Journal of Computer Science

Original Research Paper

Classification of Non-Small Cell Lung Cancer Based on Gene Expression in Cases of Smokers and Non-Smokers using Ensemble Methods with Statistical Based Feature Selection

Fhira Nhita and Isman Kurniawan

Department of Informatics, Telkom University, Indonesia

Article history
Received: 05-07-2022
Revised: 30-08-2022
Accepted: 02-09-2022

Corresponding Author:
Fhira Nhita
Department of Informatics,
Telkom University, Indonesia
Email: fhiranhita@telkomuniversity.ac.id

Abstract: Lung cancer is one of the leading causes of death globally. One of the main risk factors for lung cancer is smoking, which causes more than 90% of lung cancer cases. There are two types of lung cancer, i.e., Small Cell Lung Cancer (SCLC) and Non-Small Cell Lung Cancer (NSCLC), which the latter is the most common. One method that can be used to detect cancer is the implementation of machine learning on gene expression data. Machine learning is one approach that promises good performance in classifying gene expression data. This study aimed to predict the existence of NSCLC based on gene expression, whether including NSCLC or normal. We used three data sets, i.e., GSE10072, GSE19804, and GSE19188, which relate to the cases of NSCLC in smokers and nonsmokers. The prediction was carried out using six Ensemble Methods, i.e., Random Forest, Adaptive Boosting, Extra Tree, Gradient Boosting, Extreme Gradient Boosting, and Categorical Boosting. Feature selection was carried out by calculating the correlation between feature and target according to statistical parameters, i.e., ANOVA, Mutual Information (MI), and a combination of ANOVA and MI. We obtained the prediction model that outperformed the related studies for two similar data sets with the value of accuracy for the GSE10072, GSE19804, and GSE19188 data sets 100%, 97.22%, and 100%, respectively.

Keywords: Lung Cancer, NSCLC, Gene Expression, Ensemble Methods, Smoking

Introduction

According to the 2020 Global Cancer Statistics, cancer is ranked as the leading cause of death and has become a barrier to increasing life expectancy in the world (Li *et al*., 2018; Sung *et al*., 2021). Meanwhile, lung cancer is the second most frequently diagnosed cancer and is the leading cause of death in 2020 (Pilleron *et al*., 2021; Ferlay *et al*., 2021). It is known that about sixty-seven percent of lung cancer deaths worldwide are caused by smoking behavior (Sung *et al*., 2021). Such behavior is a significant risk factor for developing lung cancer, accounting for more than 90% of lung cancer cases (Landi *et al*., 2008; Li *et al*., 2018). Concerning gender, lung and colon cancers are most common in men, especially older men (Pilleron *et al*., 2021; Sung *et al*., 2021).

There are two types of lung cancer, i.e., Small Cell Lung Cancer (SCLC) and Non-Small Cell Lung Cancer (NSCLC), while NSCLC is the most common one that causes 85% of lung cancer cases (Ren *et al*., 2020; Lai *et al*., 2020; Moitra and Mandal, 2020; Le *et al*., 2021). The diagnosis process for NSCLC patients is very complex (Chen *et al*., 2020). Generally, NSCLC patients are diagnosed using Positron Emission Tomography (PET) or CT images to detect the location and severity of the disease (Chen *et al*., 2020). However, not all images can be analyzed efficiently due to the limited medical tools and resources. The late diagnosis of NSCLC will lead to more severe treatment for the patient, such as chemotherapy and radiotherapy, with a 20% of 5-year survival rate (Chen *et al*., 2020). In contrast, the early diagnosis of NSCLC can increase 80% of the 5-year survival rate (Chen *et al*., 2020).

To accelerate the early detection of NSCLC, one alternative technique that can be applied is the machine learning method implemented on gene expression data (Karthik and Sudha, 2018). The data is obtained from microarray technology that can capture genetic information or gene expression patterns as a sign of the disease's existence, such as lung cancer

(Almugren and Alshamlan, 2019). Meanwhile, machine learning is one approach that is widely used in many cases and known to give promising results in analyzing gene expression.

Regarding the implementation of machine learning on gene expression data in disease detection, many researchers have implemented several machine learning and feature selection approaches in many cases, including NSCLC. Yang *et al*. (2018a) used Fisher exact test and Support Vector Machine (SVM) to predict NSCLC by utilizing GSE43458, GSE10072, and GSE12667 data sets. They found that the obtained model produced a satisfactory result with an accuracy is 94.83% (Yang *et al*., 2018a). Zhao *et al*. (2018) used SVM to predict NSCLC by using GSE43458 and GSE10072 data sets with accuracy is 90.7% (Zhao *et al*., 2018).

Yang *et al*. (2018b) used several feature selections, i.e., T-test, entropy, chernoff bound, and wilcoxon test. They proposed a Single-Gene Ensemble Classifier (SGEC) method to predict NSCLC by using GSE10072, GSE19804, and GSE19188 data sets. Overall, they found that the performance of SGEC is better than other machine learning methods, such as SVM, KNN, and Random Forest, with the accuracy for each data set, being 97.08%, 97.87, and 96.88%, respectively (Yang *et al*., 2018b). Ren *et al*. (2020) used decision trees, SVM, and logistic regression to predict NSCLC using GSE10072, GSE19804, and GSE19188 data sets. They found that logistic regression gives the best performance with the accuracy for each data set being 97.75%, 97.22%, and 98.72%, respectively (Ren *et al*., 2020). Rana and Osama also implemented the Extreme Gradient Boosting (XGBoost) algorithm to predict NSCLC using GSE19188 and found a satisfactory result of the model compared to SVM and gcForest with an accuracy of 95.7% (Abdu-Aljabar and Awad, 2021).

One challenge in processing gene expression data is the ability to handle high dimensions of data. Hence, appropriate feature selection methods are needed to improve the result (Almugren and Alshamlan, 2019; Bommert *et al*., 2020). According to the performed studies, we found the accuracies of similar cases are still under 100%. Besides, according to the literature survey, they conducted the features selection based on individual methods, but in this study, we performed the combination of the individual methods called overlap features. Hence, there is room for improvement to obtain a better result. Amongst several machine learning methods, the ensemble method is known as one method that is suitable for handling a high-dimensional type of data, such as gene expression data. The method combines several weak classifiers to improve the overall model performance compared to a single classifier. The ensemble methods aim to reduce variance (with bagging/bootstrap aggregating technique) and bias (by boosting technique). Hence, the ensemble method is promising to predict NSCLC with better accuracy.

In this study, we performed a comprehensive study of the ensemble methods implementation in predicting NSCLC using three data sets, i.e., GSE10072, GSE19804, and GSE19188. There are six ensemble methods used in this study, i.e., Random Forest (RF), Adaptive Boosting (AdaBoost), Extra Tree (ET), Gradient Boosting (GB), Extreme Gradient Boosting (XGBoost), and Categorical Boosting (CatBoost). However, to the best of our knowledge, there is no comprehensive study regarding ensemble method implementation to predict NSCLC using those data sets. Furthermore, we also performed feature selection to extract the most important features by calculating the correlation between feature and target according to statistical parameters, i.e., ANOVA, Mutual Information (MI), and a combination of ANOVA and MI. The performance of our proposed method is also compared with other studies using similar data sets.

Materials and Methods

Data Set

We used three data sets, i.e., GSE10072, GSE19804, and GSE19188, retrieved from GEO (Gene Expression Omnibus). Each data set has two classes, i.e., normal and NSCLC, presented in Table 1. Each data set is divided into the train and test sets with a ratio are 70:30. Fig. 1 presents the frequency of the data sets for each class in train and test sets. According to Fig. 1, we found that the number of records for each class is almost balanced for both the train and test sets. Hence we can neglect the possibility of imbalances in-class problems.

Data Sets Distribution

The Principal Component Analysis (PCA) algorithm is used to see the data distribution for each class. PCA can project high-dimensional data into a low-dimensional space by changing the input features that are mutually dependent into new independent features called principal components (Karthik and Sudha, 2018). The distribution of the three data sets is presented in Fig. 2.

