A Stacked Autoencoder based Gene Selection and Cancer Classification Framework

Madhuri Gokhale[a,b,*], Sraban Kumar Mohanty[b] and Aparajita Ojha[b]

[a]Department of Computer Science & Engineering, Jabalpur Engineering College, Jabalpur, 482001, India,
[b]PDPM Indian Institute of Information Technology, Design and Manufacturing, Jabalpur, 482005, India,

ARTICLE INFO

Keywords:
Gene selection
Cancer classification
Stacked autoencoder

ABSTRACT

Cancer is one of the most common causes of death worldwide and is, therefore, a prominent area of biomedical research. Cancer is a genetic disease in which improperly functioning genes tend to change expressions. Thus, gene expression analysis is utilized for early diagnosis of cancer prognosis, and therapy prediction in a clinical environment. Usually, some dominant genes among thousands of them play an important role in the diagnosis of cancer. But designing a suitable framework to find out the key set of genes is a challenging task. Numerous gene selection approaches have been introduced by researchers for cancer classification, using statistical, or traditional feature selection methods. In recent years, deep learning methods have also been applied for gene selection using autoencoder networks. However, improving the accuracy of cancer classification still remains a challenging task. In the present paper, a stacked autoencoder-based framework is proposed for gene selection and cancer classification. Nine different classifiers are employed to evaluate the performance of the gene selection model. Then the best performing combination of gene selection and cancer classification models are chosen to finally select the genes. Random Forest and Support Vector Machine show better performance on ten different benchmark datasets, when the gene selection is done using the stacked autoencoder. The classifier with the highest accuracy is selected to build the cancer classification model. The proposed model outperforms seven existing methods on all the ten datasets.

1. Introduction

Globally, around 18.1 million new cases of cancer are reported every year along with 9.6 million deaths, accounting for the second-largest cause of death [1, 2]. Study on cancer is one of the most vital areas of biological studies. Accurate cancer type prediction is extremely useful in terms of ensuring better care and reducing patient toxicity. Systematic approaches based on gene expression analysis have been proposed by researchers to obtain a deeper understanding of the gene expression data and its role in cancer detection. The development of microarray technologies has resulted in a plethora of techniques for gene expression analysis. Microarray data on gene expression, however, has several issues which pose challenges for cancer detection. First, it is high dimensional with hundreds to tens of thousands of genes. Second, publicly available datasets are limited. Third, the majority of genes do not have any role in cancer detection and identification [3]. In general, irrelevant and redundant genes in high-dimensional gene expression datasets adversely affect the training of machine learning algorithms by degrading their learnability. Scientists have proposed a variety of gene selection approaches to find the most discriminating genes, or biomarkers [4]. Gene selection is a common dimensionality reduction technique to prevent issues like computational complexity, overfitting, and low model interpretability.

Gene expression data inherently exhibits a class imbalance distribution, in which samples from some groups called majority classes to outnumber those from others (minority classes). The problem of class imbalance results in a learning bias in favor of the majority classes, lowering prediction capability in minority classes. The training data in most traditional machine learning algorithms are assumed to be balanced [5]. For imbalanced gene expression datasets, traditional classification methods with correct classification rates (CCR) can favor the majority classes. Minority class misclassification usually has greater adverse effect on cancer classification results. Therefore, when analyzing gene expression data, the problem of imbalanced distribution of classes need to be addressed before training a machine learning (ML) classifier [6].

Deep Learning (DL) has enormous potential to assist medical professionals by reducing human intervention in cancer detection. Deep learning is realized through neural networks that are commonly used in the analyzing of unstructured data. It is a type of ML that learns a hierarchical representation of the given data to build efficient models capable of analysing data. Such methods identify important characteristics during the training phase without pre-engineering complex data [7]. Due to these capabilities, DL algorithms have surpassed traditional ML techniques in a variety of fields, including pattern classification, processing of medical image, and natural language processing, etc. DL algorithms have also proved to be effective in cancer prediction based on gene expression data [8, 9, 10]. These algorithms have been used to identify the most influential genes for prediction of cancer and its type. Deep autoencoders (AE) in particular, have been explored by many researchers for selecting the most relevant genes from gene

*Corresponding author
ORCID(s):

expression datasets. Denoising autoencoders [11], contractive autoencoders [12], sparse autoencoders [13], regularised autoencoders [14], stacked autoencoder [15] and variational autoencoder [16], are some of the autoencoders types that have been successfully applied by researchers for gene selection task.

Fakoor et al. [13] have employed a sparse autoencoder with a single hidden layer and sigmoid transfer function for gene expression data dimensionality reduction. Tan et al. [11] have utilized a denoising autoencoder with a cross-entropy loss and stochastic gradient descent for relevant gene selection. In Danaee et al. [17], an autoencoder with four-layer architecture is used with stochastic gradient descent optimization technique to select the most prominent genes. Way et al. [16] have applied a variational autoencoder having three hidden layers. In another work by Zhang et al. [18], an innovative yet simple technique for predicting breast cancer patients has been developed using a deep autoencoder. They combine raw gene expression data with principal component analysis (PCA) components during the preprocessing step. The data is then supplied into an autoencoder model, which selects a smaller collection of genes. These techniques have produced good cancer classification results using different gene selection methods. In the majority of the research works mentioned above, the traditional dimensionality reduction method like PCA is initially applied to remove redundant and irrelevant features then a DL model is involved for final gene selection. In contrast to PCA, the DL model can learn non-linear characteristics of the data for better representation learning.

Hinton et al. [19] demonstrated that a multi-layered feedforward neural network may be efficiently pretrained one layer at a time for learning data representation. A similar notion was proposed by Bengio et al. [20] and LeCun et al. [21] where unlabelled data is used to initialize the weights in a greedy layer-by-layer fashion, and then labeled data is used to fine-tune the network. A Stacked autoencoder (SAE) is an unsupervised layer by layer learning model that extracts data features suitable for classification. The SAE is utilized for deep representation learning as this enhances network performance [22]. Stacked autoencoders have shown good performance in the gene selection task as well [13, 15, 23]. Recently, Kamel et al. [23] have proposed a gene selection and classification model where a stacked autoencoder is applied for most relevant gene selection. They have employed three different classifiers to evaluate the performance of the stacked autoencoder. However, their study is limited to only one dataset on breast cancer.

In the present paper, a gene selection and classification framework for cancer detection is proposed that uses an autoencoder for gene selection and nine different classifiers for classification of the selected genes. The best performing classifier is chosen for building the gene selection and cancer classification model. We have experimented with a vanilla autoencoder and a stacked autoencoder for gene selection task. The vanilla autoencoder is a simple neural network based encoder-decoder model that is trained to reconstruct its input. The advantage of a vanilla autoencoder is that its training is simple and computationally convenient. Stacked autoencoder on the other hand is designed for feature extraction. Main highlights of the present work are as follows.

- A framework for gene selection and cancer classification is proposed that employs an autoencoder and a classifier from a bucket of nine classifiers.
- Selection of the autoencoder and related hyper parameters are done by an empirical study on ten different gene expression datasets.
- The number of most prominent genes is decided by iteratively decreasing the number of genes and finding the least possible number for effective cancer classification.
- The model selected through the proposed framework outperforms seven other existing models on all the ten datasets.
- Biological significance of the results has also been analysed.

The remainder of this paper is structured as follows. Section 2 covers a review of the state-of-the-art in gene selection and classification methods. Section 3 discusses the proposed gene selection and classification framework. Section 4 discusses the experimental findings with ten gene expression data. Biological significance of the selected genes is discussed in Section 5. Finally, the paper concludes with future scope in Section 6.

2. Related Work

This section summarises prior work in the field of gene selection which is also called feature selection (FS) strategies that employ ML and DL approaches. Wrapper, filter, and hybrid approaches are the three primary types of feature selection methods that are also used in gene selection [24] (see also, [25]). Our research concentrates on filter gene selection methods to pick top-ranked genes that accurately classify cancer and its type.

Many filter gene selection techniques have been proposed over the years. Lazar et al. [26] provide a comprehensive summary of current filtering techniques. Rajapakse et al. in [27] proposed a multi-class gene selection method using both parametric and non parametric test. In this method F-score and Kruskal-Wallis (KW)-score test were applied on a synthetic and six real gene expression datasets for gene selection. Selected genes are analyzed using the Pareto-Fronts approach. Almutiri et al. in [28] have proposed a combined approach of chi-square test with Support Vector Machine Recursive Feature Elimination (SVM-RFE) for gene selection and have tested their method on eleven high dimensional microarray datasets. Selected genes are then classified on five well known ML classifiers. Their method achieves highest accuracy with Artificial neural networks (ANN).

Table 1
Characteristics of previous gene selection methods in literature

Work	Dimensionality Reduction	Feature Selection	Classification Methods	No. of datasets
Rajapakse et al.[27]	-	F-score and Kruskal-Wallis (KW)	Pareto-Fronts	6
Almutiri et al.[28]	Chi-square test	SVS-RFE	RF, KNN, NB, ANN, SVM	11
Maniruzzaman et al.[29]	-	Kruskal-Wallis,t-tests,F-test and Wilcoxon sign rank-sum	LDA, QDA, NB,GPC, RF, ANN, LR, DT, AB	1
Hoque et al.[30]	-	Mutual Information	SVM, DT, RF,KNN	5
Rani et al.[31]	Mutual Information	Genetic Algorithm	SVM	3
Mandal et al.[32]	Mutual Information	mRMR	SVM	4
Yuanyu et al.[6]	-	ReliefF	KNN, SVM	4
Danaee et al. in [17]	-	Denoising Autoencoder	SVM, SNM-RFE, ANN	1
Fakoor et al.[13]	PCA	Sparse autoencoder	SVM, softmax	13
Liu et al.[15]	Inf-FS	Denoising Autoencoder	softmax	3
Zhang et al.[18]	PCA	Autoencoder	AdaBoost	1
Uzma et al.[33]	PCA,correlation, spectral-based	Autoencoder and Genetic Algorithm	SVM,KNN,RF	6
Kemal Adem [23]	-	Stacked autoencoder	Softmax, KNN, SVM	1

Maniruzzaman et al. in [29] have applied four statistical tests for gene selection namely, Kruskal-Wallis, T-tests, Wilcoxon sign rank-sum (WCSRS), and F-test on colon dataset. Selected genes are then used for classification on ten machine learning classifiers. Using a combination of WCSRS test and random forest based classifier, the Model achieves 90.50% mean accuracy. Hoque et al. in [30] have utilized the mutual information between the gene and the class label for determining the gene relevance and redundancy. Support Vector Machine (SVM), Decision tree (DT), Random forest (RF), K-Nearest neighbour (KNN) classifiers are used to evaluate the accuracy for original and reduced dimension of datasets. They have used two gene expressions datasets and three other types of datasets for building a cancer classification model. Rani et al. in [31] have presented a mutual information and genetic algorithm (MI-GA) gene selection method that works in two steps. Initially, mutual information is used to pick only those genes with strong information about cancer, followed by a Genetic Algorithm-based final gene selection to determine and choose the best collection of genes. SVM classifier is used to classify selected genes on Colon cancer, Lung cancer, and Ovarian cancer gene expression datasets. Classification accuracy obtained 96.77%, 81.37% and 98.42% for Colon cancer, Lung cancer, and Ovarian cancer data respectively.

