE, CKS2, MARCKSL1, DARS, KIF5B) are uncommon, identified by a proposed procedure in modified colon datasets is shown by a yellow and red circle, respectively in Figure 6.

Table 6. KEGG pathways for top 20 features identified by t-test using modified colon cancer dataset.

KEGG ID	Pathway	No. of genes	Adj. *p*-value
hsa05222	Small cell lung cancer	2	0.004
hsa03040	Spliceosome	2	0.005
hsa04514	Cell adhesion molecules (CAMs)	2	0.010
hsa04940	Type I diabetes mellitus	1	0.017
hsa05134	Legionellosis	1	0.024
hsa00970	Aminoacyl-tRNA biosynthesis	1	0.032
hsa05412	Arrhythmogenic right ventricular cardiomyopathy (ARVC)	1	0.038
hsa05140	Leishmaniasis	1	0.040
hsa03018	RNA degradation	1	0.046
hsa04512	ECM-receptor interaction	1	0.050

4. Conclusions

Classification of samples into two or more populations is one of the key purposes of gene expression data analysis. Various machine learning algorithms have been developed to accomplish this job. However, most of them suffer from the dimensional complexity of the gene expression data matrix. To get rid of the dimensionality problem, most of the methods incorporated prior feature selection based on training dataset with their classifier. Nevertheless, this type of feature selection can also be hampered in the presence of outlying observations in the training datasets and consequently, using this preselected feature in the downstream analysis may lead to poor classification accuracies by the popular machine learning algorithms. Consequently, an outlier modification rule is proposed to modify the outlying observation in the training datasets in this paper. The performance of the proposed procedure was verified in one simulated and two real cancer gene expression datasets (HNC and colon) using five popular MLAs (SVM, LDA, KNN, NB and RF). In the simulation and real data analysis study, improved performance of the five MLAs was seen using the modified training datasets than the original training dataset, in the presence of outliers. While, in the absence of outliers, all the five MLAs produced almost the same results using modified training datasets and original training datasets.

Abbreviations

DEGs, differentially expressed genes; MAD, median absolute deviation; SVM, support vector machine; LDA, linear discriminant analysis; KNN, K-Nearest Neighbor; NB, naïve Bayes; RF, random forest; PPV, Positive predicted value; NPV, Negative predicted value.

Availability of data and material

The colon cancer data set is available at Princeton University, gene expression project, http://genomics-pubs.princeton.edu/oncology/. The R-codes of the proposed algorithm have been implemented in the R package *MLOutMod*, which can be found in https://github.com/snotjanu/MLOutMod.

Funding

This research received no external funding.

Acknowledgments

We would like to thank the referees and the journal editorial team for providing valuable advice that improved the quality of the original manuscript. This work is supported by the National Nature Sciences Foundation of China (12071096).

Conflicts of Interest

The authors declare no conflict of interest.

References

1. Li, Y.; Chen, L. Big biological data: Challenges and opportunities. *Genomics Proteomics Bioinformatics* **2014**, *12*, 187–189, https://doi.org/10.1016/j.gpb.2014.10.001.
2. Nadon, R.; Shoemaker, J. Statistical issues with microarrays: processing and analysis. *TRENDS Genet.* **2002**,