As for GSE10072, we found that the distribution of data from each class is not overlapped in the train and test sets. This indicated that it is easier to be classified. As for GSE19804, the distribution of the train set and test set are not separated and there are samples in the normal class that is close to the NSCLC class and vice versa. This condition implied that the classification process for the GSE19804 is relatively complex, so it becomes a challenge to produce a good classification performance. As for GSE19188, the train set conditions are almost similar to the GSE19804, but the test set looks quite separate from the GSE10072. This condition indicated the possibility to obtain a good performance result in the test set in particular.

DOI: 10.3844/jcssp.2022.913.927

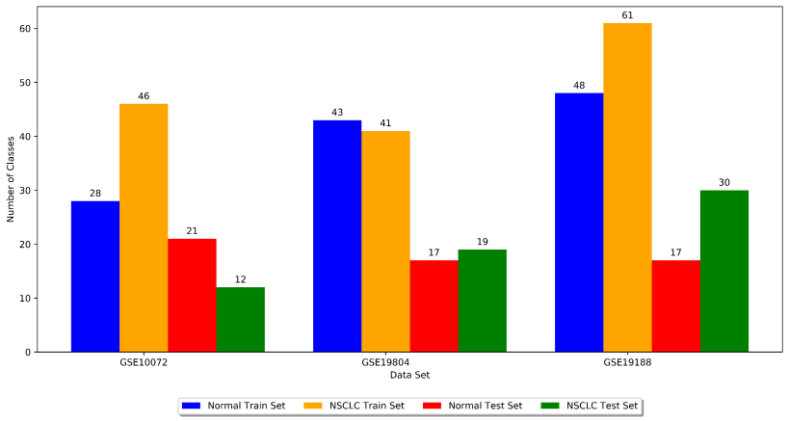


Fig. 1: Samples frequency in train and test sets

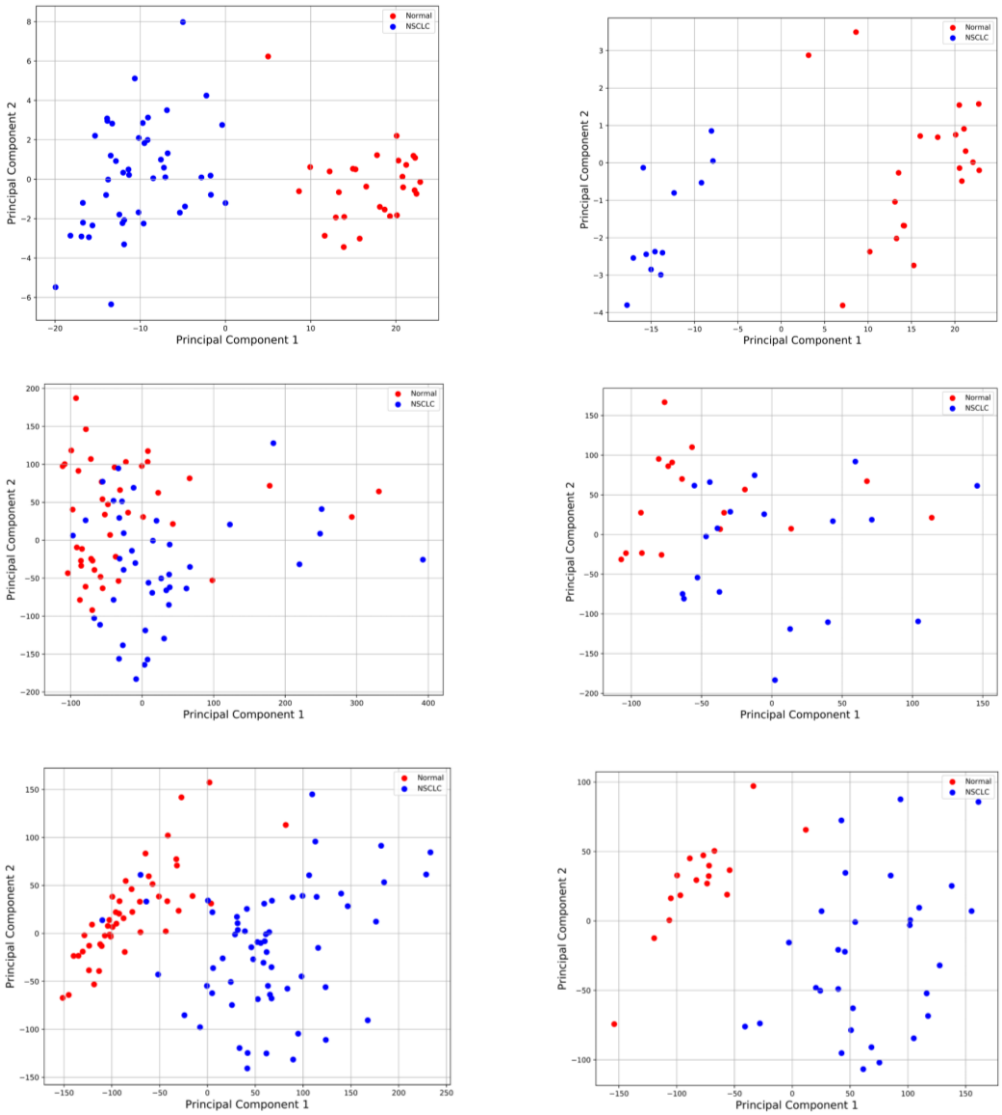


Fig. 2: Data set distribution of (a) train set and (b) test set for GSE10072, (c) train set and (d) test set for GSE19804, (e) train set and (f) test set for GSE19188 calculated using PCA

Table 1: Data sets information

Data sets	Authors	Number of genes	Classes	Number of train samples	Number of test samples
GSE10072	Landi *et al*. (2008)	250	Normal/NSCLC	74 (28/46)	33 (21/12)
GSE19804	Lu *et al*. (2010)	54675	Normal/ NSCLC	84 (43/41)	36 (17/19)
GSE19188	Hou *et al*. (2010)	54675	Normal/ NSCLC	109 (48/61)	47 (17/30)

Feature Selection

Feature selection was carried out by calculating the correlation between feature and target according to statistical parameters, i.e., Analysis of Variance (ANOVA), Mutual Information (MI), and a combination of ANOVA and MI. Those methods are classified as filter method because it is carried out before the classification process (Logotheti *et al*., 2016).

Mutual Information (MI) measures the mutual dependencies between two variables. The higher the MI value, the greater the dependence on these features. The zero value of MI indicates no relationship between feature and target (Vergara and Estévez, 2014). The calculation of MI values is performed by using Eq. (1) (Bommert *et al*., 2020):

$$MI\left(C;F\right)=E\left(C\right)-E\left(C\mid F\right) \tag{1}$$

where, *MI* (*C;F*) is mutual information for *C* and *F*, *C* is class/target, *F* is the feature, *E* (*C*) is entropy value for the class, and *E*(*C*|*F*) is entropy conditional for *C* given *F*.

Analysis of Variance (ANOVA) aims to identify potential significant differences between the mean of two or more groups (classes) by measuring between-group and within-group variations (Almugren and Alhamlan, 2019). This method is very useful to find the best features that can separate samples between two classes. The calculation for feature $a(X_a)$ is defined in Eq. (2) (Bommert *et al*., 2020; Purba *et al*., 2022):

$$f\left(X_a\right)=\frac{\sum_{i=1}^{C} n_i\left(\bar{x}_l \cdot -\bar{x}_l \cdot\cdot\right)^2/\left(c-1\right)}{\sum_{i=1}^{C}\sum_{j=1}^{n_j}\left(x_{ij}-\bar{x}_l\right)^2/\left(n-c\right)} \tag{2}$$

where, *c* is the number of classes, $\bar{x}_l$ is the mean value of X_a in class I, x_{ij} is the observed values of feature X_a for samples of class I, and $\bar{x}_{l\cdot\cdot}$ is the mean value of X_a of all samples in the data set.

