CANCER DIAGNOSIS FROM GENE EXPRESSION DATA USING FUZZY CLASSIFIERS AND DEEP LEARNING

A MAJOR PROJECT-II REPORT

SUBMITTED IN FULFILLMENT OF THE REQUIREMENTS

FOR THE AWARD OF THE DEGREE

OF

MASTER OF TECHNOLOGY

IN

INFORMATION SYSTEMS

Submitted by:

YASHPAL SINGH

2K21/ISY/25

Under the supervision of

Dr. SEBA SUSAN

DEPARTMENT OF INFORMATION TECHNOLOGY

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Bawana Road, Delhi -110042

MAY, 2023

DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi-110042

CANDIDATE'S DECLARATION

I, Yashpal Singh, 2K21/ISY/25 student of M.Tech (IT-ISY), hereby declare that the Major Project-II dissertation titled "**CANCER DIAGNOSIS FROM GENE EXPRESSION DATA USING FUZZY CLASSIFIERS AND DEEP LEARNING**" which is submitted by me to the Department of Information Technology, Delhi Technological University, Delhi in fulfillment of the requirement for the award of the degree of Master of Technology is original and not copied from any source without proper citation. This work has not previously formed the basis for the award of any Degree, Diploma Associateship, Fellowship or other similar title or recognition.

Place: Delhi **YASHPAL SINGH**

Date: 29/05/2023

INFORMATION TECHNOLOGY
DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi-110042

CERTIFICATE

I hereby certify that the Major Project-II Dissertation titled "**CANCER DIAGNOSIS FROM GENE EXPRESSION DATA USING FUZZY CLASSIFIERS AND DEEP LEARNING**" which is submitted by Yashpal Singh, 2K21/ISY/25 Information Technology, Delhi Technological University, Delhi in fulfillment of the requirement for the award of the degree of Master of Technology, is a record of the project work carried out by the student under my supervision. To the best of my knowledge this work has not been submitted in part or full for any Degree or Diploma to this university or elsewhere.

Place: Delhi **Prof. SEBA SUSAN**

Date: 29/05/2023 **SUPERVISOR**

ABSTRACT

Microarray gene expression data poses a significant challenge in classification due to its small sample size and high dimensionality. In this thesis, we propose novel approaches for the classification of lung cancer subtypes using advanced techniques and algorithms. The first approach introduces the Fuzzy Min-Max (FMM) classifier, a neuro-fuzzy neural network rarely used for high-dimensional datasets. To enhance the accuracy and speed of FMM, we incorporate the Least Absolute Shrinkage and Selection Operator (LASSO) for optimal gene subset selection. Comparative analysis with other classifiers, including SVM, Random Forest, KNN, Naïve Bayes, and Logistic Regression, validates the superior performance of FMM-LASSO in lung cancer classification. The second approach addresses the challenges of small sample sizes, high dimensionality, and class imbalance in cancer subtyping. Our proposed SMOTE-LASSO-DeepNet framework employs SMOTE for data balancing and LASSO for informative gene selection. The pruned and balanced training set is then fed into a DeepNet model with multiple hidden layers. Extensive testing on four different cancer gene expression datasets demonstrates the consistent superiority of our framework over existing methods. In the third approach, we tackle lung cancer diagnosis using gene expression data. Leveraging the Fuzzy Min-Max (FMM) classifier, specifically the general Fuzzy min-max (GFMM) and enhanced Fuzzy min-max (EFMM) models, we exploit fuzzy class definitions and hyperbox manipulation. LASSO is utilized for informative gene selection, and the performance is evaluated through hyperbox visualization and comparison with state-of-the-art methods. Empirical results showcase the exceptional performance of GFMM with LASSO, achieving validation accuracy of 98.04% and cross-validation accuracy of 94.06%. Collectively, these approaches contribute to the field of cancer diagnosis from gene expression data, offering novel solutions for small sample sizes, high dimensionality, and class imbalance issues. The proposed methodologies demonstrate superior performance compared to existing methods and highlight the potential of neuro-fuzzy systems, deep learning frameworks, and feature selection techniques in improving cancer classification accuracy.

ACKNOWLEDGEMENT

I would like to express my heartfelt gratitude to Professor Seba Susan for her invaluable guidance, support, and mentorship throughout the journey of this thesis. Her expertise, encouragement, and insightful feedback have been instrumental in shaping my research and academic growth. I am truly grateful for her unwavering dedication, patience, and belief in my abilities.

I would also like to extend my deepest appreciation to my parents for their endless love, encouragement, and unwavering support throughout my academic pursuits. Their belief in me and their sacrifices have been a constant source of inspiration. I am grateful for their unwavering faith in my abilities and for always being there to provide guidance and encouragement.

Furthermore, I would like to express my gratitude to the faculty members of the department for their knowledge sharing and for providing a conducive learning environment. Their passion for teaching and dedication to fostering academic excellence have had a profound impact on my education.

This thesis would not have been possible without the support and contributions of these individuals, and I am deeply grateful for their involvement in my academic journey.