18, 265–271, https://doi.org/10.1016/s0168-9525(02)02665-3.
3. Omae, K.; Osamu, K.; Eguchi, S. Quasi-linear score for capturing heterogeneous structure in biomarkers. *BMC Bioinformatics* **2017**, *18*, 308, https://doi.org/10.1186/s12859-017-1721-x.
4. Marisa, L.; de Reyniès, A.; Duval, A.; Selves, J.; Gaub, M.P.; Vescovo, L.; Etienne-Grimaldi, M.-C.; Schiappa, R.; Guenot, D.; Ayadi, M.; Kirzin, S.; Chazal, M.; Fléjou, J.-F.; Benchimol, D.; Berger, A.; Lagarde, A.; Pencreach, E.; Piard, F.; Elias, D.; Parc, Y.; Olschwang, S.; Milano, G.; Laurent-Puig, P.; Boige, V. Gene Expression Classification of Colon Cancer into Molecular Subtypes: Characterization, Validation, and Prognostic Value. *PLoS Med.* **2013**, *10*, e1001453, https://doi.org/10.1371/journal.pmed.1001453.
5. Dam, S.; Vosa, U.; Graaf, A. van der; Franke, L.; Magalhaes, J.P. de Gene co-expression analysis for functional classifcation and gene–disease predictions. *Briefngs Bioinforma.* **2018**, *19*, 575–592, https://doi.org/10.1093/bib/bbw139.
6. Coebergh van den Braak, R.R.J.; ten Hoorn, S.; Sieuwerts, A.M.; Tuynman, J.B.; Smid, M.; Wilting, S.M.; Martens, J.W.M.; Punt, C.J.A.; Foekens, J.A.; Medema, J.P.; Ijzermans, J.N.M.; Vermeulen, L. Interconnectivity between molecular subtypes and tumor stage in colorectal cancer. *BMC Cancer* **2020**, *20*, 850, https://doi.org/10.1186/s12885-020-07316-z.
7. Pratap Singh, M.; Rai, S.; Pandey, A.; K.Singh, N.; Srivastava, S. Molecular subtypes of colorectal cancer: An emerging therapeutic opportunity for personalized medicine. *Genes Dis.* **2019**, https://doi.org/10.1016/j.gendis.2019.10.013.
8. Singh, R.K.; Sivabalakrishnan, M. Feature Selection of Gene Expression Data for Cancer Classifcation: A Review. *Procedia Comput. Sci.* **2015**, *50*, 52–57, https://doi.org/10.1016/j.procs.2015.04.060.
9. Fisher, R.A. The use of multiple measurements in taxonomic problems. *Ann. Eugen.* **1936**, *7*, 179–188, https://doi.org/10.1111/j.1469-1809.1936.tb02137.x.
10. Altman, N. An introduction to kernel and nearest-neighbor non-parametric regression. *Am. Stat.* **1992**, *46*, 175–185, https://doi.org/10.2307/2685209.
11. Le Cessie, S.; van Houwelingen, J. Ridge estimators in logistic regression. *Appl Stat* **1992**, *41*, 191–201, https://doi.org/10.2307/2347628.