Mandal et al. in [32] discussed the generation of the final gene set by selecting the most relevant and non-redundant genes. This final gene set contains genes that are most relevant to the class and have the minimum correlation with one another. Overian Cancer, ALL/AML Leukemia, Prostate Cancer and Childhood ALL four microarray gene expression were used to test the suggested technique. Classification accuracy obtained 99.18%, 100%, 96.08% and 88.18% for Overian Cancer, ALL/AML Leukemia, Prostate Cancer and Childhood ALL dataset respectively. Relief-based algorithms [25, 34] is one of the most used filter gene selection methods in gene expression analysis. Yuanyu et al. in [6] have proposed a modified relief algorithm for gene selection called imRelief. This algorithm corrects the bias towards the majority classes and prevents the majority classes' impact while calculating gene weights and select genes for classification. This approach uses four gene expression datasets for gene selection and KNN classifier to classify selected genes.

Fakoor et al. in [13] have proposed a method to first reduce the dimensionality using a principal component analysis (PCA), and then they have applied the results of PCA to sparse stacked autoencoder to get a sparse representation of data. Finally they use SVM to build the classification model. They have evaluated their model on 13 gene expression datasets to improve the performance. Danaee et al. in [17] have proposed a Stacked Denoising Autoencoder (SDAE) model to effectively extract functional genes from high-dimensional gene expression data. They have used supervised classification models Support Vector Machine (SVM), Support Vector Machine with radial basis function kernel (SVM-RBF) and ANN to assess the performance of the retrieved genes for breast cancer diagnosis and 98.26% accuracy was achieved by SVM-RBF.

Liu et al. in [15] have used the concept of a denoising autoencoder to extend gene expression data samples. Initially, Infinite Feature Selection (Inf-FS) is employed to pick genes on extended data. Then, using a stacked autoencoder model, selected genes are further reduced and categorized using a softmax classifier on Colon, Breast cancer, and Leukemia high-dimensional gene expression data. Classification accuracy of 57.89%, 84.49%, and 87.33% is obtained for Leukemia, Colon, and Breast cancer datasets,

respectively using top 500 selected genes. Zhang et al. in [18] have proposed an unsupervised gene selection and supervised classification mechanism. Firstly the PCA is used to reduce redundant and noisy data. Then the PCA-derived genes combined with the raw gene expression data and fed to an autoencoder model, which finally selects the reduced gene set. The genes obtained by the proposed two-phase technique are classified using the AdaBoost algorithm on a breast cancer dataset, the maximum classification accuracy attained is 85%. Uzma et al. in [33] have also used a Gene encoder model, principal component analysis, correlation, and spectral gene selection approach to select the initial level of genes. The genetic algorithm is then applied that uses autoencoder based clustering to assess the chromosomes. The obtained gene subsets are classified by support vector machine, k-nearest neighbors, and random forest to evaluate the performance on six benchmark gene expression datasets. Kemal Adem in [23] have proposed a technique for disease identification on the breast cancer microarray gene expression dataset, in which the most important genes are chosen using a stacked autoencoder to improve classification performanc. KNN, SVM, and Subspace kNN are used to test the performance of the gene selection method and 91.24% accuracy was achieved by Subspace kNN.

Table 1 summarises the key aspects of previous gene selection methods in the literature. From the existing works, it is noted that all gene expression data sets are high dimensional and imbalanced, and approximately all unsupervised approaches employ principal component analysis (PCA), AE or any other method to eliminate unnecessary or redundant genes before applying a classification model. The second major issue in the gene expression datasets is the class imbalance in data distribution. The proposed framework addresses both these issues by data augmentation and selecting an optimal autoencoder model for gene selection and dimensionality reduction. A high dimensional gene expression dataset is first subjected to the process of data augmentation. Then the augmented dataset is fed to an autoencoder for effective gene selection. Nine different classifiers are used to evaluate the gene selection performance of two different autoencoder models: vanilla and stacked autoencoders. Then the classifier(s) that consistently shows high performance on all the ten benchmark datasets is selected to build the gene selection and cancer classification model. To avoid being reliant on a single classifier, we have used nine separate ones These are Quadratic Discriminant Analysis (QDA), Linear Discriminant Analysis (LDA), Naive Bayes (NB), Gaussian Processes for Classification (GPC), Support Vector Machine (SVM), Logistic Regression (LR), Decision Tree (DT), AdaBoost (AB), and Random Forest (RF).

3. Gene selection and cancer classification framework

In this section, we present a gene selection and cancer classification framework using an autoencoder and a standard supervised learning method. The framework is depicted in Figure 1. Since the gene expression data is high dimensional, and the number of samples in most of the publicly available datasets are less with class imbalance issues, a data augmentation technique is initially applied on the dataset, then the data is prepossessed, and the autoencoder is trained on the augmented dataset $S = \{(X_i, y_i) : i = 1, \ldots, n\}$, where X_i is the gene expression data and y_i is its label. One can choose an autoencoder from a collection of candidate autoencoders. In the present paper, we chose a vanilla neural network autoencoder that reconstructs its input, and a stacked autoencoder for experimentation. Once the autoencoder is trained, the latent vector z_i produced by it for each data sample X_i is stored in a set $Q = \{(z_i, y_i), i = 1, \ldots, n\}$ alongwith the original labels y_i. Then a classifier is trained for gene expression classification task using the set Q. From a set of candidate classifiers, the best performing classifier is finally selected for building the classification model. For the present work, nine different classifiers are evaluated that are most frequently used for general machine learning tasks. We elaborate all the steps and components of the complete framework in the following sections.

3.1. Data augmentation

Wide gene expression size and a small number of data samples characterize the gene expression datasets that are currently publicly accessible. Data augmentation is often a standard method for increasing the size of a dataset, and to address the class imbalance problems. Data augmentation makes it possible to create new data samples from an original set of samples. There are several approaches for data augmentation. Adding noise in the data samples to create new samples is one such classical techniques, in which mostly Gaussian noise is mixed with the original data. It is a statistical noise with the normal distribution, also known as Gaussian distribution. It is defined as follows.

$$N(X) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(x-\mu)^2 \big/ 2\sigma^2} \tag{1}$$

where sample mean $\mu = 0$ and standard deviation $\sigma = 1$. One of the primary motivations why Gaussian noise is taken for data augmentation is that Gaussian perturbation creates samples that have similarity with original data distribution. In the present work, Gaussian noise is applied with a noise factor of 0.1 to data samples to generate new samples. For this, $10 - 20\%$ samples of each class are selected randomly, and a random noise sample is generated with mean equal to sample mean and standard deviation equal to 1 using the equation 1. Then the selected samples are subjected to the random noise to generate noisy samples. These noisy samples are added to original samples to overcome the problem of the class imbalanced dataset.

3.2. Data Preprocessing

Since the large dimensions of biological data contain a high amount of noise and bias, the proposed work employs a data preprocessing step. It is known that variables evaluated at various scales do not contribute equally to model

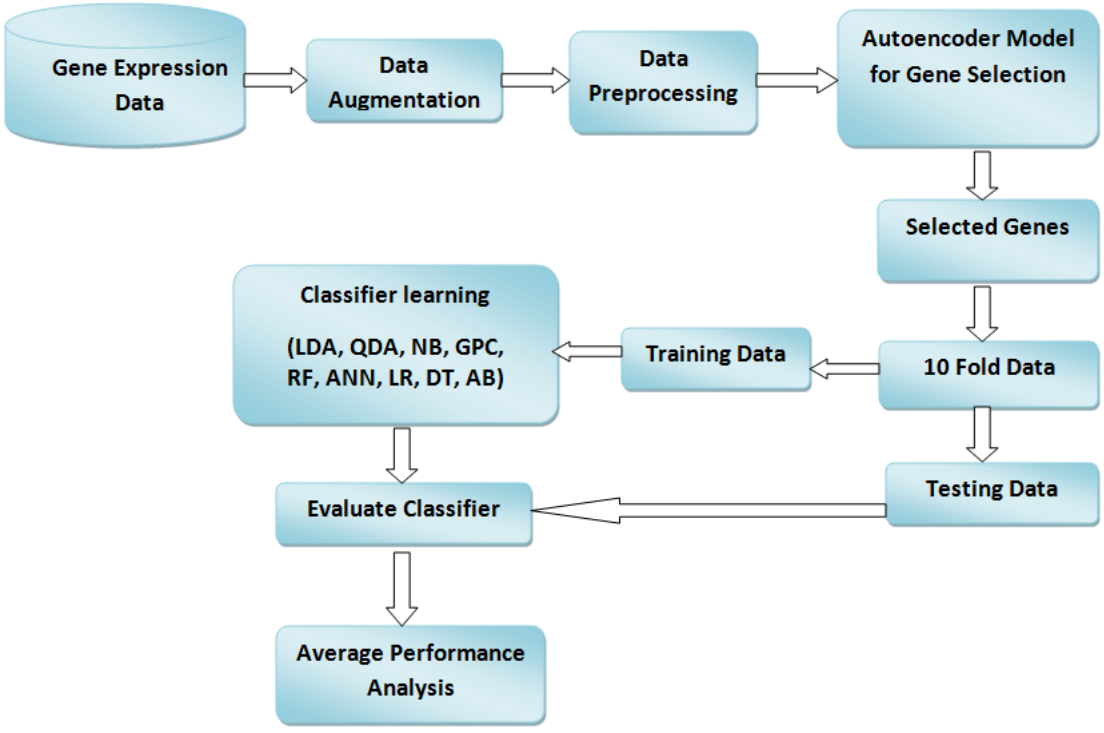


Figure 1: Schematic diagram of the proposed method of gene selection and cancer classification

building and that can lead to bias [35]. As a result, gene-wise normalization is adopted as a standard practice before gene selection and machine learning model building.