YASHPAL SINGH

CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER-1

INTRODUCTION

DNA microarrays are gene chips printed with microscopic spots in defined positions. These spots contain a known DNA sequence that can be It is utilized to analyze gene expression. With the help of microarrays, we can analyze different types of genes simultaneously [1]. Recently, Numerous microarray gene expression datasets are now accessible to the public on the internet. There are many challenges we have to face when we are using microarray datasets, like having thousands of genes in each sample, and relatively smaller number of samples in the dataset. We also have to handle the noisiness of gene expression data [2].

Cancer stands as a widely recognized leading cause of mortality worldwide. It is caused by the abnormal rapid production of cells, producing tumors with different behaviors [3]. On an average, worldwide, one out of six deaths are because of cancer [4]. These facts give rise to the need for an early and accurate diagnosis, which also reduces the side effects of treatment.

Mining of gene expression data has also attracted datamining researchers due to the numerous challenges involved that makes it distinct from patterns found in normal data. A cancer diagnosis is difficult to achieve for various reasons, but recent studies have shown that the diagnosis becomes easy when it is achieved by classifying microarray gene expression data. Gene expression profiling provides valuable and early information about differentially expressed genes associated with different cancer types [5]. The raw microarray gene expression data is in the form of two-dimensional data where the columns represent the genes or features and the rows represent the samples. One problem associated with cancer gene expression datasets is the class imbalance issue in which the population of one class (majority class) far exceeds the population of the other classes (minority classes) [6]. Mining the gene expression data is a challenging task because of the thousands of genes involved with very few samples available. And not all the genes in the dataset make an impact on the final classification results; only a few among the thousands are significant for the model training [7].

This report explains three methods for the classification of cancer gene expression data. The initial approach tackles the challenge of gene expression data handling by incorporating normalization and feature selection techniques to enhance result accuracy.. Specifically, we use Least Absolute Shrinkage and Selection Operator (LASSO) as the feature selection technique. We propose the application of the Fuzzy Min-Max (FMM) neural network classifier to classify the lung cancer dataset using an optimal gene subset constructed using LASSO.

Moreover, the second method is an end-to-end classification framework for identifying cancer subtypes. We use SMOTE and LASSO along with a deep neural network (DeepNet) to simultaneously address the class imbalance issue and high-dimensionality problem associated with cancer gene expression datasets. The DeepNet is one of the best classifiers available today and is known to achieve high accuracies [8]. The small number of samples found in cancer gene expression datasets may suffer from the overfitting problem, which we overcome by using SMOTE for minority class data augmentation. DeepNets achieve good results because of the many hidden layers that facilitate feature transformation and extraction, which train the model much better than other machine learning algorithms [9]. In this study, we use gene expression data belonging to four types of cancers (Lung, Breast cancer, brain, and Leukemia).

In our third method, we investigate the application of GFMM and EFMM neuro-fuzzy classifiers for the classification of lung cancer gene expression data. The application of GFMM and EFMM to gene expression data has not yet been explored. In this work advances on by exploring two advanced architectures of the FMM classifier for application to microarray data. The aim is to exploit the improved functionalities of hyperboxes and the expansion-contraction learning process for determining the decision boundaries between cancer subtypes. The two popular improved versions of the FMM classifier which applied in this method are GFMM and EFMM[47,48]. The GFMM improves the effectiveness of the original fuzzy min–max algorithm by suggesting a few modifications to the general FMM architecture and functioning, some of which are listed below:

1. In the realm of pattern space, the input patterns may manifest as either fuzzy hyperboxes or precise points.
2. The membership function and the hyperbox expansion constraints are modified.
3. GFMM can be used for both clustering and classification because it can process labeled and unlabeled inputs at the same time.
4. In the original algorithm, the number of hyperboxes created depends on the maximum hyperbox size hyperparameter. The smaller the value, the more the number of hyperboxes created, and this leads to overfitting; a larger value creates lesser number of hyperboxes which increases the generalization ability, but then the ability to capture the boundaries between the classes is decreased. So the settlement between these two cases is implemented in GFMM.

In the original FMM, [35] Simpson proposed two different algorithms for classification and clustering problems, but the GFMM combines them in one algorithm. The training of

GFMM is extremely efficient for almost every case because it uses very simple compare, add and subtract operations for hyperbox manipulation. The other very popular version of FMM is the enhanced fuzzy min–max (EFMM) [48] which is known to give high classification performance in case of adequate training. There are three heuristic rules introduced in EFMM which enhances the learning process. Firstly, reducing the overlapping regions of hyperbox during the expansion phase that reduces classification errors. Secondly, the already existing overlap testing phase is extended so that all the overlapping corners can be identified. Thirdly, the existing hyperbox contraction rule in FMM is not able to cover all the overlapping cases, so in EFMM they introduced a new rule for contraction for solving the different overlapping cases.

We analyzed the performance of GFMM and EFMM, and compare the results with that of several other machine learning algorithms. We perform cross-validation for all the classification algorithms, and all the comparisons are made based on accuracy and execution time of the algorithm.