Prediction Model

The prediction model used in this study is developed by using ensemble methods. The ensemble method is a meta-algorithm that aims to improve machine learning performance by combining several methods into one predictive model. The ensemble methods aim to reduce variance (with bagging/bootstrap aggregating technique) and bias (by boosting technique). The ensemble methods have a more accurate performance than the single-classifier because it combines several classifiers with the bagging or boosting approach. The ensemble methods have also been used successfully in various real-world cases (Zhou, 2012). The illustration of bagging and boosting is presented in Fig. 3.

Bagging is accomplished by bootstrap or random sampling with replacement. In this way, several random sub-datasets can be formed to create separate predictive models. Bagging performs classification in parallel, in which each model is built independently. Several algorithms that use bagging techniques are Random Forest and Extra Tree.

Boosting performs classification sequentially by developing a new model to handle the previous model's shortcomings. Boosting technique aims to strengthen the model by repeating the training for data that is still misclassified. Some of the boosting methods are AdaBoost, Gradient Boosting, XGBoost, and Cat Boost.

Decision-making on the ensemble method is done by majority vote for classification problems. Ensemble methods such as bagging can also reduce conditions of overfitting and underfitting to provide a better classification model (Altman and Krzywinski, 2017). Six ensemble methods used in this study are Random Forest (RF), Adaptive Boosting (AdaBoost), Extra Tree (ET), Gradient Boosting (GB), Extreme Gradient Boosting (XG Boost), and Categorical Boosting (Cat Boost).

Random Forest (RF) is an ensemble-based method built from several decision trees. RF is one of the popular machine learning methods that can handle high-dimensional data (Nembrini *et al*., 2018). RF builds a random decision tree using the bootstrap/bagging concept (SLRF, 2022; Ram *et al*., 2017; Kurniawan *et al*., 2020). RF performance is usually not affected by hyperparameter tuning (Logotheti *et al*., 2016). For a new observation *of* M_{new}, the output *RF* (M_{new}) of RF is predicted by Eq. (3) (Liu *et al*., 2021):

$$RF\left(M_{new}\right)=\arg\max_y \sum_{t=1}^{T} I\left(\tilde{h}_t\left(M_{new}\right)=X\right) \tag{3}$$

where, $\tilde{h}_t\left(M_{new}\right)$ is the m^{th} decision tree's prediction result with M_{new} as inputs.

Adaptive Boosting (AdaBoost) is one of the popular boosting methods (Lu *et al*., 2019). AdaBoost combines several classifiers' weaknesses to produce a robust classifier. AdaBoost works by adjusting the weights for

each cycle of the weak classifier group. AdaBoost can give better results because the diversity among classifiers is weak based on the performance of each classifier (Kurniawan *et al*., 2020).

The output of the final equation for AdaBoost classification can be represented as shown in Eq. (4) (Wang and Tang, 2020):

$$A(x) = sign\left(\sum_{m=1}^{M} \alpha_m Am(x)\right) \quad (4)$$

where, M is the train set, A_m stands for the m_{th} weak classifier, and α_m in the corresponding weight coefficient.

Extra Tree (ET) is a decision tree-based algorithm that works very randomly. The difference from RF is how the tree is built. Extra Tree looks for a threshold that separates samples into two tree branches (Logotheti *et al*., 2016). The output of the final probability ET in the testing process belonging to each of the classes is computed as the average of the probabilities on all the trees as defined in Eq. (5) (Soltaninejad *et al*., 2017):

$$p(c \mid h(x', M)) = \frac{1}{T}\sum_{t=1}^{T} p_t\left(c \mid h(x', M)\right) \quad (5)$$

where, T is the number of randomized trees, x represents the data point, M_{train} is the dataset, $h(x, M_{train})$ represents a feature vector and $p_t(c|h(x', M))$ represents the weak predictor learned by each tree.

Gradient Boosting (GB) is a powerful technique for handling various features such as noise data, recommendation systems, and weather forecasting. The main concept of GB is to build a predictive model by performing gradient descent (Prokhorenkova *et al*., 2018). The following is a gradient boosting procedure using the least squares approximation as shown in Eq. (6) (Prokhorenkova *et al*., 2018; Liu *et al*., 2017):

$$\hat{a}_l = \sum_{t=1}^{T} h_t\left(x_i\right), h_t \in H \quad (6)$$

where, t represents the number of trees, h represents the function in the functional space H and H represents the set of all possible regression trees.

Extreme Gradient Boosting (XGBoost) is an end-to-end tree boosting widely used by data scientists to achieve better results (Chen and Guestrin, 2016). In addition, XGBoost can automatically use CPU multithreading for parallel computing so that it can speed up calculations (Li *et al*., 2019). This advantage makes the model exploration process faster. XGBoost is an advanced version of GB that provides better performance and faster computing time (Abdu-Aljabar and Awad, 2021). The calculation for the objective function of XGBoost is shown in Eq. (7) (Li *et al*., 2019):

$$L = \sum_i l\left(\hat{x}_i, x_i\right) + \sum_k \Omega\left(f_k\right) \quad (7)$$

where, l is the loss function and Ω represents the function used for regularization to prevent overfitting.

Categorical Boosting (CatBoost) is a GB algorithm that trains a weak decision tree iteratively. CatBoost is a binary decision tree modified from the GB algorithm. The advantage of CatBoost is that it can get various types of data, one of which is categorical data, so it is called Categorical Boosting. CatBoost modifies the gradient calculation to avoid shifting the predictions to improve model accuracy (Bentéjac *et al*., 2021). In some cases, CatBoost gives better results than XGBoost (Prokhorenkova *et al*., 2018). The calculation for the decision tree f of CatBoost can be defined as shown in Eq. (8) (Prokhorenkova *et al*., 2018):

$$f(x) = \sum_{t=1}^{T} b_t 1_{\{x \in R_t\}} \quad (8)$$

where, R_t is the disjoint regions corresponding to the leaves of the tree.

Model Development

In this study, we defined 18 models by combining different feature selection methods and prediction models. We utilized three feature selection methods, i.e., ANOVA, Mutual Information (MI), and a combination of ANOVA and MI. The model variations used in this study are presented in Table 2, while the value of the model parameters is presented in Table 3.

Table 2: Model variations

Model	Feature selection methods	Classification methods
RF-ANOVA	ANOVA	Random Forest
AB-ANOVA	ANOVA	AdaBoost
ET-ANOVA	ANOVA	Extra Trees
GB-ANOVA	ANOVA	Gradient Boosting
XG-ANOVA	ANOVA	XGBoost
CB-ANOVA	ANOVA	Catboost
RF-MI	Mutual Information	Random Forest
AB-MI	Mutual Information	AdaBoost
ET-MI	Mutual Information	Extra Trees
GB-MI	Mutual Information	Gradient Boosting
XG-MI	Mutual Information	XGBoost
CB-MI	Mutual Information	Catboost
RF-OL	Overlap features	Random Forest
AB-OL	Overlap features	AdaBoost
ET-OL	Overlap features	Extra Trees
GB-OL	Overlap features	Gradient Boosting
XG-OL	Overlap features	XGBoost
CB-OL	Overlap features	Catboost

Table 3: Significant parameters for each method

Classification Methods	min_ samples_ leaf	min_ samples_ split	n_ estimators	learning_ rate
Random forest	1	2	100	-
AdaBoost	-	-	50	1.0
Extra trees	1	2	100	-
Gradient boosting	1	2	100	0.1
XGBoost	-	-	100	None
Catboost	-	-	None	None

Model Validation

In this study, we calculated several validation parameters, i.e., accuracy, F1-score, Precision (PC), and Recall or sensitivity (RL). The formula for each validation parameter is presented in Eq. (9) - (12) (Manju *et al*., 2019; Purba *et al*., 2022), where TP, TN, FP, and FN are True Positive, True Negative, False Positive, and False Negative, respectively (Aggarwal, 2022):

$$accuracy = \frac{TP + TN}{TP + FP + TN + FN} \times 100\% \quad (9)$$

$$RL = \frac{TP}{TP + FN} \times 100\% \quad (10)$$

$$PC = \frac{TP}{TP + FP} \times 100\% \quad (11)$$

$$F1 - score = \frac{2 \times (PC \times RL)}{PC + RL} \times 100\% \quad (12)$$

Results and Discussion

Feature Selection

We reduced the number of features by evaluating the contribution of feature number on the model performance using 5-fold cross-validation. The best number of features was searched within the range value of 2 to 20. The model performance is represented by the value of the log loss score, in which the lower value indicates the better performance of the model. The plot of feature numbers against the log loss score for GSE10072, GSE19804, and GSE19188 are presented in Fig. 4, 5, and 6, respectively. Since the score of AdaBoost in GSE10072 is significantly larger than the score of other methods, we provided two plots of the figure to highlight the fluctuation of the score.