Microarray data used in the present work is an N-dimensional array in which each column represents a single gene (feature) and each row represents a single sample. Min–max normalization performs a linear transformation on the original feature (each column, i.e., a single gene) and preserves the relations among the original data values. Therefore, min-max normalization has been used in the majority of research works on gene expression analysis [10], [15], [28], [32]. In the present work also, min-max normalization is used for a fair comparison with other competing approaches. Min-Max Scaling is applied on all the datasets, which is computed using the equation 2.

$$X_{Norm} = \frac{X - X_{min}}{X_{max} - X_{min}} \tag{2}$$

where X_{min} and X_{max} are the minimum and maximum values of a feature in the dataset.

3.3. Autoencoder for gene selection

An autoencoder is an artificial neural network that learns efficient data representation using unsupervised learning. The Autoencoder model is made up of two parts: an encoder and a decoder as shown in Figures 2. The encoder learns a representation of an input sample, and transforms it to a low-dimensional vector called 'latent vector' for feature selection

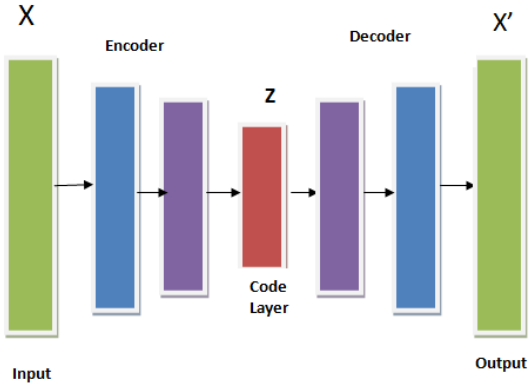


Figure 2: Autoencoder model

task, while the decoder learns to reconstruct the input data from the latent vector [36]. This type of autoencoder, generally called a vanilla autoencoder attempts to replicate the input sample as accurately as possible. The encoder has an input layer, a number of hidden layers, and the output layer, which produces the latent representation of the input. The decoder accepts input from the output layer of the encoder, and has its own hidden layers. The output layer produces a reconstructed sample, and hence the input and output vector

size is the same in the autoencoder. The primary goal of the autoencoder model in deep learning is retrieving relevant attributes to reconstruct its input [37, 38]. The reconstruction loss helps evaluate how closely an output matches its input. The loss function used in the proposed autoencoder is the cross entropy loss function given in equation 3.

$$Loss = \Sigma(I(x)\log(O(x)) + (1 - I(x))(\log(1 - O(x)))) \quad (3)$$

where $I(x)$ is the input data and $O(x)$ is the reconstructed output generated by the decoder.

Optimization of the loss is performed using standard optimization methods such as stochastic gradient descent or its variants like gradient descent with moment, or ADAM [39]. In the present work, two models of autoencoder are chosen for gene selection. The first model is the vanilla autoencoder, whose encoder part consists of four layers, one input layer, two hidden layers and a code layer, and the decoder part consists of two hidden layers and one output layer. Since the gene expression data is very high dimensional in nature, we gradually reduce the size of the data samples. In the encoder architecture, numbers of neurons in each layer is as follows. In the input layer number of neurons are the same as the dimension of the dataset, the first hidden layer contains 1024 neurons, the second hidden layer contains 512 neurons and the size of the code layer is gradually reduced from 200 to 10 taking values from the data $\{200, 100, 64, 32, 10\}$. In the decoder part, the first hidden layer contains 512 neurons, the second hidden layer has 1024 neurons and the number of neurons in output layer is the same as those in the input layer.

The second autoencoder is a stacked autoencoder that is most commonly used in the literature for gene selection. The Stacked Autoencoder (SAE) is trained in a different way than the reconstruction autoencoder [40]. Each layer of the encoder is independently trained to produce its input. Initially, the first hidden layer is trained to reproduce the input vector. Once the training is complete and the first layer is able to reconstruct its input with high accuracy, the second hidden layer is added to the encoder. Input to the second layer is the latent vector of the input sample produced by the first layer. The second layer is trained to perform reconstruction of its input in a similar way. This way any number of layers are added to the encoder, and each layer learns to perform the reconstruction task. In other words, each layer acts as an independent autoencoder, and hence this autoencoder model is termed as a stacked autoencoder. The final hidden layer's output is passed to a supervised learning classifier [41]. The stacked autoencoder model used in the present framework has the same architecture as that of the vanilla autoencoder.

Both the autoencoders are trained to learn the most suitable representative genes from the samples of a dataset. While building an autoencoder gene selection model, the number of neurons in the code layer of the encoder part were gradually reduced from 200 to 10 and for each size of the encoded gene samples, the autoencoders were trained to select the most weighted genes as representatives of the samples. After completing the training of the encoder, the latent vectors from the code layer were considered as the features to be fed to a machine learning classifier. For each latent vector size ranging from 10 to 200, all the nine classifiers were trained and their classification performances were evaluated. Through this empirical study, the autoencoder with 32 neurons in the code layer was finalized as the most suitable model for gene selection, as it gave the highest classification accuracy for all the ten datasets. The latent vector was further analyzed to find out the highest weighted 32 genes using strongly related genes algorithm as disused in Section 5. Details of the training process are given in Section 4.

3.4. Proposed Classification Framework

The proposed Gene Selection and Cancer Classification Framework (GSCCF) consists of two parts, an autoencoder and a classifier. The autoencoder can be either a vanilla autoencoder or a stacked autoencoder that produces a latent vector for each sample. The classification of the gene data is performed using the latent vector produced by autoencoder. The classification model could be chosen from any of the nine standard classifiers used in the present work, namely – RF, LR, LDA, QDA, GPC, SVM, NB, DT, and AB. The classification model labels the gene expression data input to a predefined class. The number of classes for a classifier are defined as per the existing classes in the data. For example, binary class datasets have two classes defined as abnormal (cancerous) and normal (non-cancerous). Similarly, for multiclass datasets, each class defines a subtype of cancer. For example, mixed-lineage leukemia (MLL) dataset has three classes namely, Acute lymphocytic leukemia (ALL), Acute myeloid leukemia (AML), Mixed-lineage leukemia (MLL) which are subtypes of MLL cancer.

The classification techniques used in this study are evaluated for their suitability in the cancer classification task. For a given dataset, if a particular combination of gene selection and classification model exhibits better performance than other combinations, it is chosen for building the gene selection and classification model. It is worthwhile to mention here that the performance of a machine learning algorithm varies from dataset to dataset. Further, datasets have their own limitations, sometimes they are noisy, and are imbalance. All these factors limit the performance of any machine learning algorithm, and therefore nine different classifiers are chosen to assess which one performs uniformly well on all the ten datasets.

In the next section, we present the experimental results with two autoencoder models and nine classifiers.

4. Experimental results and discussion

For building a gene selection and cancer classification model, there are two different components of training and testing (1) for the autoencoder and (2) for the classifier. As explained in Section 3.3, the autoencoder and classifier models are trained in tandem and the most suitable combination of autoencoder and classifier is selected for the model

Table 2
Datasets used in experiment

SNo.	Dataset Name	Genes	Classes	Samples	Class Distribution	Augmented data Samples	Augmented data Distribution
DS1	Colon	2000	2	62	40/22	122	60/62
DS2	Breast Cancer	24481	2	97	51/46	194	97/97
DS3	Central Nervous System	7130	2	60	39/21	155	58/57
DS4	ALL-AML	7130	2	72	25/47	144	72/72
DS5	ALL-AML (Leukemia-3c)	7130	3	72	38/9/35	144	48/48/48
DS6	ALL-AML (Leukemia-4c)	7130	4	72	38/9/21/4	144	43/34/34/33
DS7	MLL	12583	3	72	24/20/28	144	48/48/48
DS8	BreastA	1214	3	98	11/51/35	196	63/66/66
DS9	BreastB	1214	4	49	12/11/7/19	98	24/24/23/27
DS10	LungA	1000	4	197	139/20/17/21	487	139/120/102/126

building. For the gene selection part, two different autoencoder models are proposed. The architecture of autoencoder models is selected based on an empirical experiment that is detailed in subsection 4.3 of this section. For training the autoencoder, cross-entropy loss function is chosen. A batch size of 30 is chosen with a learning rate of 0.001 and ADAM optimizer. Both the autoencoder models are trained for 100 epochs with 80-20 rules, 80 percent data for training and 20 percent for validation. Experiment is also performed with different number of selected genes from 10 to 200 as mentioned earlier.

Once the autoencoders are trained, all the nine classifiers are also trained using the selected genes from the autoencoders and performing 10-fold cross-validation method. The entire gene expression dataset is randomly divided into 10 identical sub-datasets, nine of which are used for training and the final one for testing. The advantage of this strategy is that it is unconcerned with how the samples are partitioned. Each sample appears once in a test set and nine times in a training set. These proposed procedures are separately applied to all of the datasets listed in Table 2, and the results are recorded in the subsection 4.4 of this section.

All the experiments are performed on an Intel Core i7 processor, 5 GHz and 8GB RAM. Python programming with Keras and SciKit library is used for the experiments. Performance analysis of different combinations of gene selection and classification models is done on ten different benchmark gene expression datasets.

4.1. Experimental Datasets

For the experiments, we collected ten different benchmark datasets available at `http://csse.szu.edu.cn/staff/zhuzx/Datasets.html` [42]. We chose these datasets because the majority of them had been used in the relevant literature. These datasets are very diverse from one another. Although most gene expression datasets have only two class labels, we select some multi-class datasets in our experiment for a better understanding of the outcomes and methods' performances. Since the datasets are small in size, data augmentation is also performed on all the datasets by applying Gaussian noise on some randomly selected samples from each class of the datasets, as detailed in Section 3.1. Here it is pertinent to mention that adding noise may cause some change in the data characteristics and hence it is important to know if the nature of the augmented data remains unchanged after augmentation. To establish that both the augmented and actual samples are correlated, we conducted an experiment and computed the Pearson correlation between original samples and augmented data samples. It showed a linear relationship between the two sets of data with the correlation coefficient value = 1. We further conducted another experiment by randomly selecting 10 samples from each class of the dataset before and after data augmentation and computing the Pearson correlation coefficient. The process was repeated 10 times and then the average value of Pearson correlation coefficient was computed. In all the experiments, the Pearson correlation coefficient turned out to be more than 0.7 for all the datasets. Statistical parameters of the experimental datasets are given in Table 2 that include the numbers of genes, classes, samples with class distribution, augmented data samples and augmented data class distribution achieved after the process of data augmentation.