CHAPTER-2

LITERATURE REVIEW

In the field of bioinformatics, numerous researchers have explored microarray data employing various approaches and methodologies, including the utilization of diverse soft computing techniques. [10].

Chen *et al.* [11] presented a deep learning method called D-GEX using an omnibus dataset with 111K gene expression profiles. D-GEX revealed complex patterns of gene expression. It was proved that deep learning-based D-GEX outperformed Linear Regression for gene expression inference on GEO microarray data.

Tabares-Soto *et al.* [12] compared different machine learning algorithms on the 11_tumors dataset, which has eleven types of tumors and achieved high accuracies using the Convolution Neural Network (CNN).

Lyu *et al.* [13], Gullien *et al.* [14], and Mohammed *et al.* [15] also implemented classification models based on deep learning architectures for cancer gene expression data classification.

Mostavi *et al.* [16] tested different Convolutional Neural Network (CNN) models on 33 cancer types; they implemented 1D-CNN, 2D-CNN, and 2D-Hybrid-CNN. The authors achieved the highest accuracies (> 95%) for ID-CNN and 2D-Hybrid-CNN.

A host of classifiers such as the support vector machine [49] random forest of decision trees [50] logistic regression [51] and naïve Bayes classifier [52] have been successfully used for the classification of gene expression data. All these classifiers work on crisp data without transiting to the fuzzy domain. Khan et al. [53] used artificial neural networks (ANN)for the categorization of cancer using gene expression profiles; the main advantage found was that it could work with nonlinear features and has high sensitivity. For the classification of gene expression profiles, Ahmed et al. [54] Among the various neural network models such as the deep neural network (DNN), improved DNN, convolutional neural network (CNN), and recurrent neural network (RNN), coupled with effective preprocessing techniques, it was observed that the improved DNN yielded the most favorable outcome among them all.
Microarray gene expression data facilitates highly efficient cancer diagnosis [17]. Most of the related research involves feature selection for selecting the most informative genes for cancer diagnosis, such as [18], that uses the Particle Swarm Optimization algorithm in a fuzzy multi-objective framework. Hu et al. [19] compared five classification approaches

on seven different microarray cancer datasets with and without gene selection; they proved that data preprocessing improves classification accuracy. Lee et al. [20] compared different feature selection methods for microarray datasets. A fuzzy rough quick, reduced method was proposed in [21] to find the most informative genes using a similarity measure for the classification of lung cancer.

Fuzzy classifiers have proved to perform well in the past due to the computation of fuzzy decision boundaries for classifying the difficult-to-classify samples [22, 23]. A simple fuzzy system was devised in [24] for classifying tumors. Several researchers have tested feature selection and classifier combinations for gene expression data classification, such as Principal Component Analysis (PCA) with Support Vector Machine (SVM) [25] and T-Test with Fuzzy Neural Network and SVM [26] . So in our first method, we used Fuzzy Min-Max (FMM) neural network to classify the microarray gene expression data for lung cancer diagnosis. Before classification, we used the LASSO feature selection technique for extracting the essential features (i.e., genes) [27]. LASSO automatically selects those genes useful for lung cancer classification and discards the redundant genes. LASSO has proved very effective for high-dimensional datasets [28, 29, 30]. It hence is deemed suitable for the microarray gene expression dataset where thousands of genes represent each sample.

Our SMOTE-LASSO-DeepNet is most related to Urda *et al.* [31], who proposed a deep neural network with 2 to 4 hidden layers, each having neurons in the range 10 to 200, in combination with LASSO feature selection, for effective gene expression classification. This method, however, restricts the maximum number of hidden units to 200; it does not recommend any optimal customized architecture suitable for the classification of cancer gene expression data. Also, no solution for class imbalance is provided.

Some works available in the literature have implemented SMOTE for balancing the gene expression datasets [32]. A few have also combined SMOTE or variants of SMOTE with feature selection techniques such as LASSO and Information Gain [33, 34]. However, no prior work has combined SMOTE and LASSO with DeepNets for Cancer subtyping from gene expression data.

In the GFMM and EFMM method, we implemented these two algorithm for the classification of lung cancer subtypes from gene expression data.

CHAPTER-3

CLASSIFCATION OF MICROARRAY DATA USING FUZZY CLASSIFIER WITH LASSO

3.1. LUNG CANCER DATASET

In this method, all the classification experiments are performed on the microarray lung cancer dataset, containing 203 samples and 12600 genes [38]. In other words, there are 203 rows and 12600 columns, and one target class column. In the lung cancer dataset, there are five types of classes which are:

Class 1: adenocarcinomas (139 samples)

Class 2: normal lung tissue sample (17 samples)

Class 3: small cell lung cancer (6 samples)

Class 4: squamous cell carcinomas (21 samples)

Class 5: pulmonary carcinoids (20 samples)

As we see, there are 12600 genes among these, and only a few of them impact the classification. So LASSO extracted 176 genes out of 12600, which are important for the classification task. If we use these extracted genes for the classification, the whole process becomes faster, and accuracy is improved.