As for the GSE10072, Fig. 4(a) and 4(c) point out that the fluctuation of the AdaBoost score is significantly higher than the score of other methods, as we mentioned before. We found the fluctuation of the AdaBoost score in both ANOVA and MI methods. Meanwhile, the score fluctuation for other methods can be observed in Fig. 4(b) and 4(d). Interestingly, we did not find the fluctuation of the XGBoost score in both ANOVA and MI methods. This indicates that the number of features does not significantly contribute to the XGBoost method. Also, we found that the change in Gradient Boosting score is more fluctuative than the score of other methods. This is confirmed by the high value of the standard deviation of the Gradient Boosting score. This point out that the method's performance is very dependent on the number of features.

Meanwhile, we found that the score of other methods, i.e., Random Forest, Extra Trees, and CatBoost, show a similar tendency, even though the absolute value is different. This might be implied that those methods have similar characteristics. Overall, we can confirm that the increase in the feature number did not guarantee an increase in the model performance.

As for GSE19804, Fig. 5(a) and 5(b) point out that the fluctuation of the AdaBoost and Gradient Boosting is significantly higher than the score of other methods. We found these fluctuations in both ANOVA and MI methods. This shows that the performance of the AdaBoost and Gradient Boosting methods is highly dependent on the number of features. Meanwhile, fluctuations in the XGBoost score are more clearly seen in this data set for the two feature selection methods used. This indicates that the number of features is sufficient to significantly contribute to the XGBoost method. Interestingly, Fig. 5(b) shows a very significant decrease in the score of a feature in the Extra Tree method. Meanwhile, we found that the scores of other methods, i.e., Random Forest and CatBoost, showed the same trend even though the scores were different. The summary of the optimal number of features, minimum log loss score, and standard deviation generated from ANOVA and MI is presented in Tables 4, 5, and 6. As for the GSE10072, Table 4 describes the optimal number of features reached by Extra Tree-ANOVA, AdaBoost-MI, and Gradient Boosting-MI with the optimal number of features are 2, 2, and 4, respectively, while the minimum log loss is 0.030, 0.000 and 0.000, respectively. This result indicates that a less number of features give a better performance model. Meanwhile, the AdaBoost gives a high value of the standard deviation in both ANOVA and MI. However, AdaBoost reached the minimum log loss in MI. This indicates that the number of features significantly contributes to the AdaBoost method.

As for the GSE19804, Table 5 describes the optimal number of features reached by CatBoost in ANOVA and MI, with the optimal number of features being 2 and 3, respectively, while the minimum log loss is 0.129 and 0.106, respectively. This confirms that the increase in the feature number did not guarantee the increase in the model performance. Meanwhile, the standard deviation reached the highest value in AdaBoost in ANOVA and MI. This point out that the method's performance is very dependent on the number of features. As for the GSE19188, Table 6

describes the optimal number of features reached by Extra Tree-ANOVA and Random Forest-MI, with the optimal number of features being 13 and 2, respectively, while the minimum log loss is 0.141 and 0.118, respectively. Meanwhile, the standard deviation reached the highest value in Extra Trees-ANOVA and Gradient Boosting-MI.

Features Selection Evaluation

We evaluated the effect of the feature selection process by comparing the model performance developed by a varied number of features, i.e., all features, ANOVA feature, MI feature, and ANOVA-MI feature (overlap features). The model performance was determined by calculating the F-1 score value. The comparison of the performance for GSE10072, GSE19804, and GSE19188 is presented in Fig. 7a, 7b, and 7c, respectively.

As for GSE10072, we found that the overlap feature gave the best results (100%) when utilized by Random Forest and CatBoost methods compared to other feature sets. This might indicate that the overlap feature can increase feature quality in both methods. Meanwhile, we found several methods, i.e., AdaBoost and Extra Trees, that give a better performance with all features. However, the higher value of the F1 score obtained by all features is not worthed as the score is the consequence of the high dimension and complexity of the model. We also found that the MI feature gives the best score in the Gradient Boosting method.

As for GSE19804, we found that the overlap features gave the highest f1-score when it utilized RF and AdaBoost. While all features and MI achieved the highest f1-score on XGBoost, ANOVA has not provided the highest f1-score for any prediction methods in this data set. Feature selection methods give the best results on RF, AdaBoost, Extra Trees, and Gradient Boosting. In comparison, all features give the same f1-score on XGBoost and CatBoost. These results indicate that many features do not always give good predictive results.

As for GSE19188, we found that RF obtained the smallest f1-score on overlap features, but the best results are 100% on other feature selections. MI gives the highest F1 score on AdaBoost, Extra Trees, and XGBoost. Interestingly, MI also gives the best F1 score in GB, which other models do not produce. Meanwhile, in CatBoost and RF, a 100% f1-score was obtained by all features, ANOVA and MI. Generally, the overlap features give the best performance in GSE10072, which reach a 100% F1 score. As for GSE19804, the highest F1-score value is 97.3%, obtained by using overlap features and MI. As for GSE19188, feature selection using ANOVA and MI gives the best results, with the F1-score being 100%. We can conclude that feature selection effectively analyzes NSCLC in gene expression data.

Validation Results

The model generated from the training process is then validated using the test set. Model performance was measured using accuracy, precision, recall, and F1-score. We consider the accuracy of the test set as the overall measurement to determine the best model. The values of the validation parameter of GSE10072, GSE19804, and GSE19188 are summarized in Table 7, 8, and 9.

As for GSE10072, we found the recall value for all models is 100%, which indicates all models' ability to predict true positives and avoid false-negative predictions perfectly. Meanwhile, the best model is obtained from model RF-OL and CB-OL with the value of accuracy and F-1 score are 100 and 100%, respectively. This point out the ability of both models to predict all of the test sets perfectly. Also, this confirmed the suitability of the overlap feature to the data set, as we discussed before. Meanwhile, we found several methods, i.e., AB-ANOVA, ET-ANOVA, GB-ANOVA, AB-MI, ET-MI, CB-MI, AB-OL, and GB-OL, that give the worst performance with the value of accuracy and F-1 score are 93.94% and 92.31% respectively.

As for GSE19804 in Table 8, we found that the precision reached the maximum score (100%) in several methods. This point out the ability of those models to classify data as positive compared to all positive predictions perfectly. However, several models provided the best recall (94.74%), which indicates the ability of those models to predict true positives and avoid the false-negative. Meanwhile, the best model is obtained from models XG-MI, RF-OL, and AB-OL, with the value of accuracy and F-1 score are 97.22 and 97.30%, respectively. This condition indicates the ability of those models to predict all of the test sets perfectly. Also, this confirms the suitability of the RF and overlap feature to the data set, as similar to GSE10072. Meanwhile, we found the GB-MI model with the worst performance, with the accuracy and F1 score value of 88.89 and 88.24%, respectively.