4.2. Performance evaluation measures

In general, the accuracy of classification is one of the frequently used measures for analyzing the performance of classifiers. When the data distribution is unequal, however, minority samples are insufficient, and the classifiers' ability to predict minority classes is not sensitively reflected. Single-class measures such as precision, recall, and F1 score are frequently used to evaluate classifier performance on unbalanced data [43]. These measures are defined as follows.

Accuracy: Accuracy is defined as the ratio of correctly categorised data samples to all data samples, expressed as:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{4}$$

where TP, TN, FP, and FN denote the number of true positive, true negative, false positive, and false-negative samples, respectively.

Table 3
Hyper-parameter for Deep learning models

Hyper-parameter	Model	
	Vanilla Autoencoder	Stacked Autoencoder
Batch size	30	30
Loss function	Cross-entropy	Cross-entropy
Optimizer	Adam	Adam
Learning rate	0.001	0.001
Epochs	100	100
Neurons in code layer	200,100,64,3210,	200,100,64,32,10

Precision: Precision is defined as the ratio of all correct positive predictions to all positive predictions, also known as positive predictive value (PPV).

$$Precision = \frac{TP}{TP + FP} \tag{5}$$

Recall: Recall is the proportion of accurately predicted positive observations to all actual class observations.

$$Recall = \frac{TP}{TP + FN} \tag{6}$$

F1 Score: Harmonic mean of Precision and Recall is called F1 measure.

$$F1score = 2 \times \frac{Precision * Recall}{Precision + Recall} \tag{7}$$

4.3. Experiment for autoencoder architecture

An empirical study is performed to determine three critical hyper parameters of an autoencoder architecture, such as the the size of the code layer, number of hidden layers in the autoencoder, the number of neurons present in each hidden layer. The aim of this experiment is to find out the best possible parameters for the autoencoders and results are provided in Table 3. For this experiment, mainly three combinations of these parameters are used. The first combination is set to 1 hidden layer of 1024 neurons with various code layer size, such as 200, 100, 64, 32, and 10. The second combination is set to 2 hidden layers, one hidden layer of 1024 neurons and second hidden layer of 512 and was evaluated with various code layer size, such as such as $200, 100, 64, 32$, and 10, and the last combination was set to 2 hidden layers, one hidden layer of 1024 neurons and second hidden layer of 256 and is evaluated with the same code layer size as mentioned before for all the ten datasets. For each combination, 6 times experiments were conducted, respectively using all ten datasets on the vanilla autoencoder and the stacked autoencoder models. Although, the experiments with the vanilla autoencoder model with all the combinations give less average accuracy than the stacked autoencoder model on all the datasets, they are comparable with existing gene selection and cancer classification models. The experiment on the staked autoencoder with the combination of 2 hidden layers of 1024 and 512 neurons and with various code layer performs better and gives an average accuracy of more than 95% and the combination of 1 hidden layer of 1024 neurons with various code layer sizes in both the vanilla autoencoder and the stacked autoencoder shows less accuracy. Accordingly, we select the vanilla autoencoder with 2 hidden layers, 1024 neurons and 512 neurons and code size of 200, 100, 64, 32, and 10. Further, the finally selects stacked autoencoder with 2 hidden of 1024 neurons and 512 neurons with various code layer size as 200, 100, 64, 32, and 10. Since the parameters mentioned above yield better results; hence, same setting is used for the remaining experiments. The models' performances are summarized in Table 4 and Figure 3.

4.4. Performance analysis

The aim of this experiment is to demonstrate that the suggested framework works well for cancer classification based on gene expression data. Further, it can be concluded from the experiments that the proposed framework is not dependent on any particular classifier. Depending on the data characteristics, a classifier may be chosen that performs better than others. For the experiments with the proposed framework, nine classifiers are employed as mentioned in section 3 subsection 3.4. These are QDA, LDA, NB, GPC, SVM, LR, DT, AB, and RF. Appendix Table A1 and Table A2 demonstrate the performance of various combinations of gene selection and cancer classification models. One can see that the models with RF and SVM generate higher accuracy, precision, recall, and F1 score as compared to other classifiers across all gene expression datasets. The dimension of the gene expression datasets varies from 1000 to $25,000$ features. For all the datasets, the models are evaluated for multiple gene subsets selection ranging from 10 to 200. Table 4 summarizes the best results among the nine classifiers from table A1 and table A2. The results demonstrate that the vanilla autoencoder and stacked autoencoder models pick the top 32 genes from 2000 genes of DS1 (Colon) dataset, achieving the maximum classification accuracy of 98.40% and 99.23%. For the DS2 (Breast Cancer) dataset out of 24824 genes top 32 genes are taken, improving classification accuracy by 91.81% and 96.97%. DS3 (Center Nervous System), DS4 (ALL-AML Leukemia 2 classes), DS5 (ALL-AML Leukemia 3 classes) and DS5 (ALL-AML Leukemia 4 classes) include 7130 genes. Classification accuracy of DS3 dataset with top 32 gene selected are 95.36%, and 96.52%. Using our models, the best-selected genes for DS4 are 10 and provides maximum classification accuracy of 98.67% and 99.89%. For DS5 and DS6 classification accuracy is of 98.62%, 87.00% by autoencoder model and 100%, 87.76% by Stacked autoencoder when selecting 32 genes.

With a total of 12483 genes in the DS7 (MLL) dataset, 32 best genes are picked, giving 99.33% and 100% maximum classification accuracy. DS8 (BreastA) and DS9 (BreastB) are breast cancer datasets containing $1214 genes$, and the highest classification accuracy is 98.52%, 97.00%, and 96.97%,

Table 4
Summary of the best result

Datasets	Total no of Genes	Accuracy without FS	Vanila AE			SAE		
			Accuracy	Selected genes	Classifier	Accuracy	Selected genes	Classifier
DS1	2000	86.86	98.40	32	RF,SVM	99.23	32	RF,SVM
DS2	24481	66.00	91.81	64	RF,SVM	96.97	32	RF,SVM
DS3	7130	70.00	95.36	64	SVM	96.52	32	SVM
DS4	7130	90.43	98.67	10	LR,SVM	99.89	10	LR,SVM
DS5	7130	94.46	98.62	32	RF	100	32	SVM
DS6	7130	85.57	87.00	64	RF	87.76	32	RF,LR,SVM
DS7	12583	94.40	99.33	32	LR	100	32	RF,LR,SVM
DS8	1214	93.79	98.52	64	RF	96.97	32	RF
DS9	1214	89.50	97.00	64	SVM	99.00	32	RF,SVM
DS10	1000	96.47	99.59	10	RF	99.98	10	RF,SVM

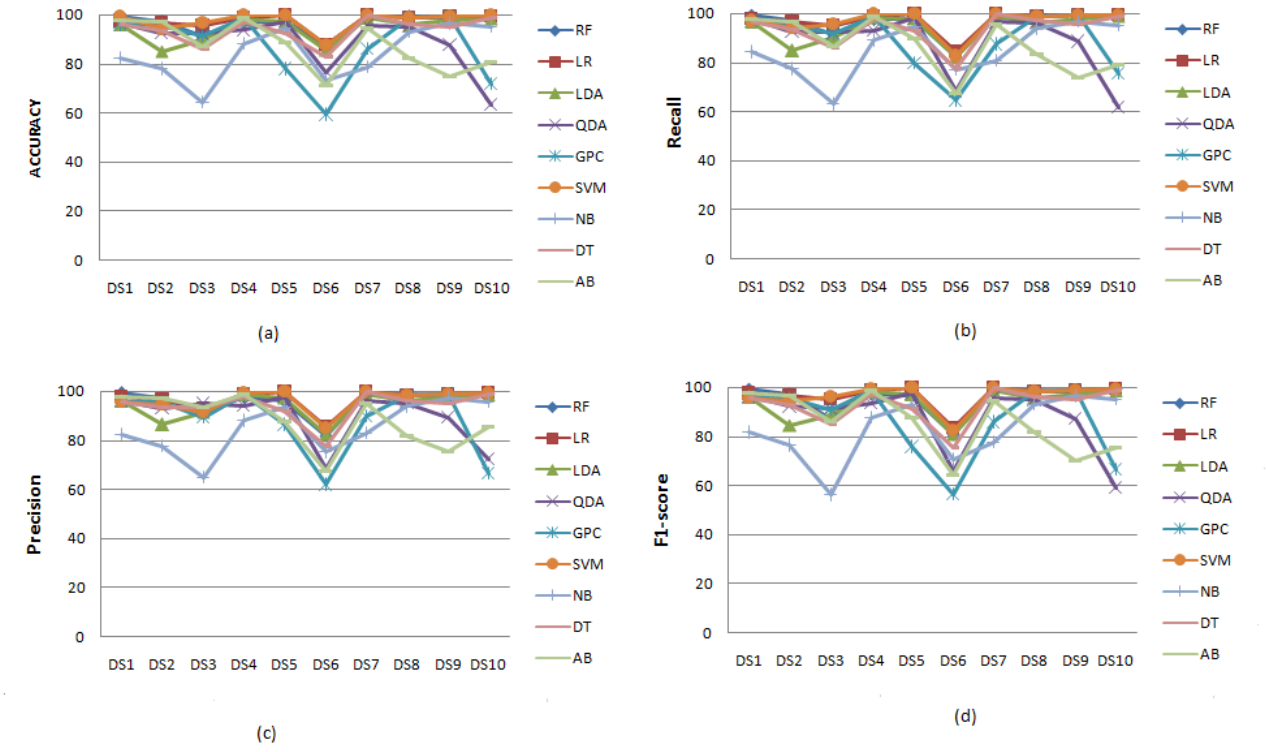


Figure 3: Performance of the proposed framework on nine classifiers (a) Accuracy (b) Recall (c) Precision (d) F1-score

99.00%, respectively, with 32 most relevant genes. In DS10 (LungA) dataset, 10 most significant genes are chosen from 1000 genes with the highest classification accuracy of 99.59% and 99.98%. Figure 3 shows the classification accuracy, precision, recall, and F1-score for all the classifiers. We discover that the RF, SVM and LR classifier deliver a higher classification rate in all data sets. The proposed strategy achieves a maximum classification accuracy of 100% for DS4 and DS7 datasets even if only the top 32 genes are chosen. It is observed that the classification accuracy is not particularly impressive, in the case of DS6 data set.