3.2. FUZZY MIN-MAX CLASSIFIER

Fuzzy min max classifier outstands among all the classifier we have used for the microarray lung cancer data set because this classifier has ability to learn from a single pass through the data. Simpson et al. [36] proves that the fuzzy min max classifier performs better in case of overlapping classes by finding rational decision boundaries.

In general, fuzzy min max classification training set F contains N ordered pairs

$\{I_h , c_h\}$, where $I_h = (i_{h1} , i_{h2} , \dots , i_{hn}) \in I^n$ is the input string and $c_h \in$

$\{1 , 2 , \dots , m\}$ is the index on one of the m classes. The learning process starts with selecting a ordered pair from set F and find a hyperbox for the same class (if present). If no hyper box found for an ordered pair, then create one and add it to the neural network.

Fuzzy min max classification learning process have three phases:

I. Hyperbox Expansion:

For a given ordered pair {I_ (h), c_h} ∈F, find a hyperbox B_j which provides highest degree of membership and expand the hyperbox if the other hyperbox have the same class as c_h. There is a hyper parameter 0 ≤ϑ ≤1 which bounds the size of hyperbox. For expanding the hyperbox B_jto include I_h, these are the conditions which have to meet:

$$\mathrm{n}\vartheta \ \geq \sum_{k=1}^{n} \left(\max\left(w_{jk} , i_{hk} \right) - \min\left(v_{jk}, i_{hk}\right)\right) \quad (3.1)$$

If this condition got satisfied , we expand hyperbox B_j and the min point of the hyperbox is calculated by :

$$v_{jk}^{new} = \min\left(v_{jk}^{old} , i_{hk}\right) \forall \ \mathrm{k} \ = \ 1,2,\dots,\mathrm{n} \qquad (3.2)$$

$$w_{jk}^{new} \ = \ \max(w_{jk}^{old} , i_{hk}) \ \forall \ \mathrm{k} \ = \ 1,2,\dots,\mathrm{n} \qquad (3.3)$$

After successful hyper box expansion, we move to the next step which is hyperbox overlap test.

II. Hyperbox Overlap Test

Expansion of hyperbox may create an overlap with other hyperbox, overlapping of two hyperbox creates a problem only if the other hyperbox is from a different class. A dimension by dimension comparison between hyperboxes is performed for determining whether an expansion creates an overlap or not.

Now assume that B_j is expanded in the last step and the B_l is from another class and we have to test for the possible overlap. While testing, for each dimension at least one of the four cases is satisfied. Initially, δ^old=1 and the test cases are as follows:

$$\boldsymbol{case\ 1: v_{jk} < v_{lk} < w_{jk} < w_{lk}\ ,}$$

$$\boldsymbol{\delta^{new} = min(w_{jk} - v_{lk}\ , \delta^{old})}$$

$$\boldsymbol{case\ 2: v_{lk} < v_{jk} < w_{lk} < w_{jk}\ ,}$$

$$\boldsymbol{\delta^{new} = min(w_{lk} - v_{jk}\ , \delta^{old})}$$

$$\boldsymbol{case\ 3: v_{jk} < v_{lk} < w_{lk} < w_{jk}\ ,}$$

$$\boldsymbol{\delta^{new} = min(min(w_{lk} - v_{jk}\ , w_{jk} - v_{lk})\ , \delta^{old})}$$

$$\boldsymbol{case\ 4: v_{lk} < v_{jk} < w_{jk} < w_{lk}\ ,}$$

$$\boldsymbol{\delta^{new} = min(min(w_{jk} - v_{lk}\ , w_{lk} - v_{jk})\ , \delta^{old})}$$

Now , if $\delta^{old} - \delta^{new} > 0$, then $X = k$ and $\delta^{old} = \delta^{new}$, indicates that there was an overlap for the X^{th} dimension and we choose only where the overlap is minimal. If this not the case, then testing stops and the next step for contraction is not necessary. Otherwise, after testing we have to perform the contraction step to remove the overlap between the hyperboxes.

III. Hyperbox Contraction

After the testing, we got the dimension where the overlap is minimal between two hyperboxes. Now we want to contract the expanded box in such a way that the contraction size is as small as possible and removes the overlap. For determining the proper adjustment of hyperboxes there are four cases.

In fuzzy min max these three phases are repeat until all the ordered pairs of F are processed.

3.3. HYPER PARAMETERS

Table 3.1 shows the hyper parameter values for all the algorithms that are used in this method.

Classifier	**Hyper parameter**	**Values**
Fuzzy Min Max Classifier (FMMC)	Hyper box coefficient (P)	0.7
	Sensitivity (ω)	1
Support vector Machine (SVM)[41]	Regularization parameter (c)	1
	Gamma	0.0018 , 0.126(WITH SELECTED FEATURES)

K- Nearest Neighbor[44]	No. of neighbors	7
Logistic Regression[43]	C	1
	Penalty	l2
	Solver	lbfgs
Naïve Bayes [45]	Var_smoothing	$1e^{-9}$
Random Forest [42]	N_estimators	100
	Max_depth	2

Table 3.1 Hyper parameter of algorithms.