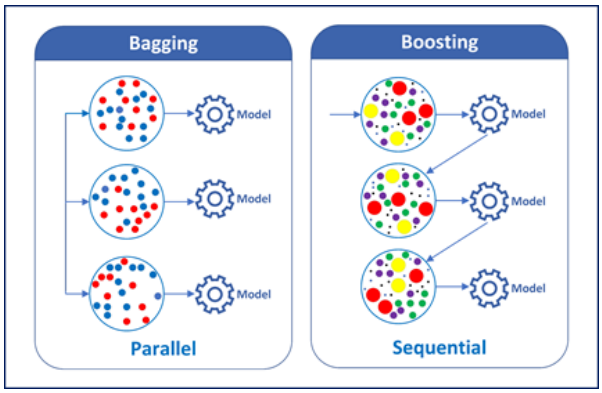


Fig. 3: Bagging and boosting illustration

DOI: 10.3844/jcssp.2022.913.927

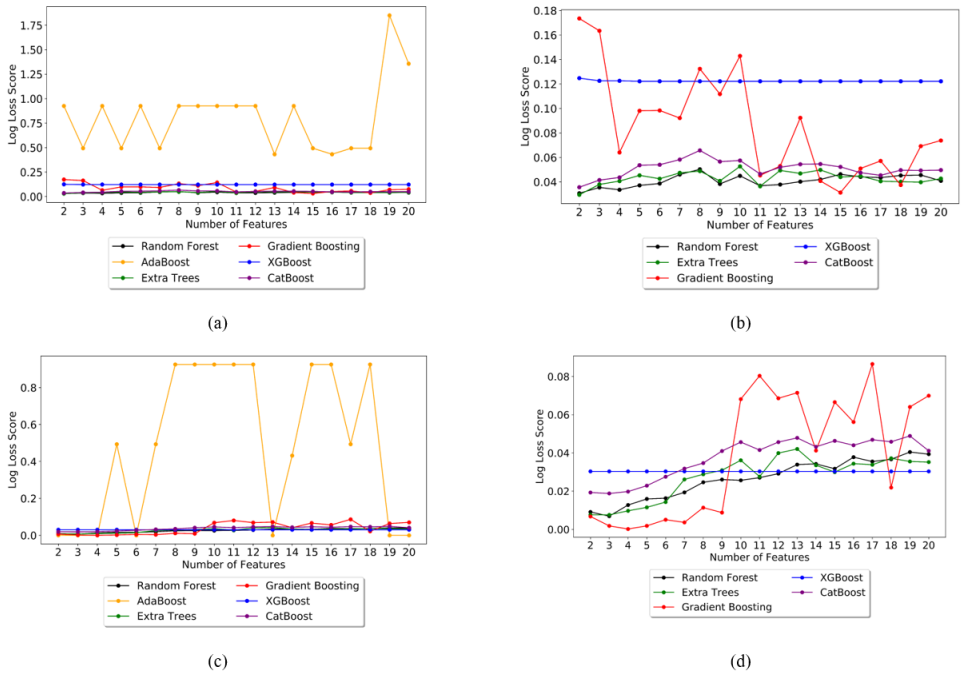


Fig. 4: The contribution of feature number to model performance for GSE10072 by using (a) ANOVA for all methods, (b) ANOVA without AdaBoost, (c) Mutual Information for all methods, and (d) Mutual Information without AdaBoost

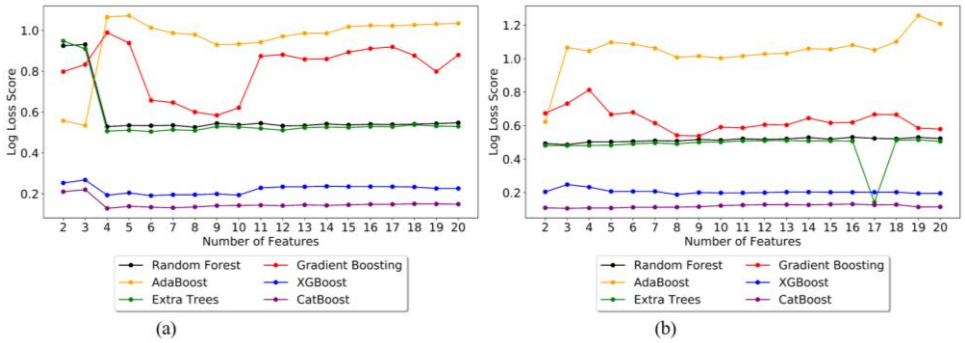


Fig. 5: The contribution of feature selection to log loss value for GSE19804 by using (a) ANOVA and (b) Mutual Information

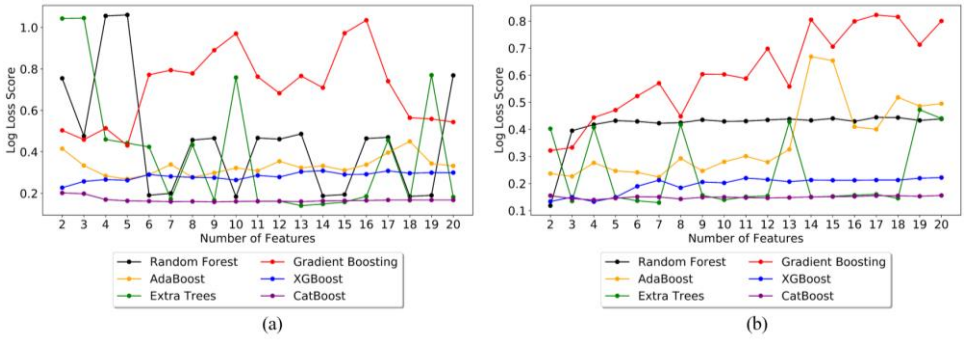
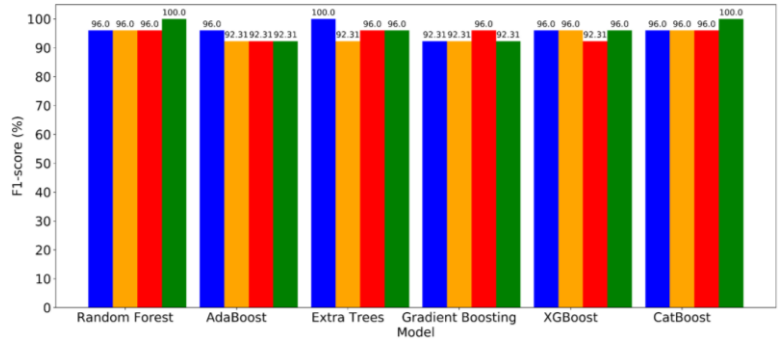
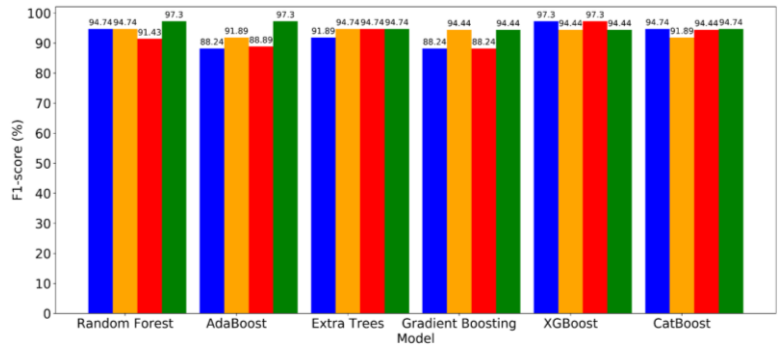


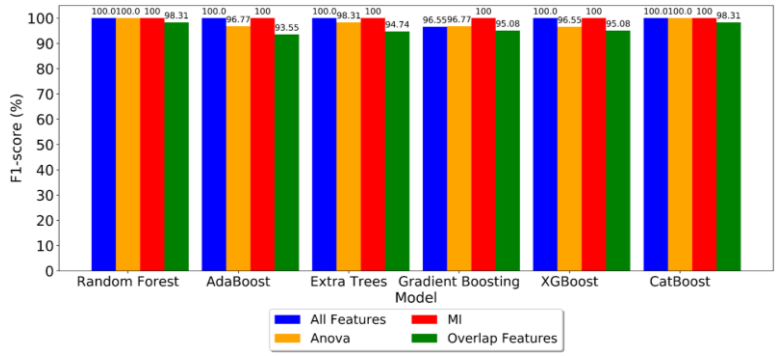
Fig. 6: The contribution of feature selection to log loss value for GSE19188 by using (a) ANOVA and (b) Mutual Information



(a)



(b)



(c)

Fig. 7: Features selection evaluation for (a) GSE10072, (b) GSE19804 and (c) GSE19188