12. John, G.; Langley, P. Estimating continuous distributions in Bayesian classifiers. In *Proceedings of the In: Besnard P, Hanks S (eds) Proceedings of the 17th conference on uncertainty in artificial intelligence*; Morgan Kaufmann, Ed.; USA, **1995**; 338–345.
13. Vapnik, V. *Statistical Learning Theory*; Wiley-Interscience: Chichester, **1998**;
14. Ho, T.K. Random decision forests. *Proc. Int. Conf. Doc. Anal. Recognition, ICDAR* **1995**, *1*, 278–282, https://doi.org/10.1109/ICDAR.1995.598994.
15. Shahjaman, M.; Mollah, M.M.H.; Rahman, M.R.; Islam, S.M.S.; Mollah, M.N.H. Robust identification of differentially expressed genes from RNAseq data. *Genomics* **2020**, *112*, 2000–2010, https://doi.org/10.1016/j.ygeno.2019.11.012.
16. Jubair, S.; Alkhateeb, A.; Tabl, A.A.; Rueda, L.; Ngom, A. A novel approach to identify subtype-specific network biomarkers of breast cancer survivability. *Network Modeling Analysis in Health Informatics and Bioinformatics* **2020**, *9*, 43, https://doi.org/10.1007/s13721-020-00249-4.
17. Chen, R.-C.; Dewi, C.; Huang, S.-W.; Caraka, R.E. Selecting critical features for data classification based on machine learning methods. *Journal of Big Data* **2020**, *7*, 52, https://doi.org/10.1186/s40537-020-00327-4.
18. Alelyani, S. Stable bagging feature selection on medical data. *J. Big data* **2021**, *8*, https://doi.org/10.1186/s40537-020-00385-8.
19. Omuyaa, Erick Odhiambo Okeyob, George Onyango KimwelecMichael, W. Feature Selection for Classification using Principal Component Analysis and Information Gain. *Expert Syst. Appl.* **2021**, *174*, https://doi.org/10.1016/j.eswa.2021.114765.
20. Masoudi-Sobhanzadeh, Y.; Motieghader, H.; Omidi, Y.; Masoudi-Nejad, A. A machine learning method based on the genetic and world competitive contests algorithms for selecting genes or features in biological applications. *Sci. Rep.* **2021**, *11*, 3349, https://doi.org/10.1038/s41598-021-82796-y.
21. Auwul, M.R.; Rahman, R.; Gov, E.; Shahjaman, M.; Moni, M.A. Bioinformatics and machine learning approach identifies potential drug targets and pathways in COVID-19. *Brief. Bioinform.* **2021**, *bbab120*, https://doi.org/10.1093/bib/bbab120.
22. Rostami, M.; Berahmand, K.; Forouzandeh, S. A novel method of constrained feature selection by the measurement of pairwise constraints uncertainty. *J. Big data* **2020**, *7*, https://doi.org/10.1186/s40537-020-00352-3.
23. Li, X.; Yi, P.; Wei, W.; Jiang, Y.; Le, T. LNNLS-KH: A Feature Selection Method for Network Intrusion