4.5. Performance Comparison

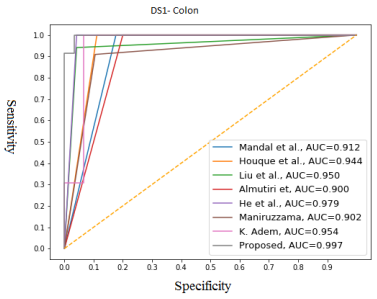
This section is devoted to performance comparison of the proposed method with seven existing gene selection methods. The methods are Mandal et al. [32], Houque et al. [30], Almutiri et al. [28], He et al. [6], Maniruzzama et al. [29], autoencoder based gene selection methods K. Adem [23] and Liu et al. [15]. Table 5 presents the values of average accuracy, recall, precision, and F1-score for ten different datasets for all the methods. The method in Mandal et al. [32] selects 100 best genes using the Minimum Redundancy Maximum Relevance (mRMR) criteria for all the dataset. The method in Houque et al. [30] uses mutual information

Table 5
Performance comparisons of the proposed models with existing models

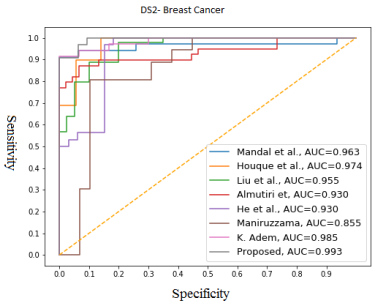
Dataset	Measures	Mandal et al. [32] 2013	Houque et al. [30] 2014	Liu et al. [15] 2017	Almutiri et al.[28]2019	He et al.[6] 2019	Maniruzzama et al.[29] 2019	K. Adem[23] 2020	Proposed SAE
DS1	Accuracy	97.63	98.04	97.57	97.63	96.67	97.63	98.23	**99.23**
	Recall	97.05	98.33	97.50	98.17	96.33	97.75	98.17	**99.17**
	Precision	98.17	98.66	97.09	97.05	97.64	97.84	98.38	**99.38**
	F1-score	97.54	98.37	97.54	97.54	96.41	97.66	98.21	**99.21**
DS2	Accuracy	90.78	95.92	95.41	94.32	91.21	95.53	95.92	**96.97**
	Recall	89.79	95.58	95.42	94.79	91.69	95.07	96.28	**96.90**
	Precision	90.18	95.38	95.51	94.18	91.44	95.19	95.70	**97.08**
	F1-score	90.65	95.45	95.40	93.90	90.95	95.36	95.83	**96.94**
DS3	Accuracy	93.03	94.85	93.04	95.53	94.62	95.53	93.94	**96.52**
	Recall	93.83	94.03	93.00	95.04	95.04	95.00	94.60	**96.43**
	Precision	93.96	95.90	92.54	95.16	95.45	95.70	93.58	**95.67**
	F1-score	92.48	93.98	92.07	95.29	94.35	95.50	93.52	**96.07**
DS4	Accuracy	93.03	97.24	98.62	98.67	93.39	99.33	98.33	**99.89**
	Recall	93.83	97.75	98.82	98.75	93.45	99.50	98.50	**99.22**
	Precision	93.63	97.22	98.57	98.89	93.05	99.17	98.17	**99.69**
	F1-score	93.45	97.11	98.61	98.66	93.52	99.28	98.28	**99.19**
DS5	Accuracy	96.60	97.09	98.31	98.57	97.16	99.29	100	**100**
	Recall	97.36	98.00	98.33	98.86	97.50	99.50	99.05	**100**
	Precision	96.74	97.41	98.58	97.50	98.17	99.17	99.33	**100**
	F1-score	94.74	97.28	98.43	97.75	97.54	99.28	99.33	**100**
DS6	Accuracy	84.46	85.00	84.83	85.47	85.38	86.33	86.00	**87.67**
	Recall	84.54	80.50	78.20	81.00	85.96	82.04	82.04	**86.99**
	Precision	82.28	83.78	79.54	83.56	85.19	81.37	81.35	**86.69**
	F1-score	83.59	80.03	78.47	80.81	85.09	80.18	80.18	**86.43**
DS7	Accuracy	96.57	99.29	99.31	99.33	96.57	98.29	97.09	**100**
	Recall	96.10	98.33	99.26	99.33	96.03	98.67	98.22	**100**
	Precision	95.91	99.52	99.52	99.52	96.15	98.26	98.75	**100**
	F1-score	96.20	98.63	99.36	99.37	95.59	98.75	98.26	**100**
DS8	Accuracy	98.19	98.98	98.57	95.95	98.47	98.00	98.00	**99.47**
	Recall	98.33	98.33	98.79	96.76	98.67	98.49	98.49	**98.67**
	Precision	98.06	98.52	98.79	96.16	98.89	98.89	98.67	**99.49**
	F1-score	97.98	98.63	98.67	96.18	98.66	98.06	98.89	**98.89**
DS9	Accuracy	97.00	96.89	97.00	96.89	97.89	97.89	97.00	**99.00**
	Recall	96.75	96.75	96.00	95.83	97.05	98.33	97.05	**99.07**
	Precision	97.89	97.17	97.73	95.97	98.17	98.12	97.26	**99.49**
	F1-score	96.66	96.67	96.26	95.49	97.54	97.81	97.37	**98.89**
DS10	Accuracy	96.47	98.78	98.97	98.98	98.76	99.39	98.59	**99.98**
	Recall	97.23	99.10	98.13	99.18	98.50	99.33	98.71	**99.91**
	Precision	96.47	98.90	98.02	98.99	99.05	99.33	98.59	**99.21**
	F1-score	96.33	98.90	98.05	99.04	98.64	99.26	98.63	**99.02**

for selecting optimal subset of best genes, number of selected genes vary from 100 to 200 for different datasets. The method in Almutiri et al. [28] uses chi square test for selecting best 100 genes for all the dataset. The method of He et al. [6] uses improved reliefF algorithm to select the best 30 genes in all the gene expression datasets. In the method of Maniruzzama et al. [29] statistical test are used to select the most relevant genes, number of selected genes vary from 100 to 1000 for different datasets. In the method of Liu et al. [15] sparse autoencoder is used to select top 100 genes in all the datasets. In the method of K. Adam [23] a stacked autoencoder is used for selecting the most dominant 100 genes in all the dataset. Two stage of evaluation are carried out in the proposed method for gene selection and classification. The best gene data categorized in the range of 10 to 200 is chosen in the first stage. The classification accuracy is evaluated in the second stage utilizing the the best-selected genes for the gene expression datasets. As shown in Table 5, the proposed gene selection and cancer classification model gives high precision with with a minimal selection of genes in all gene expression datasets.

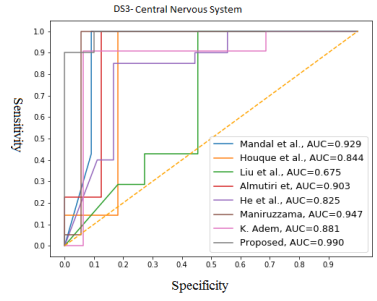
The sensitivity and specificity of classification techniques are frequently used to assess their effectiveness. The sensitivity of a test refers to the percentage of patients who test positive for illness. Specificity, on the other hand, relates to the percentage of healthy patients that test negative. Receiver Operating Characteristics (ROC) curves have been widely utilised in bioinformatics to analyse the performance of classifiers in terms of sensitivity versus specificity. The area under the ROC curve (AUC), which is a significant performance parameter in medical diagnostics, is also computed. The Sensitivity is shown on the Y axis against the Specificity on the X axis in a ROC curve. An ROC curve closer to the upper left hand corner, implies that the overall performance of the classifier is better. The area under the curve (AUC) is used to determine how well a classifier works by measuring how near a ROC curve is to the top left hand corner. The ROC curve for a perfect classifier moves from point (0, 0) to point (0, 1), then from (0, 1) to point



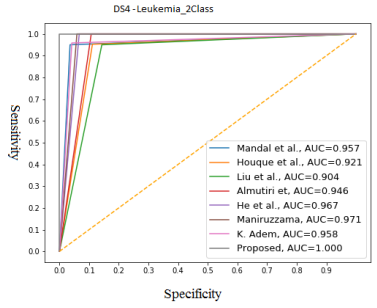
(a) ROC curves of the DS1-Colon Cancer



(b) ROC curves of the DS2-Breast Cancer



(c) ROC curve of the DS3-Central Nervous System



(d) ROC curve of the DS4-Leukemia

Figure 4: Comparision of ROC curves of the proposed method with seven existing methods.

(1, 1), with an AUC of 1.0 [44]. Figure 4 shows the ROC curves with AUC scores obtained by the proposed model and seven existing gene selection methods with four binary class datasets.

5. Biological Significance

An autoencoder model is expected to collect the most relevant information about cancer from gene samples and translate it into a lower-dimensional space called "latent space". The latent space form of the gene data is used to classify the dataset into different labels, and identify biomarkers in cancer. The latent vector may play a useful role in revealing highly interacting genes to better understand cancers. In this work, a stacked autoencoder (SAE) model is proposed for extracting relevant genes from gene expression datasets. Further, the 'Strongly Related Genes' technique introduced in [45] is applied here for locating highly weighted genes in the latent space using Algorithm 1. The algorithm identifies significant genes using the top weighted genes. To show that the stacked autoencoder model selects the most useful genes by giving them higher weightage over the other genes, it is important to find their biological significance. If these highly weighted genes obtained through the latent vector of the autoencoder are useful as cancer bio-markers, it can be established that the results are biologically significant. Therefore, we first apply Algorithm 1, on the latent vectors obtained from a trained stacked autoencoder.Suppose the weight matrix for a single layer model is $W = G \times 32$, where G is the number of genes in the input layer. An $n-$ layer neural network's weight matrix may be computed as follows.

$$W = \prod_{i=1}^{n} W_i$$

Algorithm 1 An algorithm to select strongly related genes

Require: weight matrix ($genes \times weights$)
Ensure: $top\_genes \leftarrow$ Sorted list of top Highly Weighted Genes
  for $G \leftarrow 0$ to Maxgene **do**
    $total\_weight \leftarrow 0$
    $weighted\_gene\_list$;
    for $node \leftarrow 0$ to totalnode **do**
      $total\_weight \leftarrow total\_weight + weight(node)$
    end for
    Append $weighted\_gene\_list \leftarrow (G \times total\_weight)$
  end for
  Sort $weighted\_gene\_list$ in descending order
  $top\_gene \leftarrow weighted\_gene\_list$

Algorithm 1 gives the weight sorting technique 'Strongly related genes'. Finally, the top 32 highly weighted genes are subjected to the Gene Ontology (GO) enrichment analysis using a standard GO enrichment analysis tools. These tools uses an agglomeration algorithm to condense a list of genes or associated biological terms into clusters of related genes and determine their biological relevance. The functional enrichment of a set of genes is based on three different

Table 6
Enriched GO terms associated with Strongly Related Genes in breast cancer.