3.4. METHODOLOGY

Step 1: Load the Microarray lung cancer dataset.

Step 2: Normalize the dataset using min max normalization and the range of the normalization is [0, 1]. In min-max normalization all the minimum values set to 0 and all the maximum values are set to 1. And the values which lies between maximum and minimum values are set with a decimal value within a range of [0, 1].

Step 3: In this step we have two choices first is to go for classification without any feature selection and the second one is before going for the classification process extract features with LASSO and then perform classification.

1. Directly perform classification by using these six classifiers (FMMC, SVM, RF, KNN, Logistic R, NB)
2. Extract the important features from the lung cancer dataset. There is the requirement for this step because our dataset have 12603 genes (i.e. features) and not all the genes make impact on the final result. I used LASSO feature extraction technique for extracting the features. After the extraction process there are 72 features got extracted and now these extracted features are used for the training the model. This step makes whole this classification faster and more efficient.

Step 4: Divide the dataset into training and testing set. The ratio of training testing split is 70:30.

Step 5: Now fit the model with training set. Calculate the accuracy & F1-score and store it into their respective array. Repeat step 4&5 for the five times because we are performing fivefold cross validation.

Step 6: Average out the accuracies of the all five iterations of the validation. Same for the F1-score.

CHAPTER-4

CLASSIFICATION OF CANCER GENE EXPRESSION DATA USING SMOTE AND DEEP LEARNING

4.1. DATASET

We used four benchmark gene expression datasets for cancer subtype identification: - Lung cancer, Brain cancer, Breast cancer, and Leukemia (Blood cancer) [37, 38, 39]. In Table 4.1, we show the description of these datasets.

The datasets have good sample quality and they are manually curated for research purposes. All four datasets have large numbers of genes with limited samples, and an imbalanced class distribution. In the Brain and Breast cancer dataset, there are 54,676 genes which is the highest among all datasets.

Cancer type	**Samples**	**Genes**	**Classes**
Leukemia	64	22,284	5
Lung Cancer	203	12,600	5
Brain Cancer	130	54,676	5
Breast Cancer	151	54,676	6

Table 4.1. Cancer gene expression datasets

4.2. OVERALL PROCESS FLOW

We start by loading the cancer gene expression data. The first step is preprocessing the dataset in which the values of the features are scaled in the range of 0 to 1 using the *MinMaxScaler* function. Then we divide the dataset into two equal sets: - validation (*V*) and cross-validation (*CV*), by selecting alternate samples for training and testing. For the *V* part, we use the original training and test sets for training and testing, respectively. For the *CV* part, the original testing set is the new training set, and the initial training set is used for testing. Before training the model, we extract the essential genes from the training set using the LASSO feature selector. We use six different classifier models for performance evaluation apart from the proposed DeepNet model. For all the models, the process flow is the same. However, our proposed method uses SMOTE for minority class data augmentation before the feature selection stage. The last step is predicting the test set and analyzing the performance of different models. We compare all classifiers' validation (*V*) and cross-validation (*CV*) accuracies and calculate the F1-Scores.