Table 4: Summary of the features selection process for GSE10072

Methods	The optimal number of features	Minimum log loss	Std deviation
ANOVA			
Random forest	2	0.031	0.005
AdaBoost	13	0.432	0.355
Extra trees	2	0.030	0.005
Gradient boosting	15	0.031	0.042
XGBoost	5	0.122	0.001
Catboost	2	0.036	0.007
MI			
Random forest	3	0.007	0.010
AdaBoost	2	0.000	0.410
Extra trees	3	0.008	0.011
Gradient boosting	4	0.000	0.032
XGBoost	2	0.030	0.000
Catboost	3	0.019	0.010

Table 5: Summary of the features selection process for GSE19804

Methods	The optimal number of features	Minimum log loss	Std deviation
ANOVA			
Random Forest	8	0.526	0.120
AdaBoost	3	0.534	0.145
Extra Trees	6	0.504	0.125
Gradient Boosting	9	0.584	0.122
XGBoost	6	0.191	0.022
Catboost	4	0.129	0.023
MI			
Random Forest	3	0.486	0.011
AdaBoost	2	0.623	0.118
Extra Trees	17	0.141	0.081
Gradient Boosting	9	0.538	0.064
XGBoost	8	0.188	0.013
Catboost	3	0.106	0.009

Table 6: Summary of the features selection process for GSE19188

Methods	Optimal number of features	Minimum log loss	Std deviation
ANOVA			
Random Forest	10	0.184	0.272
AdaBoost	5	0.269	0.045
Extra Trees	13	0.141	0.296
Gradient Boosting	5	0.431	0.176
XGBoost	2	0.227	0.020
Catboost	9	0.158	0.012
MI			
Random Forest	2	0.118	0.071
AdaBoost	7	0.224	0.139
Extra Trees	7	0.129	0.131
Gradient Boosting	2	0.322	0.157
XGBoost	4	0.133	0.029
Catboost	4	0.139	0.004

Table 7: Validation results for GSE10072

Model	TP	TN	FP	FN	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
RF-ANOVA	12	20	1	0	96.97	92.31	100.00	96.00
AB-ANOVA	12	19	2	0	93.94	85.71	100.00	92.31
ET-ANOVA	12	19	2	0	93.94	85.71	100.00	92.31
GB-ANOVA	12	19	2	0	93.94	85.71	100.00	92.31
XG-ANOVA	12	20	1	0	96.97	92.31	100.00	96.00
CB-ANOVA	12	20	1	0	96.97	92.31	100.00	96.00
RF-MI	12	20	1	0	96.97	92.31	100.00	96.00
AB-MI	12	19	2	0	93.94	85.71	100.00	92.31
ET-MI	12	19	2	0	93.94	85.71	100.00	92.31
GB-MI	12	20	1	0	96.97	92.31	100.00	96.00
XG-MI	12	20	1	0	96.97	92.31	100.00	96.00
CB-MI	12	19	2	0	93.94	85.71	100.00	92.31
RF-OL	12	21	0	0	100.00	100.00	100.00	100.00
AB-OL	12	19	2	0	93.94	85.71	100.00	92.31
ET-OL	12	20	1	0	96.97	92.31	100.00	96.00
GB-OL	12	19	2	0	93.94	85.71	100.00	92.31
XG-OL	12	20	1	0	96.97	92.31	100.00	96.00
CB-OL	12	21	0	0	100.00	100.00	100.00	100.00

Table 8: Validation results for GSE19804

Model	TP	TN	FP	FN	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
RF-ANOVA	18	16	1	1	94.44	94.74	94.74	94.74
AB-ANOVA	17	16	1	2	91.67	94.44	89.47	91.89
ET-ANOVA	18	16	1	1	94.44	94.74	94.74	94.74
GB-ANOVA	17	17	0	2	94.44	100.00	89.47	94.44
XG-ANOVA	17	17	0	2	94.44	100.00	89.47	94.44
CB-ANOVA	17	16	1	2	91.67	94.44	89.47	91.89
RF-MI	16	17	0	3	91.67	100.00	84.21	91.43
AB-MI	16	16	1	3	88.89	94.12	84.21	88.89
ET-MI	18	16	1	1	94.44	94.74	94.74	94.74
GB-MI	15	17	0	4	88.89	100.00	78.95	88.24
XG-MI	18	17	0	1	97.22	100.00	94.74	97.30
CB-MI	17	17	0	2	94.44	100.00	89.47	94.44
RF-OL	18	17	0	1	97.22	100.00	94.74	97.30
AB-OL	18	17	0	1	97.22	100.00	94.74	97.30
ET-OL	18	16	1	1	94.44	94.74	94.74	94.74
GB-OL	17	17	0	2	94.44	100.00	89.47	94.44
XG-OL	17	17	0	2	94.44	100.00	89.47	94.44
CB-OL	18	16	1	1	94.44	94.74	94.74	94.74

Table 9: Validation results for GSE19188

Model	TP	TN	FP	FN	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
RF-ANOVA	30	17	0	0	100.00	100.00	100.00	100.00
AB-ANOVA	30	15	2	0	95.74	93.75	100.00	96.77
ET-ANOVA	29	17	0	1	97.87	100.00	96.67	98.31
GB-ANOVA	30	15	2	0	95.74	93.75	100.00	96.77
XG-ANOVA	28	17	0	2	95.74	100.00	93.33	96.55
CB-ANOVA	30	17	0	0	100.00	100.00	100.00	100.00
RF-MI	30	17	0	0	100.00	100.00	100.00	100.00
AB-MI	30	17	0	0	100.00	100.00	100.00	100.00
ET-MI	30	17	0	0	100.00	100.00	100.00	100.00
GB-MI	30	17	0	0	100.00	100.00	100.00	100.00
XG-MI	30	17	0	0	100.00	100.00	100.00	100.00
CB-MI	30	17	0	0	100.00	100.00	100.00	100.00
RF-OL	29	17	0	1	97.87	100.00	96.67	98.31
AB-OL	29	14	3	1	91.49	90.62	96.67	93.55
ET-OL	27	17	0	3	93.62	100.00	90.00	94.74
GB-OL	29	15	2	1	93.62	93.55	96.67	95.08
XG-OL	29	15	2	1	93.62	93.55	96.67	95.08
CB-OL	29	17	0	1	97.87	100.00	96.67	98.31

Table 10: Average performance for all data sets

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
RF-ANOVA	95.80	95.68	98.25	96.91
AB-ANOVA	93.78	91.30	96.49	93.66
ET-ANOVA	95.42	94.49	97.14	95.12
GB-ANOVA	94.71	93.15	96.49	94.51
XG-ANOVA	95.72	97.44	94.27	95.66
CB-ANOVA	96.21	95.58	96.49	95.96
RF-MI	96.21	97.44	94.74	95.81
AB-MI	94.28	93.28	94.74	93.73
ET-MI	96.13	94.48	98.25	95.68
GB-MI	95.29	97.44	92.98	94.75
XG-MI	98.06	97.44	98.25	97.75
CB-MI	96.13	95.24	96.49	95.58
RF-OL	98.36	100.00	97.14	98.54
AB-OL	94.22	92.11	97.14	94.39
ET-OL	95.01	95.68	94.91	95.16
GB-OL	94.00	93.09	95.38	93.94
XG-OL	95.01	95.29	95.38	95.17
CB-OL	97.44	98.25	97.14	97.68

Table 11: Performance comparison of other methods

Author	Methods	Accuracy (%)	Recall/sensitivity (%)	True negative rate/ Specificity (%)
GSE10072				
Ren *et al*. (2020)	Logistic with L_1	97.75	98.55	99.20
	SG-w	78.79	65.00	95.56
Yang *et al*. (2018b)	SVM	100.00	100.00	100.00
	Logistic with L_1	100.00	100.00	100.00
Our proposed method	Mutual information-random forest	100.00	100.00	100.00
	overlap features-random forest	100.00	100.00	100.00
GSE19804				
Ren *et al*. (2020)	Logistic with L_1	97.22	95.44	93.02
	SG-w	94.44	91.67	69.60
Yang *et al*. (2018b)	SGEC	97.87	95.89	100.00
	SVM	94.44	94.44	94.44
Our proposed method	Mutual information-XGBoost	97.22	94.74	100.00
	Overlap features-random forest	97.22	94.74	100.00
GSE19188				
Ren *et al*. (2020)	Logistic with L_1	98.72	98.59	99.03
	SG-w	89.36	100.00	83.33
Yang *et al*. (2018b)	SGEC	96.88	95.83	97.98
	SVM	95.65	100.00	90.48
Our proposed method	Mutual information-XGBoost	100.00	100.00	100.00
	Mutual information-random forest	100.00	100.00	100.00

As for GSE19188 in Table 9, we found the best model is obtained from models RF-ANOVA, CB-ANOVA, and all classification methods in MI, with the value of accuracy and F-1 score of 100% and 100%, respectively. This confirms the suitability of the MI to the data set. Meanwhile, we found the AB-OL models that give the worst performance with the value of accuracy and F-1 score are 91.49% and 93.55%, respectively. Meanwhile, the best recall value is 100% in several models i.e., RF-ANOVA, AB-ANOVA, GB-ANOVA, CB-ANOVA, and all classification methods in MI, which indicates all models' ability to predict true positive and avoid the false negative prediction perfectly.