Go Biologically Process	Total	Observed	Expected	Enrichment	P-value
negative regulation of mitotic sister chromatid separation (GO:2000816)	30	3	0.03	98.07	3.66E-02
negative regulation of mitotic sister chromatid segregation (GO:0033048)	30	3	0.03	98.07	3.66E-02
negative regulation of sister chromatid segregation (GO:0033046)	30	3	0 .03	98.07	3.66E-02
negative regulation of chromosome segregation (GO:0051985)	32	3	0 .03	91.94	4.44E-02
negative regulation of chromosome separation (GO:1905819)	32	3	0 .03	91.94	4.44E-02
mitotic cell cycle phase transition (GO:0044772)	169	5	0 .17	29.02	6.16E-03
cell cycle phase transition (GO:0044770)	177	5	0.18	27.70	7.73E-03
mitotic cell cycle process (GO:1903047)	509	8	0 .52	15.41	1.93E-04
cell division (GO:0051301)	499	7	0 .51	13.76	3.84E-03
mitotic cell cycle (GO:0000278)	599	8	0 .61	13.10	6.74E-04
cell cycle process (GO:0022402)	802	9	0 .82	11.01	3.59E-04
regulation of cell cycle process (GO:0010564)	628	7	0 .64	10.93	1.77E-02
regulation of cell cycle (GO:0051726)	1001	11	1.02	10.78	7.27E-06
positive regulation of cell population proliferation (GO:0008284)	935	8	0 .95	8.39	1.96E-02
cell cycle (GO:0007049)	1205	9	1.23	7.32	1.12E-02
regulation of cell population proliferation (GO:0042127)	1650	12	2.40	4.99	1.39E-02

factors namely, molecular function, biological process, and cellular component. The degree of functional enrichment is determined using the cumulative hypergeometric distribution. The functional enrichment of strongly related genes was investigated in the present paper using the GO term analysis and PANTHER pathway analysis [46]. PANTHER pathway information is used alongwith the Go term analysis to analyze gene expression experimental data and their biological significance.

GO term analysis was performed on the top 32 genes to determine their biological significance. Biologically relevant genes obtained by the proposed strongly related genes algorithm were tested for all the ten gene expression datasets. To present the result analysis, the Breast Cancer dataset was used as a case study. The significantly enriched GO terms in the category "biological process" with a Bonferroni-corrected p-value less than $1E-06$ are shown in Table 6. Many of the most important terms have to do with mitotic sister chromatid segregation, separation and chromosomal segregation and separation, both of which are well-known biological processes in metastasis. Cell cycle and mitotic cell cycle-related processes are two more GO terms that are well-known biological processes relevant with cell proliferation. Other key signaling pathways, such as cell division and cell population proliferation regulation, are also identified. The informative genes for Breast cancer are listed in Table 7. The genes identified by the proposed model with different transcriptional characteristics, are related to several gene co-expression modules and significant cancer genes. As cancer progresses, many genetic events activate dominant-acting oncogenes and damage the activity of specific tumour suppressor genes. Using the GO term analysis and pathway analysis, it is established that the proposed model is able to successfully select those genes that are frequently mutated in breast cancer. These are - ANXA1, TRPS1, TBX3, PCSK1, ATP6AP1, CYP2B7, AGT, FOXA1, SCGB2A2, BIRC5 [47, 48, 49, 50, 51] and tumour suppressor genes as NMT1, LRRC29, CAV1, POLL, SLC22A18 [52, 53].

6. Conclusion

Gene selection is essential when dealing with high-dimensional gene expression data. There are many techniques for gene selection, based on machine learning algorithms. Recently deep learning algorithms have proven to be quite efficient in high-dimensional gene expression data. In the last few decades, researchers have proposed numerous gene selection approaches utilizing deep learning methods that aid in the identification of relevant biomarkers. Here we present a new deep learning framework for gene selection using a stacked autoencoder for cancer classification. The main aim of our work is to gauge the impact of stacked autoencoder gene selection on the final cancer classification accuracy. For the experimental studies, we have used ten different benchmark microarray gene expression datasets. The suggested model using the framework employs a stacked auto-encoder to choose the most helpful gene set, after which nine different classifiers RF, LR, LDA, QDA, GPC, SVM, NB, DT, AB are applied on the selected genes to obtain the most suitable classifier for the problem. The experimental findings reveal that the model accuracy ranging from 90% to 100% is attained on ten different datasets. A comparison with seven existing techniques is also made, and the proposed method is shown to outperform in the majority of cases.

Acknowledgement. The first author acknowledges the research support provided by the Quality Improvement Program of All India Council of Technical Education, Govt. of India.

References

[1] S. M. Alladi, P. Shinde Santosh, V. Ravi, U. S. Murthy, Colon cancer prediction with genetic profiles using intelligent techniques, Bioinformation 3 (3) (2008) 130.

Table 7
Informative genes for Breast cancer

Affymetrix	Gene Symbol and Name	Description	Reference
NM_004034	ANXA1- annexin A1	ANXA1 is a potential marker of development of breast cancer.	[47]
AF043324	DNMT1- N-myristoyltransferase 1	NMT1 is a Tumor suppressor genes related to DNA replication	[52]
NM_021136	TRPS1- transcriptional repressor GATA binding 1	TRPS1 is a specific gene for breast carcinoma	[48]
U47671	TBX3- TBX3 antisense RNA 1	TBX3 is a promising diagnostic marker for breast cancer	[49]
M33318	PARP6- cytochrome P450 family 2 subfamily A member 6	PARP6 directly inhibits cell proliferation and induction of apoptosis in breast cancer cells.	[52]
NM_000439	PCSK1- proprotein convertase subtilisin/kexin type 1	PCSK1 is highly expressed in breast cancers.	[52]
NM_001183	ATP6AP1-ATPase H+ transporting accessory protein 1	ATP6AP1 expression promote carcinogenesis wchich is associated with breast cancer	[54]
AL133109	ARPP21-cAMP regulated phosphoprotein 21	ARPP21is a Protein Coding gene	[52]
AF176701	LRRC29-leucine rich repeat containing 29	LRRC29 is tumour suppressor genes in lobular breast cancer	[53]
NM_014770	AGAP2-ArfGAP with GTPase domain, ankyrin repeatand PH domain 2	AGAP2 mediates anti-apoptotic effects of cell growth factor.	[52]
NM_002424	MMP8-matrix metallopeptidase 8	MMP8 inhibits cancer cell invasion and proliferation.	[55]
M33318	CYP2A6-cytochrome P450 family 2 subfamily A member 6	CYP2A6 catalyze many reactions involved in synthesis of cholesterol, steroids and other lipids	[56]
NM_001759	CCNA2-cyclin A2	CCNA2 function as regulators of the cell cycle.	[57]
NM_004780	PTTG1-pituitary tumor-transforming 1	PTTG is an oncogene for pituitary tumors	[58]
NM_000611	CD 59-CD59 molecule	CD59 role in signal transduction pathways in the activation of T cells	[52]
NM_006835	CDKN3-cyclin dependent kinase inhibitor 3	CDKN3 identified as a cyclin-dependent kinase inhibitor.	[52]
AB033105	KIF11-family member kinesin family member 11	KIF11 perform various kinds of spindle dynamics during cell mitosis.	[52].
NM_013409	PRC1-protein regulator of cytokinesis 1	PRC1 substrate of several cyclin-dependent kinases during cell mitosis.	[52]
NM_001759	CCND2-cyclin D2	CCND2 functions in the regulation of CDK kinases in the cell cycle.	[52]
NM_007065	CDC37-cell division cycle 37	CdC 37 a cell division cycle control protein	[52]
NM_006622	PLK2-polo like kinase 2	PLK2 play an important role in cells undergoing rapid cell division	[52]
NM_003118	SPARC-secreted protein acidic and cysteine rich	SPARC correlated with metastasis based on changes to cell shape which can promote tumor cell invasion.	[52]
M29873	CYP2B7-cytochrome P450 family 2 subfamily B , member 7 pseudogene	CYP2B7 is a Estrogen receptor, related to the progression of breast cancer.	[50]
NM_000029	AGTR2-angiotensinogen	AGT is a Estrogen receptor, related to the progression of breast cancer.	[50]
L20688	ARHGDIB-Rho GDP dissociation inhibitor beta	Involved in diverse cellular events, including cell signaling, proliferation.	[52]
NM_001753	CAV1-caveolin 1	CAV1 a tumor suppressor gene and related to negative regulator of the Ras-p42/44 mitogen-activated kinase cascade.	[52]
NM_007105	SLC22A18-solute carrier family 22 member 18	SLC22A18 AS an important tumor-suppressor gene in breast cancer	[52]
NM_004496	FOXA1-forkhead box A1	FOXA1 expression correlates with estrogen receptor, related to the progression of breast cancer	[51]
NM_013274	POLL-DNA polymerase lambda	POLL is a Tumor suppressor genes realted to DNA repair processes	[52]
NM_002411	SCGB2A2-secretoglobin family 2A member 2	SCGB2A2 has been found over expressed in breast tumors	[59]
NM_002733	TRIP13-thyroid hormone receptor interactor 13	TRIP13 regulating mitotic processes.	[60]
NM_005727	BIRC5-baculoviral IAP repeat containing 5	BIRC5 is higher expressed in different breast cancer subtypes	[61]

[2] U. Alon, N. Barkai, D. A. Notterman, K. Gish, S. Ybarra, D. Mack, A. J. Levine, Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays, Proceedings of the National Academy of Sciences 96 (12) (1999) 6745–6750.

[3] M. Schena, D. Shalon, R. W. Davis, P. O. Brown, Quantitative monitoring of gene expression patterns with a complementary dna microarray, Science 270 (5235) (1995) 467–470.

[4] H. M. Zawbaa, E. Emary, C. Grosan, V. Snasel, Large-dimensionality small-instance set feature selection: a hybrid bio-inspired heuristic approach, Swarm and Evolutionary Computation 42 (2018) 29–42.

[5] P. Chaudhari, H. Agarwal, V. Bhateja, Data augmentation for cancer classification in oncogenomics: an improved knn based approach, Evolutionary Intelligence (2019) 1–10.

[6] Y. He, J. Zhou, Y. Lin, T. Zhu, A class imbalance-aware relief algorithm for the classification of tumors using microarray gene expression data, Computational biology and chemistry 80 (2019) 121–127.