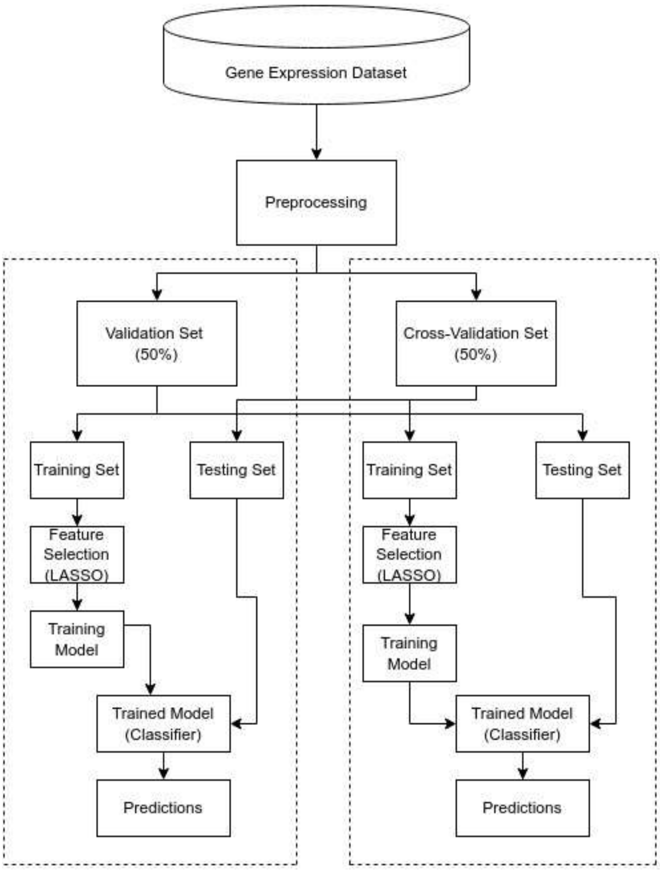


Fig 4.1. Overall process flow

4.3. FEATURE SELECTION USING LASSO

While analyzing the complexities of handling the gene expression data, the main problem found is that these datasets are high-dimensional, containing thousands of genes, and the majority of the features are irrelevant for cancer subtyping, and have minimal impact on the final classification results. These unwanted genes make the whole process slow and also lower the performance of the model. To overcome this problem, we used LASSO feature selection [6] whose main purpose is to select only those genes that are important for the learning process, and remove those that are unwanted. LASSO uses the following cost function to minimize the differences between the real and predicted values.

$$l_1 = \frac{1}{2X_{training}} \sum_{i=1}^{X_{training}} \left(Y_{real}^{(i)} - Y_{pred}^{(i)}\right)^2 + \alpha \sum_{j=1}^{n} |\vartheta_j| \quad (4.1)$$

In (1), ϑ_j is the coefficient of the j^{th} feature and α is the hyper parameter that sets the penalty term; we set it to 0.001.

The aim is to optimize the cost function by reducing the absolute values of the coefficient. It selects those features which are useful and discards those which are unwanted by making their coefficient value zero.

4.4. HYPER PARAMETERS

The methods along with their hyper parameters are listed in Table 4.2.

Methods	**Hyper parameters**	**values**
SMOTE-LASSO-DeepNet (proposed)	Activation Function	ReLU (for i/p and hidden layer), softmax (o/p layer)
	Hidden Layers	4
	Number of units per layer	[512, 256, 128, 64]
	Optimizer	adam
	Loss Function	sparse_categorical_crossentropy
LASSO-SMOTE-DeepNet	Activation Function	ReLU (for i/p and hidden layer), softmax (o/p layer)
	Hidden Layer	4
	Number of Units per layer	[512, 256, 128, 64]
	Optimizer	adam
	Loss Function	sparse_categorical_crossentropy
LASSO-DeepNet	Activation Function	ReLU (for i/p and hidden layer), softmax (o/p layer)
	Hidden Layers	4
	Number of units per layer	[512, 256, 128, 64]
	Optimizer	adam
	Loss Function	sparse_categorical_crossentropy
FMM-LASSO [40]	Hyperbox Coefficient	0.7

	Sensitivity	1
Support Vector Machine [41] (with LASSO)	Regularization parameter c	0.0018
	gamma	0.126
Random Forest [42] (with LASSO)	N_estimators	100
	Max_depth	3
Logistic Regression [43] (with LASSO)	C	1
	Penalty	12
	Solver	lbfgs
K-Nearest Neighbor [44] (with LASSO)	No. of neighbors	5
Naïve Bayes [45] (with LASSO)	Var_Smoothing	$1e^{-9}$

Table 4.2 Hyper Parameter settings

4.5. SMOTE-LASSO-DeepNet FRAMEWORK

The proposed framework uses DeepNet to identify the cancer subtypes from gene expression datasets. A deep neural network has an architecture resembling a multi-layer perceptron [46] but with deeper layers. The proposed DeepNet architecture has four hidden layers with [512, 256, 128, 64] units, respectively, as shown in Fig. 4.2. The SMOTE-LASSO-DeepNet framework proposed for the classification of cancer gene expression data offers an end-to-end solution consisting of three distinct phases, each designed to address specific challenges associated with the gene expression data. In the first phase, SMOTE is employed on the minority classes of the training set to achieve class distribution balance. SMOTE is particularly favored for handling imbalanced data in multi-class classification tasks due to its straightforward application, as it generates synthetic data points that closely resemble the original data points through an interpolation process.

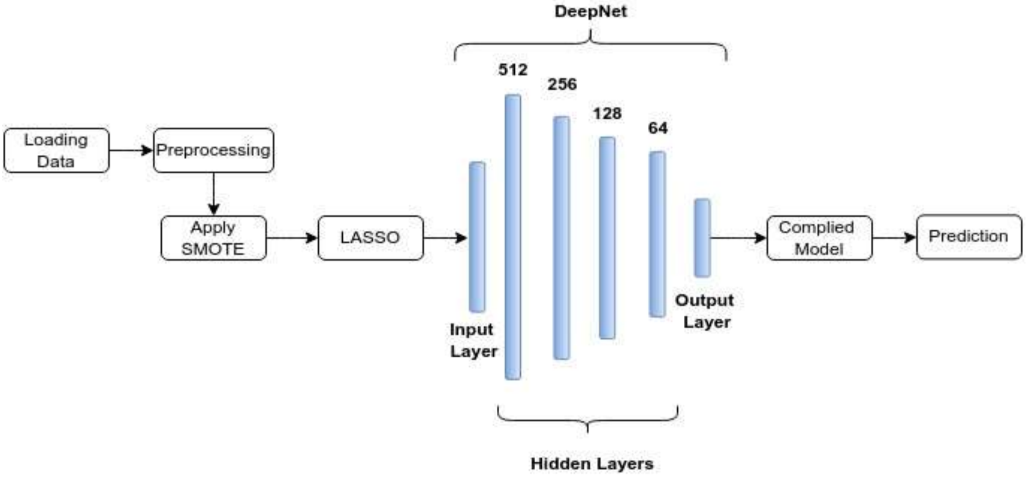


Fig 4.2. Architecture of proposed SMOTE-LASSO-DeepNet framework for cancer subtype classification