Interestingly, the overlap features in all classification methods give the recall value, not 100%. But the overlap features reached the maximum score (100%) of precision in several methods i.e., RF-OL, ET-OL, and CB-OL. This point out the ability of those models to classify data as positive compared to all positive predictions perfectly.

Table 10 points out the average performance for all datasets. We calculated the average value of each model for all datasets. We found the best model is obtained from models RF-OL with the value of accuracy and F-1 scores are 98.36% and 98.54%, respectively. This confirms the suitability of the RF-OL to the majority of data sets, as we discussed before in GSE10072 and GSE19184. Meanwhile, we found the AB-ANOVA model with the worst performance, with the accuracy and F-1 score value of 93.78% and 93.66%, respectively. We can conclude that the overlap features most significantly contribute to two datasets (GSE10072 and GSE19188) but MI in one dataset (GSE19804).

However, the best recall value is 98.25% in several models, i.e., RF-ANOVA, ET-MI, and XG-MI, which indicates the models' ability to predict true positives and avoid the false-negative prediction almost perfectly. Besides that, the precision reached the maximum score (100%) in the RF-OL model. This point out the ability of the models to classify data as positive compared to all positive predictions perfectly.

Comparison of Competitive Methods

We also compared our results with other studies (Ren *et al*., 2020; Yang *et al*., 2018b), which used similar data sets, as shown in Table 11. The performance comparison was carried out by taking the two best methods in each study with the highest accuracy for each data set. In references, the authors used Logistic with L1 and SG-w (Ren *et al*., 2020), SVM and Logistic with L1 (Yang *et al*., 2018b) to predict NSCLC. As for GSE10072, our proposed method gives better accuracy (100%) than the reference (Ren *et al*., 2020) and similar accuracy to the reference (Yang *et al*., 2018b). This indicates that our proposed methods are suitable to process the data set.

As for the GSE19804, the highest accuracy obtained by another study is 97.87% (Yang *et al*., 2018b), with a difference of 0.65 compared to the accuracy obtained by our proposed method. Nevertheless, none of the studies achieved 100% accuracy for the GSE19804. This is challenging to explore the GSE19804 using data augmentation (Kaur *et al*., 2022) or different feature selection and classification methods for subsequent analyses. Meanwhile, as for GSE19188, we achieved 100% accuracy while all competitors did not perform.

This point out the novelty of our proposed methods that reached the optimal accuracy while using MI-XGBoost and MI-Random Forest.

Generally, we obtained better results for GSE19188 and quite similar results for GSE10072 and GSE19804. This might be related to feature selection and ensemble methods that we proposed in similar data set. We use a different feature selection with competitor studies. Also, competitive studies do not use the overlap feature. We concluded that the overlap features are suitable for the data set and contribute better to the classification process.

Conclusion

In this study, we developed six ensemble methods, i.e., Random Forest, Adaptive Boosting, Extra Tree, Gradient Boosting, Extreme Gradient Boosting, and Categorical Boosting, to classify gene expression data for NSCLC. The three data sets discussed in this study, i.e., GSE10072, GSE19804, and GSE19188, contain gene expression on NSCLC influenced by smoking. Feature selection was carried out by calculating the correlation between feature and target according to statistical parameters, i.e., ANOVA, Mutual Information (MI), and a combination of ANOVA and MI. On the overall average performance of the prediction model, overlap features or a combination of ANOVA and MI give the best results with Random Forest as the classifier. For the GSE10072 and GSE19188, our proposed method has provided the highest accuracy of 100%, while GSE19804 has not yet reached 100% and this condition is a challenge for the future. For future work, we suggested further improving the performance of GSE19804 using the data augmentation or the other feature selection and classification methods i.e., the deep learning model.

Acknowledgment

This study was funded by Telkom University (grant number: 180/PNLT3/PPM/2021).

Author's Contributions

Both the authors have equally contributed to this manuscript.

Ethics

All authors read and approved the final version of this manuscript. There are not any ethical issues to declare that could arise after the publication of this manuscript.

References

Abdu-Aljabar, R. D. A., & Awad, O. A. (2021, February). A Comparative analysis study of lung cancer detection and relapse prediction using XGBoost classifier. In *IOP conference series: Materials Science and Engineering* (Vol. 1076, No. 1, p. 012048). IOP Publishing. https://iopscience.iop.org/article/10.1088/1757-899X/1076/1/012048/meta

Aggarwal, A. K. (2022). Learning Texture Features from GLCM for Classification of Brain Tumor MRI Images using Random Forest Classifier. *Journal: Wseas Transactions on Signal PROCESSING*, 60-63.

Almugren, N., & Alshamlan, H. (2019). A survey on hybrid feature selection methods in microarray gene expression data for cancer classification. *IEEE Access*, *7*, 78533-78548. https://ieeexplore.ieee.org/abstract/document/8736725

Altman, N., & Krzywinski, M. (2017). Ensemble methods: bagging and random forests. *Nature Methods*, *14*(10), 933-935.

Bentéjac, C., Csörgő, A., & Martínez-Muñoz, G. (2021). A comparative analysis of gradient boosting algorithms. *Artificial Intelligence Review*, *54*(3), 1937-1967. https://link.springer.com/article/10.1007/s10462-020-09896-5

Bommert, A., Sun, X., Bischl, B., Rahnenführer, J., & Lang, M. (2020). Benchmark for filter methods for feature selection in high-dimensional classification data. *Computational Statistics & Data Analysis*, *143*, 106839. https://doi.org/10.1016/j.csda.2019.106839

Chen, H., Xiong, W., Wu, J., Zhuang, Q., & Yu, G. (2020). Decision-making model based on ensemble method in the auxiliary medical system for non-small cell lung cancer. *IEEE Access*, *8*, 171903-171911. https://ieeexplore.ieee.org/abstract/document/9200569

Chen, T., & Guestrin, C. (2016, August). xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining* (pp. 785-794). https://doi.org/10.1145/2939672.2939785

Ferlay, J., Colombet, M., Soerjomataram, I., Parkin, D. M., Piñeros, M., Znaor, A., & Bray, F. (2021). Cancer statistics for the year 2020: An overview. *International journal of cancer*, *149*(4), 778-789. https://doi.org/10.1002/ijc.33588

Hou, J., Aerts, J., Den Hamer, B., Van Ijcken, W., Den Bakker, M., Riegman, P., ... & Philipsen, S. (2010). Gene expression-based classification of non-small cell lung carcinomas and survival prediction. PloS one, 5(4), e10312. https://doi.org/10.1371/journal.pone.0010312

Karthik, S., & Sudha, M. (2018). A survey on machine learning approaches in gene expression classification in modeling computational diagnostic systems for complex diseases. *International Journal of* Engineering and Advanced Technology, 8(2), 182-191.