[7] S. H. Shah, M. J. Iqbal, I. Ahmad, S. Khan, J. J. Rodrigues, Optimized gene selection and classification of cancer from microarray gene expression data using deep learning, Neural Computing and Applications (2020) 1–12.

[8] Y.-C. Chen, W.-C. Ke, H.-W. Chiu, Risk classification of cancer survival using ann with gene expression data from multiple laboratories, Computers in biology and medicine 48 (2014) 1–7.

[9] R. R. Bhat, V. Viswanath, X. Li, Deepcancer: detecting cancer through gene expressions via deep generative learning, arXiv preprint arXiv:1612.03211 (2016).

[10] A. K. Dwivedi, Artificial neural network model for effective cancer classification using microarray gene expression data, Neural Computing and Applications 29 (12) (2018) 1545–1554.

[11] J. Tan, M. Ung, C. Cheng, C. S. Greene, Unsupervised feature construction and knowledge extraction from genome-wide assays of breast cancer with denoising autoencoders, in: Pacific symposium on biocomputing co-chairs, World Scientific, 2014, pp. 132–143.

[12] L. Macías-García, J. M. Luna-Romera, J. García-Gutiérrez, M. Martínez-Ballesteros, J. C. Riquelme-Santos, R. González-Cámpora, A study of the suitability of autoencoders for preprocessing data in breast cancer experimentation, Journal of biomedical informatics 72 (2017) 33–44.

[13] R. Fakoor, F. Ladhak, A. Nazi, M. Huber, Using deep learning to enhance cancer diagnosis and classification, in: Proceedings of the international conference on machine learning, Vol. 28, ACM, New York, USA, 2013, pp. 3937–3949.

[14] K. Chaudhary, O. B. Poirion, L. Lu, L. X. Garmire, Deep learning–based multi-omics integration robustly predicts survival in liver cancer, Clinical Cancer Research 24 (6) (2018) 1248–1259.

[15] J. Liu, X. Wang, Y. Cheng, L. Zhang, Tumor gene expression data classification via sample expansion-based deep learning, Oncotarget 8 (65) (2017) 109646.

[16] G. P. Way, C. S. Greene, Extracting a biologically relevant latent space from cancer transcriptomes with variational autoencoders, in: PACIFIC SYMPOSIUM ON BIOCOMPUTING 2018: Proceedings of the Pacific Symposium, World Scientific, 2018, pp. 80–91.

[17] P. Danaee, R. Ghaeini, D. A. Hendrix, A deep learning approach for cancer detection and relevant gene identification, in: Pacific symposium on biocomputing 2017, World Scientific, 2017, pp. 219–229.

[18] D. Zhang, L. Zou, X. Zhou, F. He, Integrating feature selection and feature extraction methods with deep learning to predict clinical outcome of breast cancer, IEEE Access 6 (2018) 28936–28944.

[19] G. E. Hinton, S. Osindero, Y.-W. Teh, A fast learning algorithm for deep belief nets, Neural computation 18 (7) (2006) 1527–1554.

[20] Y. Bengio, P. Lamblin, D. Popovici, H. Larochelle, Greedy layer-wise training of deep networks, in: Advances in neural information processing systems, 2007, pp. 153–160.

[21] M. Ranzato, C. Poultney, S. Chopra, Y. LeCun, et al., Efficient learning of sparse representations with an energy-based model, Advances in neural information processing systems 19 (2007) 1137.

[22] R. Vargas, A. Mosavi, R. Ruiz, Deep learning: a review (2017).

[23] K. Adem, Diagnosis of breast cancer with stacked autoencoder and subspace knn, Physica A: Statistical Mechanics and its Applications 551 (2020) 124591.

[24] L. Rangarajan, et al., Bi-level dimensionality reduction methods using feature selection and feature extraction, International Journal of Computer Applications 4 (2) (2010) 33–38.

[25] K. Kira, L. A. Rendell, et al., The feature selection problem: Traditional methods and a new algorithm, in: Aaai, Vol. 2, 1992, pp. 129–134.

[26] C. Lazar, J. Taminau, S. Meganck, D. Steenhoff, A. Coletta, C. Molter, V. de Schaetzen, R. Duque, H. Bersini, A. Nowe, A survey on filter techniques for feature selection in gene expression microarray analysis, IEEE/ACM transactions on computational biology and bioinformatics 9 (4) (2012) 1106–1119.

[27] J. C. Rajapakse, P. A. Mundra, Multiclass gene selection using pareto-fronts, IEEE/ACM transactions on computational biology and bioinformatics 10 (1) (2013) 87–97.

[28] T. Almutiri, F. Saeed, Chi square and support vector machine with recursive feature elimination for gene expression data classification, in: 2019 First International Conference of Intelligent Computing and Engineering (ICOICE), IEEE, 2019, pp. 1–6.

[29] M. Maniruzzaman, M. J. Rahman, B. Ahammed, M. M. Abedin, H. S. Suri, M. Biswas, A. El-Baz, P. Bangeas, G. Tsoulfas, J. S. Suri, Statistical characterization and classification of colon microarray gene expression data using multiple machine learning paradigms, Computer methods and programs in biomedicine 176 (2019) 173–193.

[30] N. Hoque, D. K. Bhattacharyya, J. K. Kalita, Mifs-nd: A mutual information-based feature selection method, Expert Systems with Applications 41 (14) (2014) 6371–6385.

[31] M. J. Rani, D. Devaraj, Two-stage hybrid gene selection using mutual information and genetic algorithm for cancer data classification, Journal of medical systems 43 (8) (2019) 1–11.

[32] M. Mandal, A. Mukhopadhyay, An improved minimum redundancy maximum relevance approach for feature selection in gene expression data, Procedia Technology 10 (2013) 20–27.

[33] F. Al-Obeidat, A. Tubaishat, B. Shah, Z. Halim, et al., Gene encoder: a feature selection technique through unsupervised deep learning-based clustering for large gene expression data, Neural Computing and Applications (2020) 1–23.

[34] I. Kononenko, Estimating attributes: Analysis and extensions of relief, in: European conference on machine learning, Springer, 1994, pp. 171–182.

[35] T. Muhammad, Z. Halim, Employing artificial neural networks for constructing metadata-based model to automatically select an appropriate data visualization technique, Applied Soft Computing 49 (2016) 365–384.

[36] P. Vincent, H. Larochelle, Y. Bengio, P.-A. Manzagol, Extracting and composing robust features with denoising autoencoders, in: Proceedings of the 25th international conference on Machine learning, 2008, pp. 1096–1103.

[37] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, P.-A. Manzagol, L. Bottou, Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion., Journal of machine learning research 11 (12) (2010).

[38] Y. Bengio, Practical recommendations for gradient-based training of deep architectures, in: Neural networks: Tricks of the trade, Springer, 2012, pp. 437–478.

[39] C. Zhang, Q. Liao, A. Rakhlin, B. Miranda, N. Golowich, T. Poggio, Theory of deep learning iib: Optimization properties of sgd, arXiv preprint arXiv:1801.02254 (2018).

[40] Y. Guo, Y. Liu, A. Oerlemans, S. Lao, S. Wu, M. S. Lew, Deep learning for visual understanding: A review, Neurocomputing 187 (2016) 27–48.

[41] O. Kaynar, A. G. Yüksek, Y. Görmez, Y. E. Işik, Intrusion detection with autoencoder based deep learning machine, in: 2017 25th Signal Processing and Communications Applications Conference (SIU), IEEE, 2017, pp. 1–4.

[42] I. Jain, V. K. Jain, R. Jain, Correlation feature selection based improved-binary particle swarm optimization for gene selection and cancer classification, Applied Soft Computing 62 (2018) 203–215.

[43] H. He, E. A. Garcia, Learning from imbalanced data, IEEE Transactions on knowledge and data engineering 21 (9) (2009) 1263–1284.

[44] K. Hajian-Tilaki, Receiver operating characteristic (roc) curve analysis for medical diagnostic test evaluation, Caspian journal of internal medicine 4 (2) (2013) 627.

[45] R. K. Mondol, N. D. Truong, M. Reza, S. Ippolito, E. Ebrahimie, O. Kavehei, Afexnet: An adversarial autoencoder for differentiating breast cancer sub-types and extracting biologically relevant genes, IEEE/ACM Transactions on Computational Biology and Bioinformatics (2021).

[46] The gene ontology resource, http://geneontology.org/, accessed: 2021-10-18 (1999).

[47] Y. Cao, Y. Li, M. Edelweiss, B. Arun, D. Rosen, E. Resetkova, Y. Wu, J. Liu, A. Sahin, C. T. Albarracin, Loss of annexin a1 expression in breast cancer progression, Applied Immunohistochemistry & Molecular Morphology 16 (6) (2008) 530–534.

[48] D. Ai, J. Yao, F. Yang, L. Huo, H. Chen, W. Lu, L. M. S. Soto, M. Jiang, M. G. Raso, S. Wang, et al., Trps1: a highly sensitive and specific marker for breast carcinoma, especially for triple-negative breast cancer, Modern Pathology 34 (4) (2021) 710–719.

[49] W. Yarosh, T. Barrientos, T. Esmailpour, L. Lin, P. M. Carpenter, K. Osann, H. Anton-Culver, T. Huang, Tbx3 is overexpressed in breast cancer and represses p14arf by interacting with histone deacetylases, Cancer research 68 (3) (2008) 693–699.

[50] R. Lo, L. Burgoon, L. MacPherson, S. Ahmed, J. Matthews, Estrogen receptor-dependent regulation of cyp2b6 in human breast cancer cells, Biochimica et Biophysica Acta (BBA)-Gene Regulatory Mechanisms 1799 (5-6) (2010) 469–479.

[51] H. Nakshatri, S. Badve, Foxa1 in breast cancer, Expert reviews in molecular medicine 11 (2009).

[52] Genecards: The human gene database, https://www.genecards.org/, accessed: 2021-10-18 (2016).

[53] T. van Wezel, M. Lombaerts, E. H. van Roon, K. Philippo, H. J. Baelde, K. Szuhai, C. J. Cornelisse, A.-M. Cleton-Jansen, Expression analysis of candidate breast tumour suppressor genes on chromosome 16q, Breast Cancer Research 7 (6) (2005) 1–7.

[54] J. Wang, Y. Liu, S. Zhang, Prognostic and immunological value of atp6ap1 in breast cancer: implications for sars-cov-2, Aging (Albany NY) 13 (13) (2021) 16904.

[55] K. Juurikka, G. S. Butler, T. Salo, P. Nyberg, P. Åström, The role of mmp8 in cancer: a systematic review, International journal of molecular sciences 20 (18) (2019) 4506.