Detection. *Secur. Commun. Networks* **2021**, *2021*, https://doi.org/10.1155/2021/8830431.
24. Mansour, N.A.; Saleh, A.I.; Badawy, M.; Ali, H.A. Accurate detection of Covid-19 patients based on Feature Correlated Naïve Bayes (FCNB) classification strategy. *Journal of Ambient Intelligence and Humanized Computing* **2021**, https://doi.org/10.1007/s12652-020-02883-2.
25. Xu, D.; Zhang, J.; Xu, H.; Zhang, Y.; Chen, W.; Gao, R.; Dehmer, M. Multi-scale supervised clustering-based feature selection for tumor classification and identification of biomarkers and targets on genomic data. *BMC Genomics* **2020**, *21*, 650, https://doi.org/10.1186/s12864-020-07038-3.
26. Shahjman, M.; Kumar, N.; Mollah, N.H. Performance Improvement of Gene Selection Methods using Outlier Modi- fication Rule. *Curr. Bioinform.* **2019**, *14*, 491–503, https://doi.org/10.2174/1574893614666181126110008.
27. Tkachev, V.; Sorokin, M.; Mescheryakov, A.; Simonov, A.; Garazha, A.; Buzdin, A.; Muchnik, I.; Borisov, N. FLOating-Window Projective Separator (FloWPS): A Data Trimming Tool for Support Vector Machines (SVM) to Improve Robustness of the Classifier. *Front. Genet.* **2019**, *15*, 717, https://doi.org/10.3389/fgene.2018.00717.
28. Sun, H.; Cui, Y.; Wang, H.; Liu, H.; Wang, T. Comparison of methods for the detection of outliers and associated biomarkers in mislabeled omics data. *BMC Bioinformatics* **2020**, *21*, 357, https://doi.org/10.1186/s12859-020-03653-9.
29. Nnamoko, N.; Korkontzelos, I. Efficient treatment of outliers and class imbalance for diabetes prediction. *Artif. Intell. Med.* **2020**, *104*, https://doi.org/10.1016/j.artmed.2020.101815.
30. Tkachev, V.; Sorokin, M.; Borisov, C.; Garazha, A.; Buzdin, A.; Borisov, N. Flexible Data Trimming Improves Performance of Global Machine Learning Methods in Omics-Based Personalized Oncology. *Int. J. Mol. Sci.* **2020**, *21*, 713, https://doi.org/10.3390/ijms21030713.
31. Wang, C.; Long, Y.; Li, W.; Dai, W.; Xie, S.; Liu, Y.; Zhang, Y.; Liu, M.; Tian, Y.; Li, Q.; Duan, Y. Exploratory study on classification of lung cancer subtypes through a combined K-nearest neighbor classifier in breathomics. *Sci. Rep.* **2020**, *10*, 5880, https://doi.org/10.1038/s41598-020-62803-4.
32. Ala'raj, M.; Majdalawieh, M.; Abbod, M. Improving binary classification using filtering based on k-NN proximity graphs. *J. Big data* **2020**, *7*, https://doi.org/10.1186/s40537-020-00297-7.
33. Mangiola, S.; A Thomas, E.; Modrák, M.; Vehtari, A.; T Papenfuss, A. Probabilistic outlier identification for RNA sequencing generalized linear models. *NAR Genomics Bioinforma.* **2021**, *3*, https://doi.org/10.1093/nargab/lqab005.
34. Boser, B.; Guyon, I.; Vapnik, V. A training algorithm for optimal margin classes. In *Proceedings of the In: Proceedings of the 5th annual workshop on computational learning theory*; Pittsburg, USA, **1992**; 144–152.
35. Duda, R.; Hart, P. *Pattern Classification and Scene Analysis*; John Wiley & Sons: New York, NY, USA, **1973**.
36. Breiman, L. Random forest. *Mach. Learn.* **2001**, *45*, 5–32, https://doi.org/10.1023/A:1010933404324.
37. Kuriakose, A.; Chen, W.T.; He, Z.M. Selection and validation of differentially expressed genes in head and neck cancer. *Cell. Mol. Life Sci.* **2004**, *61*, 1372–1383, https://doi.org/10.1007/s00018-004-4069-0.
38. Alon, U.; Barkai, N.; Notterman, D.A.; Gish, K.; Ybarra, S.; Mack, D.; Levine, A.J. Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *PNAS* **1999**, *96*, 6745–6750, https://doi.org/10.1073/pnas.96.12.6745.
39. Liao, Y.; Wang, J.; Jaehnig, E.J.; Shi, Z.; Zhang, B. WebGestalt 2019: gene set analysis toolkit with revamped UIs and APIs. *Nucleic Acids Res.* **2019**, *47*, W199–W205, https://doi.org/10.1093/nar/gkz401.
40. Benjamini, Y.; Hochberg, Y. Controlling the false discovery rate: A practical and powerful approach to multiple testing. *J. R. Stat. Soc. Ser. B* **1995**, *57*, 289–300.
41. Mostafavi, S.; Ray, D.; Warde-farley, D.; Grouios, C.; Morris, Q. GeneMANIA : a real-time multiple association network integration algorithm for predicting gene function. *Genome Biol.* **2008**, *9*, 1–15, https://doi.org/10.1186/gb-2008-9-s1-s4.
42. Smoot, M.E.; Ono, K.; Ruscheinski, J.; Wang, P.L.; Ideker, T. Cytoscape 2.8: New features for data integration and network visualization. *Bioinformatics* **2011**, *27*, 431–432, https://doi.org/10.1093/bioinformatics/btq675.
43. Boulesteix, A. PLS Dimension Reduction for Classification of Microarray Data. *Stat. Appl. Genet. Mol. Biol.* **2004**, *3*, 1–30, https://doi.org/10.2202/1544-6115.1075.