In Phase 2, we apply LASSO feature selection on the balanced training data; it selects the important features and also speeds up the whole process of classification. We use the final selected features for the model training; the scores are computed separately for the V and CV cross-validation process as illustrated in the process flow in Fig. 4.1. In Phase 3, we use a DeepNet of architecture $X - 512 - 256 - 128 - 64 - Y$, where, X is the number of input features selected by LASSO, and Y is the number of target classes or cancer subtypes. For the hidden layers, we used 'ReLU' as an activation function, and for the output layer 'softmax' function is used because this model is used for multi-class classification. For all the datasets, we used a batch size of 30, with 100 epochs. We have compared the performance of the proposed deep learning framework with popular machine learning algorithms using the performance metrics of accuracy and F1-score. The performance of the proposed framework is compared both with and without SMOTE.

CHAPTER-5

CLASSIFCATION OF MICROARRAY DATA USING GFMM & EFMM CLASSIFIER WITH LASSO

5.1. DATASET

The dataset used in this method is same as section 3.1. For more info about the dataset please refer section 3.1.

5.2. GENERAL FUZZY MIN-MAX NEURAL NETWORK

In this section, we discuss about the input patterns of GFMM, learning algorithm phases, and the neural network at the core of GFMM for the current task of classification of microarray gene expression data.

5.2.1. Input Pattern

The input that is processed by GFMM is the ordered pair of the h^{th} input pattern and the class index of one of the classes. The ordered pair is given by

$$\{\boldsymbol{I_h} \ , \boldsymbol{c_h}\} \qquad (5.1)$$

Where I_h is the h^{th} input pattern in form of I_h^l (lower) and I_h^u (upper) i.e. $[\ I_h^l\ , I_h^u\ ]$ these are the vector inputs. $c_h \epsilon \{\ 0, 1\ , 2\ , 3, \dots., p\ \}$ is the class index of any one of the p+1 classes. If $c_h = 0$, it means the input is unlabeled.

5.2.2. Membership function

The fuzzy hyperbox membership function plays an important role in deciding whether a particular input belongs to a particular class or not. In GFMM, a new membership function is defined which fulfills the limitations of original fuzzy min-max. In the original function, it was observed that by increasing the distance from the hyperbox, the membership does not decrease steadily, which is a major drawback of this membership function. The degree of membership of I_h for the hyperbox b_q is 1 if I_h is inside the hyperbox b_q , and the membership decreases as the distance from the hyperbox is increases. In the membership equation the $\gamma = [\gamma_1\ , \gamma_2\ , \dots\dots, \gamma_n\ ]$ is the sensitivity parameter this regulates how fast the membership values decreases.

$$\boldsymbol{b_q(I_h) = min_{p=1\,to\,n}\left(min\begin{pmatrix}[1 - f(I_{hp}^{u} - w_{qp}\,,\gamma_p)]\,, \\ [1 - f(v_{qp} - I_{hp}^{l}\,,\gamma_p)]\end{pmatrix}\right)} \qquad (5.2)$$

$$\boldsymbol{where\,, f(r,\gamma) = \begin{Bmatrix} 1 \;\; if\; r\gamma > 1 \\ r\gamma \;\; if\; 0 \le r\gamma \;\le 1 \\ 0 \;\; if\; r\gamma < 0 \end{Bmatrix}}$$

5.2.3. GFMM Learning Algorithm

The steps of the GFMM learning algorithm are given below.

1. Min and Max point initialization

   For the new hyperbox the algorithm initializes its min point $V_q = 0$ and the max point $W_q = 0$, this can be automatically used in the expansion phase of the algorithm. The values of min and max point when the q^{th} hyperbox is adjusted for the first time by using the $I_h = [I_h^l\,, I_h^u]$ is

   $$\boldsymbol{V_q = I_h^l \quad , \quad W_q = I_h^u} \qquad (5.3)$$

   This values are similar to the input pattern.