Kaur, A., Chauhan, A. S., & kumar Aggarwal, A. (2022). Prediction of Enhancers in DNA Sequence Data Using a Hybrid CNN-DLSTM Model. IEEE/ACM *Transactions on Computational Biology and Bioinformatics*. https://ieeexplore.ieee.org/abstract/document/9756940

Kurniawan, I., Rosalinda, M., & Ikhsan, N. (2020). Implementation of ensemble methods on QSAR Study of NS3 inhibitor activity as anti-dengue agent. *SAR and QSAR in Environmental Research*, *31*(6), 477-492. https://doi.org/10.1080/1062936X.2020.1773534

Lai, Y. H., Chen, W. N., Hsu, T. C., Lin, C., Tsao, Y., & Wu, S. (2020). Overall survival prediction of non-small cell lung cancer by integrating microarray and clinical data with deep learning. *Scientific reports*, *10*(1), 1-11. https://doi.org/10.1038/s41598-020-61588-w

Landi, M. T., Dracheva, T., Rotunno, M., Figueroa, J. D., Liu, H., Dasgupta, A., ... & Jen, J. (2008). Gene expression signature of cigarette smoking and its role in lung adenocarcinoma development and survival. *PloS one*, *3*(2), e1651. https://doi.org/10.1371/journal.pone.0001651

Le, N. Q. K., Kha, Q. H., Nguyen, V. H., Chen, Y. C., Cheng, S. J., & Chen, C. Y. (2021). Machine learning-based radiomics signatures for EGFR and KRAS mutations prediction in non-small-cell lung cancer. *International Journal of Molecular Sciences*, *22*(17), 9254. https://doi.org/10.3390/ijms22179254

Li, W., Yin, Y., Quan, X., & Zhang, H. (2019). Gene expression value prediction based on XGBoost algorithm. *Frontiers in genetics*, *10*, 1077. https://doi.org/10.3389/fgene.2019.01077

Li, X., Li, J., Wu, P., Zhou, L., Lu, B., Ying, K., ... & Liu, P. (2018). Smoker and non-smoker lung adenocarcinoma is characterized by distinct tumor immune icroenvironments. *Oncoimmunology*, *7*(10), e1494677. https://doi.org/10.1080/2162402X.2018.1494677

Liu, K., Hu, X., Zhou, H., Tong, L., Widanage, W. D., & Marco, J. (2021). Feature analyses and modeling of lithium-ion battery manufacturing based on random forest classification. *IEEE/ASME Transactions on Mechatronics*, *26*(6), 2944-2955. https://ieeexplore.ieee.org/abstract/document/9314252

Liu, L., Ji, M., & Buchroithner, M. (2017). Combining partial least squares and the gradient-boosting method for soil property retrieval using visible near-infrared shortwave infrared spectra. *Remote Sensing*, *9*(12), 1299. https://doi.org/10.3390/rs9121299

Logotheti, M., Pilalis, E., Venizelos, N., Kolisis, F., & Chatziioannou, A. (2016). Studying microarray gene expression data of schizophrenic patients for derivation of a diagnostic signature through the aid of machine learning. *Biom Biostat Int J*, *4*(5), 00106.

Lu, H., Gao, H., Ye, M., & Wang, X. (2019). A hybrid ensemble algorithm combining AdaBoost and genetic algorithm for cancer classification with gene expression data. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, *18*(3), 863-870. https://ieeexplore.ieee.org/abstract/document/8894520

Lu, T. P., Tsai, M. H., Lee, J. M., Hsu, C. P., Chen, P. C., Lin, C. W., ... & Chuang, E. Y. (2010). Identification of a Novel Biomarker, SEMA5A, for Non–Small Cell Lung Carcinoma in Nonsmoking WomenSEMA5A as a Novel Biomarker in Nonsmoking Lung Cancer. *Cancer Epidemiology, Biomarkers & Prevention*, *19*(10), 2590-2597. https://doi.org/10.1158/1055-9965.EPI-10-0332

Manju, N., Harish, B. S., & Prajwal, V. (2019). Ensemble feature selection and classification of internet traffic using XGBoost classifier. *International Journal of Computer Network and Information Security*, *10*(7), 37. https://j.mecs-press.net/ijcnis/ijcnis-v11-n7/IJCNIS-V11-N7-6.pdf

Moitra, D., & Mandal, R. K. (2020). Classification of non-small cell lung cancer using one-dimensional convolutional neural network. *Expert Systems with Applications*, *159*, 113564. https://doi.org/10.1016/j.eswa.2020.113564

Nembrini, S., König, I. R., & Wright, M. N. (2018). The revival of the Gini importance?. *Bioinformatics*, *34*(21), 3711-3718. https://doi.org/10.1093/bioinformatics/bty373

Pilleron, S., Soto-Perez-de-Celis, E., Vignat, J., Ferlay, J., Soerjomataram, I., Bray, F., & Sarfati, D. (2021). Estimated global cancer incidence in the oldest adults in 2018 and projections to 2050. *International Journal of Cancer*, *148*(3), 601-608. https://doi.org/10.1002/ijc.33232

Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A. V., & Gulin, A. (2018). CatBoost: Unbiased boosting with categorical features. *Advances in neural information processing systems*, *31*. https://proceedings.neurips.cc/paper/2018/hash/14491b756b3a51daac41c24863285549-Abstract.html

Purba, D. N., Nhita, F., & Kurniawan, I. (2022). Implementation of Ensemble Method in Schizophrenia Identification Based on Microarray Data. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, *6*(1), 64-69. https://doi.org/10.29207/resti.v6i1.3788

Ram, M., Najafi, A., & Shakeri, M. T. (2017). Classification and biomarker genes selection for cancer gene expression data using random forest. *Iranian Journal of Pathology*, *12*(4), 339. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5844678/

Ren, Y., Yang, Z. Y., Zhang, H., Liang, Y., Huang, H. H., & Chai, H. (2020). A Genotype-Based Ensemble Classifier System for Non-Small-Cell Lung Cancer. *IEEE Access*, *8*, 128509-128518. https://ieeexplore.ieee.org/abstract/document/9139244

SLRF. (2022). Scikit Learn-Random Forest https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html#sklearn.ensemble.RandomForestClassifier.

Soltaninejad, M., Yang, G., Lambrou, T., Allinson, N., Jones, T. L., Barrick, T. R., ... & Ye, X. (2017). Automated brain tumour detection and segmentation using superpixel-based extremely randomized trees in FLAIR MRI. *International Journal of Computer Assisted Radiology and Surgery*, *12*(2), 183-203. https://link.springer.com/article/10.1007/s11548-016-1483-3

Sung, H., Ferlay, J., Siegel, R. L., Laversanne, M., Soerjomataram, I., Jemal, A., & Bray, F. (2021). Global cancer statistics 2020: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries. *CA: A Cancer Journal for Clinicians*, *71*(3), 209-249. https://doi.org/10.3322/caac.21660

Vergara, J. R., & Estévez, P. A. (2014). A review of feature selection methods based on mutual information. *Neural computing and applications*, *24*(1), 175-186.

Wang, J., & Tang, S. (2020). Time series classification based on arima and adaboost. In *MATEC Web of Conferences* (Vol. 309, p. 03024). EDP Sciences. https://doi.org/10.1051/matecconf/202030903024

Yang, L., Sun, L., Wang, W., Xu, H., Li, Y., Zhao, J. Y., ... & Zhang, L. Y. (2018a). Construction of a 26-feature gene support vector machine classifier for smoking and non-smoking lung adenocarcinoma sample classification. *Molecular Medicine Reports*, *17*(2), 3005-3013. https://doi.org/10.3892/mmr.2017.8220

Yang, Z., Ren, Y., Zhang, H., & Liang, Y. (2018b, August). Microarray-Based Cancer Prediction Using Single-Gene Ensemble Classifier. In *International Conference on Intelligent Science and Big Data Engineering* (pp. 589-600). Springer, Cham. https://link.springer.com/chapter/10.1007/978-3-030-02698-1_51

Zhao, J., Cheng, W., He, X., Liu, Y., Li, J., Sun, J., ... & Gao, Y. (2018). Construction of a specific SVM classifier and identification of molecular markers for lung adenocarcinoma based on lncRNA-miRNA-mRNA network. *Onco Targets and therapy*, *11*, 3129. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5975616/

Zhou, Z. H. (2012). *Ensemble methods: Foundations and algorithms*. CRC press. ISBN-10: 1439830037