Supplementary materials

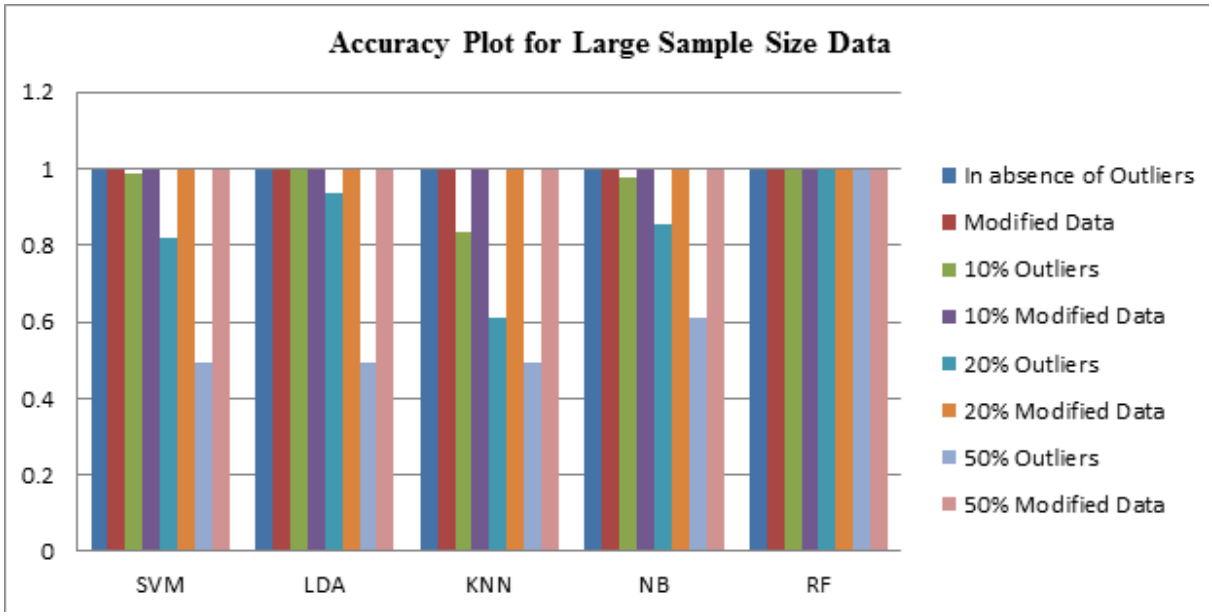

Figure S1. Performance evaluation using the average value of accuracy for large-sample case.