[56] Z. Desta, Y. Kreutz, A. Nguyen, L. Li, T. Skaar, L. Kamdem, N. Henry, D. Hayes, A. Storniolo, V. Stearns, et al., Plasma letrozole concentrations in postmenopausal women with breast cancer are associated with cyp2a6 genetic variants, body mass index, and age, Clinical Pharmacology & Therapeutics 90 (5) (2011) 693–700.

[57] T. Gao, Y. Han, L. Yu, S. Ao, Z. Li, J. Ji, Ccna2 is a prognostic biomarker for er+ breast cancer and tamoxifen resistance, PloS one 9 (3) (2014) e91771.

[58] R. Yu, S. Melmed, et al., Pituitary tumor transforming gene: an update, Frontiers of Hormone Research 32 (2004) 175–185.

[59] X.-f. Guan, M. K. Hamedani, A. Adeyinka, C. Walker, A. Kemp, L. C. Murphy, P. H. Watson, E. Leygue, Relationship between mammaglobin expression and estrogen receptor status in breast tumors, Endocrine 21 (3) (2003) 245–250.

[60] S. Lu, J. Qian, M. Guo, C. Gu, Y. Yang, Insights into a crucial role of trip13 in human cancer, Computational and structural biotechnology journal 17 (2019) 854–861.

[61] J.-b. Dai, B. Zhu, W.-j. Lin, H.-y. Gao, H. Dai, L. Zheng, W.-h. Shi, W.-x. Chen, Identification of prognostic significance of birc5 in breast cancer using integrative bioinformatics analysis, Bioscience reports 40 (2) (2020) BSR20193678.

Table A1
Performance of Autoencoder model for gene selection

Dataset	Measure	RF	LR	LDA	QDA	GPC	SVM	NB	DT	AB
DS1-Colon	Accuracy	87.38	87.62	84.29	68.81	87.62	88.81	84.05	75.45	85.48
	Precision	87.38	87.62	84.29	68.81	87.62	88.81	84.05	75.45	85.48
	Recall	87.38	87.62	84.29	68.81	87.62	88.81	84.05	75.45	85.48
	F1-score	87.38	87.62	84.29	68.81	87.62	88.81	84.05	75.45	85.48
DS2-Breast Cancer	Accuracy	91.81	91.14	90.43	87.90	91.81	89.14	75.48	87.14	88.52
	Precision	91.17	91.47	90.96	88.02	92.14	89.80	75.76	85.73	87.99
	Recall	90.16	91.56	90.67	87.23	92.16	89.18	74.83	86.76	88.43
	F1-Score	90.18	91.05	90.02	86.75	91.72	88.78	74.60	85.77	87.65
DS3-Central Nervous System	Accuracy	89.39	90.23	85.91	92.12	93.86	95.36	80.68	88.03	90.3
	Precision	90.32	90.96	87.2	94.43	93.91	97.41	81.48	88.5	91.64
	Recall	89.57	91.22	87.74	93.17	93.69	95.75	81.26	88.49	90.68
	F1-Score	88.65	89.62	85.03	91.64	93.5	96.05	79.48	87.46	89.5
DS4-ALL-AML-Leukemia	Accuracy	97.24	98.67	98	93.76	98.57	98	88.19	97.29	98.67
	Precision	97.43	98.89	98.5	94.16	98.54	99.00	87.92	97.5	98.89
	Recall	97.48	98.75	98.12	93.14	98.73	99.00	89.07	97.62	98.75
	F1-Score	97.19	98.66	97.96	93.35	98.53	99.00	87.81	97.15	98.66
DS5-ALL-AML-Leukemia-3C	Accuracy	98.62	97.95	93.86	88.81	79.24	97.33	95.95	95.90	90.57
	Precision	98.78	98.38	95.25	90.20	81.65	98.10	96.68	96.87	92.11
	Recall	97.86	97.92	94.33	88.50	86.99	97.93	96.00	96.56	90.32
	F1-Score	97.96	97.94	93.54	87.27	78.04	97.64	95.80	96.33	89.24
DS6-ALL-AML-Leukemia-4C	Accuracy	87	84.95	85.57	66.19	59.29	84.24	86.29	82.14	62.86
	Precision	83.29	80.66	82.81	68.00	64.61	83.04	84.83	80.06	51.04
	Recall	84.18	76.74	83.15	65.77	62.00	79.26	83.72	80.03	37.42
	F1-Score	82.07	77.19	80.87	62.85	56.33	78.63	81.72	78.27	41.98
DS7-MLL	Accuracy	98	99.33	93.05	94.48	86.19	98.67	92.48	96.57	97.29
	Precision	98.47	99.58	93.59	95.98	89.74	99.58	93.11	97.33	98
	Recall	98.38	99.33	94.22	95.86	87.36	99.33	92.13	96.05	97.71
	F1-Score	96.82	84.01	82.14	60.79	88.66	95.34	63.87	93.9	88.03
DS8-BreastA	Accuracy	98.5	97.47	95.39	97.5	96.95	97.97	94.45	96.42	96.97
	Precision	98.19	97.15	95.3	98.15	96.68	97.82	94.74	96.08	96.6
	Recall	98.93	98.04	95.84	98.72	97.48	98.46	95.74	97.07	97.71
	F1-Score	98.33	97.28	95.1	97.92	96.72	97.88	94.43	96.14	96.84
DS9-BreastB	Accuracy	91.89	93.89	89.56	81.33	89.56	92.89	97	88.56	60.89
	Precision	91.01	93.26	88.29	80.56	90.26	91.25	97.56	87.89	52.95
	Recall	91.25	92.92	88.12	82.08	89.58	90.42	97.5	89.17	58.75
	F1-Score	89.66	91.97	86.39	76.98	87.78	90.02	97.46	86.1	51.77
DS10-LungA	Accuracy	99.59	99.18	98.16	99.18	83.58	99.18	90.34	98.78	66.34
	Precision	99.47	99.18	98.17	99.28	89.87	99.16	90.35	98.69	58.85
	Recall	99.71	99.41	98.53	99.16	85.92	99.39	90.79	99	67.47
	F1-Score	99.57	99.23	98.25	99.18	83.43	99.23	89.58	98.77	59.36

Table A2
Performance of Stacked Autoencoder model for gene selection

Dataset	Measure	RF	LR	LDA	QDA	GPC	SVM	NB	DT	AB
DS1-Colon	Accuracy	99.23	97.56	95.90	96.67	95.96	99.23	82.24	95.90	97.5
	Precision	99.38	97.83	96.40	97.64	96.84	97.01	84.27	96.41	97.92
	Recall	99.17	97.62	95.95	96.33	95.67	95.50	82.45	95.79	97.62
	F1-score	99.21	97.53	95.86	96.41	95.81	95.62	81.67	95.78	97.45
DS2-Breast Cancer	Accuracy	96.97	96.42	85.05	92.29	95.42	94.34	77.92	93.26	96.92
	Precision	97.08	96.67	86.29	92.84	95.78	94.75	77.61	93.59	97.31
	Recall	96.90	96.46	84.79	92.42	95.25	94.09	77.38	93.47	96.84
	F1-score	96.94	96.39	84.56	92.13	95.33	94.23	76.21	93.01	96.81
DS3-Central Nervous System	Accuracy	91.29	95.61	89.47	92.12	91.29	96.52	64.39	86.06	86.89
	Precision	91.08	93.15	91.47	94.79	89.19	91.43	64.92	93.26	93.17
	Recall	91.57	95.04	90.15	92.12	91.71	95.67	63.02	86.19	86.49
	F1-score	86.96	95.02	88.72	91.38	90.75	96.07	56.12	85.09	85.94
DS4-ALL-AML-Leukemia	Accuracy	97.24	98.67	98.00	93.76	98.57	99.89	88.19	97.29	98.67
	Precision	97.43	98.89	98.5	94.16	98.54	99.22	87.92	97.5	98.89
	Recall	97.48	98.75	98.12	93.14	98.73	99.69	89.07	97.62	98.75
	F1-score	97.19	98.66	97.96	93.35	98.53	99.19	87.81	97.15	98.66
DS5-ALL-AML-Leukemia-3C	Accuracy	97.24	100	97.29	97.19	77.81	100	94.38	92.38	88.76
	Precision	97.55	100	97.83	97.94	79.71	100	94.54	93.04	89.83
	Recall	96.25	100	97.07	97.80	86.41	100	93.53	91.94	87.76
	F1-score	96.35	100	96.99	97.57	75.74	100	92.90	91.31	87.53
DS6-ALL-AML-Leukemia-4C	Accuracy	87.76	87.76	85.62	76.57	59.29	87.71	73.05	82.86	71.24
	Precision	82.46	84.46	82.11	68.24	64.61	82.69	77.25	77.63	67.45
	Recall	81.76	85.49	82.33	68.51	62.00	84.99	75.17	77.51	67.56
	F1-score	80.87	83.57	80.69	65.77	56.33	82.43	70.45	75.55	64.34
DS7-MLL	Accuracy	100	100	98.62	95.81	86.19	100	78.67	100	94.48
	Precision	100	100	98.92	96.27	89.74	100	83.05	100	95.14
	Recall	100	100	98.67	96.44	87.36	100	80.63	100	95.33
	F1-score	100	100	98.67	95.72	85.74	100	77.69	100	94.25
DS8-BreastA	Accuracy	99.47	98.50	95.95	94.97	98.50	98.50	92.92	95.92	82.29
	Precision	98.67	98.19	96.03	94.99	98.19	98.30	93.75	95.91	81.73
	Recall	99.49	98.93	96.15	96.14	98.93	98.93	93.78	96.91	83.26
	F1-score	98.89	98.33	95.59	94.81	98.33	98.39	92.85	95.84	81.53
DS9-BreastB	Accuracy	98.50	99.00	97.45	87.82	99.00	99.00	96.47	94.95	74.66
	Precision	98.00	98.67	97.63	89.20	98.67	98.67	96.90	95.05	75.40
	Recall	99.01	99.49	97.68	88.49	99.49	99.49	96.81	95.50	73.96
	F1-score	98.26	98.89	97.38	87.12	98.89	98.89	96.42	94.88	69.97
DS10-LungA	Accuracy	98.98	99.18	98.36	63.24	71.88	99.98	95.28	98.77	80.5
	Precision	98.91	99.31	98.47	72.34	66.39	99.1	95.81	98.62	85.69
	Recall	99.21	99.35	98.7	61.59	75.37	99.2	95.04	98.99	78.87
	F1-score	99.02	99.29	98.5	59.09	66.37	99.1	95.05	98.74	75.44