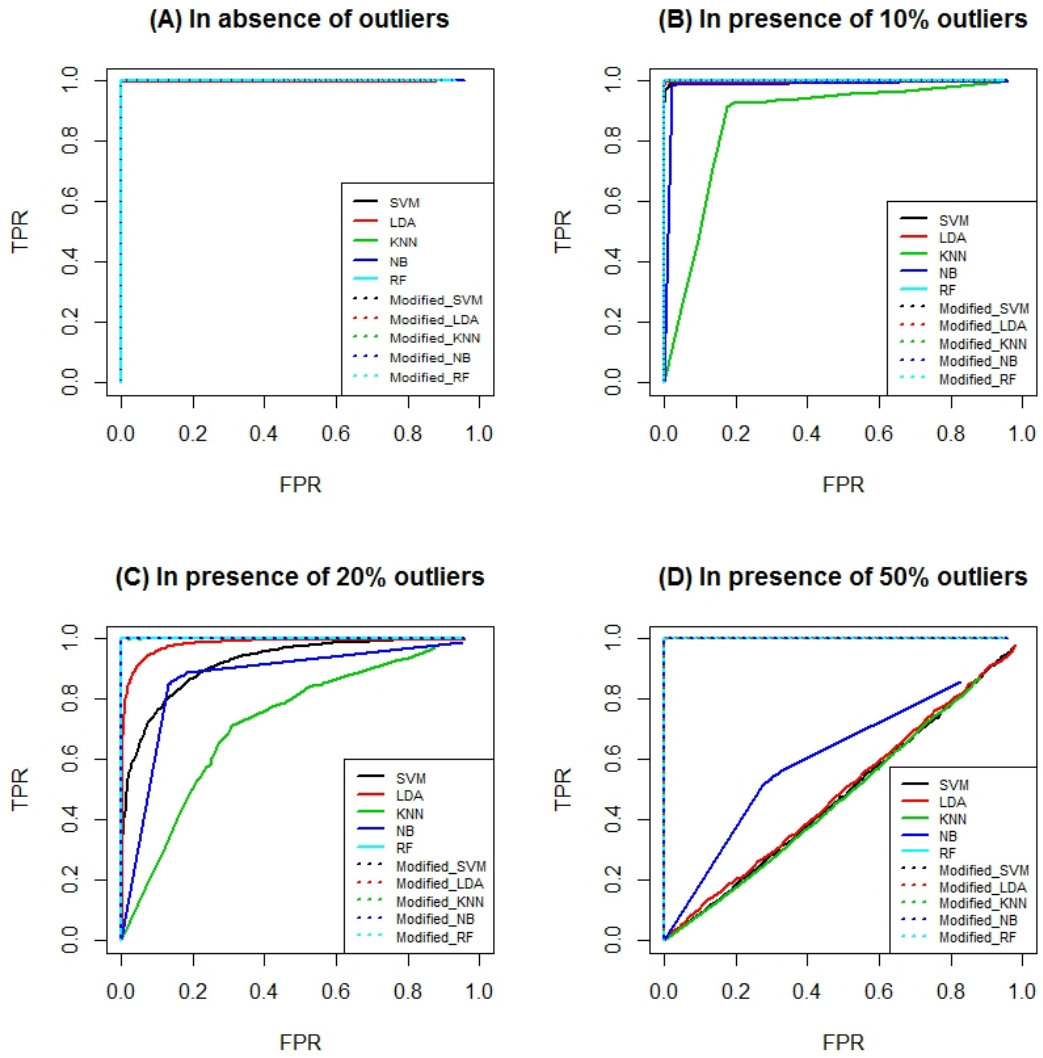

Figure S2. Performance evaluation of five classifiers using ROC curve for a small-sample case. (A) In the absence of outliers (B) in the presence of 10% outliers (C) in the presence of 20% outliers and (D) in the presence of 50% outliers.

Table S1. GO (Gene ontology) enrichment results for top 20 features identified by t-test using modified HNC dataset.

KEGG ID	Pathway	No. of genes	Adj. *p*-value
GO:0030198	extracellular matrix organization	7	4.17e-8
GO:0043062	extracellular structure organization	7	1.10e-7
GO:0009611	response to wounding	6	3.98e-5
GO:0022617	extracellular matrix disassembly	3	7.07e-5
GO:1903035	negative regulation of response to wounding	3	9.54e-5
GO:0032101	regulation of response to external stimulus	6	1.10e-4
GO:0032963	collagen metabolic process	3	1.46e-4
GO:0042060	wound healing	5	1.97e-4
GO:0032102	negative regulation of response to external stimulus	4	2.99e-4
GO:0002831	regulation of response to biotic stimulus	3	3.47e-4

The *p*-values were calculated using hypergeometric test and then adjusted by Benjamini-Hochberg method for multiple testing corrections.

Table S2. GO (Gene ontology) enrichment results for top 20 features identified by t-test using modified Colon cancer dataset.

KEGG ID	Pathway	No. of genes	Adj. *p*-value
GO:0044093	positive regulation of molecular function	6	1.72e-4
GO:0046907	intracellular transport	6	2.26e-4
GO:0071826	ribonucleoprotein complex subunit organization	3	3.48e-4
GO:0006913	nucleocytoplasmic transport	3	0.001
GO:0051169	nuclear transport	3	0.018
GO:0042303	molting cycle	2	0.001
GO:0042633	hair cycle	2	0.001
GO:0051656	establishment of organelle localization	3	0.002
GO:0006405	RNA export from nucleus	2	0.002
GO:0009991	response to extracellular stimulus	3	0.002

The *p*-values were calculated using hypergeometric test and then adjusted by Benjamini-Hochberg method for multiple testing corrections.