2. Hyperbox expansion

   Suppose the h^{th} input pattern has to be expanded with the hyperbox B_q which have the highest degree of membership;before expansion the following condition has to be satisfied

   $$\forall_{\boldsymbol{p=1..n}} \left(\boldsymbol{max}\left((\boldsymbol{w_{qp}}\,, \boldsymbol{I_{hp}^u}) - \boldsymbol{min}(\boldsymbol{v_{qp}}\,, \boldsymbol{I_{hp}^l})\right)\right) \le \boldsymbol{\vartheta} \qquad (5.4)$$

   In (5.4), $\vartheta$is a user-defined value which sets an upper bound on the maximum size of a hyperbox. If the condition in (5.4) got satisfied, then the new min and max points of the hyperbox B_q are given by (5.5) and (5.6), respectively.

   $$\boldsymbol{v_{qp}^{new} = min(v_{qp}^{old}\,, I_{hp}^l)}(5.5)$$

   $$\boldsymbol{w_{qp}^{new} = max(w_{qp}^{old}\,, I_{hp}^u)}\ (5.6)$$

3. Hyperbox overlap test

    After the successful expansion, there are chances of overlap between the two hyperboxes and if both these hyperboxes belongs to the different classes, then the classifier will give wrong results. The algorithm conducts the hyperbox overlap test to check for the overlap

    Let the hyperbox B_q is expanded and test for the overlap with B_p if

$$\boldsymbol{class(B_q)} = \left\{ \begin{array}{ll} \boldsymbol{0}, & \boldsymbol{test\ with\ all\ the} \\ & \boldsymbol{other\ hyperboxes.} \\ \boldsymbol{other}, & \boldsymbol{go\ for\ the\ overlapping} \\ & \boldsymbol{test\ only\ if} \\ & \boldsymbol{class(B_q) \neq class(B_p)} \end{array} \right\} \quad (5.7)$$

4. Hyperbox contraction

    Δ^{th} dimension of the two hyperboxes is adjusted only if $\Delta > 0$. To make minimal effect on the size and shape of the hyperbox, the only one dimension is adjusted in each hyperbox. The contraction phase of GFMM is very similar to the original Fuzzy min–max.

5.2.4. GFMM Learning Algorithm

There are only two changes between the GFMM network architecture shown in Figure 5.1 and Simpson's original FMM network architecture. Firstly, the input node gets doubled to 2×n. Secondly, in the output layer, an additional node is introduced which handles the unlabeled hyperbox from the second layer of the network.

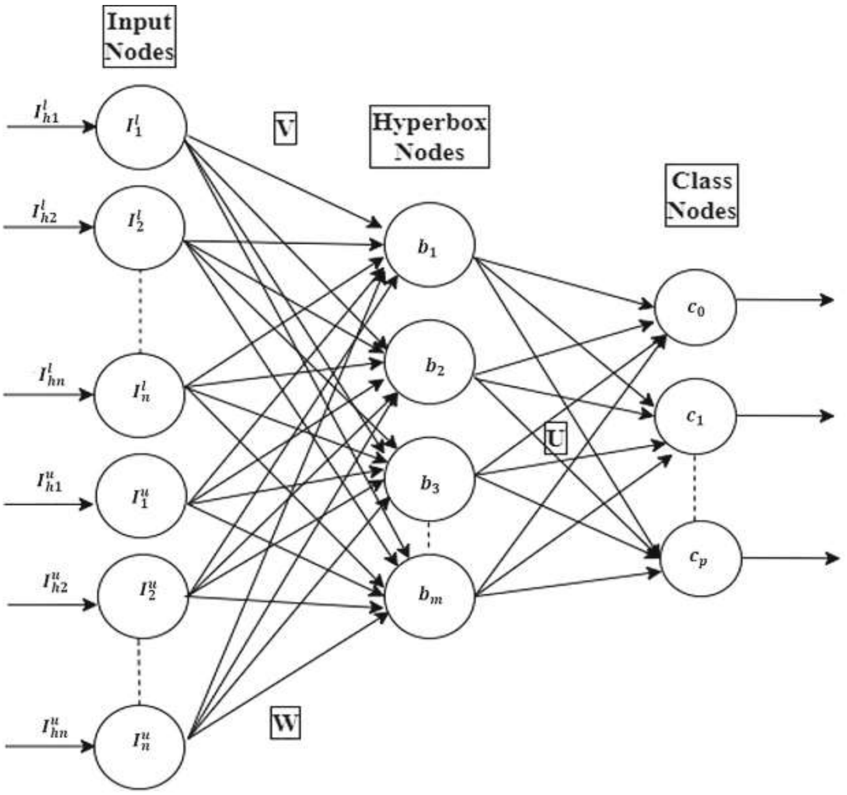


Fig 5.1 The network of GFMM.

5.3. ENHANCED FUZZY MIN-MAX NEURAL NETWORK

The enhanced Fuzzy min–max neural network (EFMM) overcomes the limitation of the original FMM learning algorithm and enhances its performance. There are three heuristic rules for the learning algorithm, as will be discussed in this section

5.3.1. Shortcomings of FMM

The three shortcomings of FMM that are overcome by EFMM are summarized below

1. Hyperbox expansion: In this phase it is shown that when the overlapping regions are increasing between two classes it makes an impact on the performance of the FMM. In FMM they first calculate the sum of all the differences between min and max points of the dimensions, and then they compare this sum with nϑ. There are very high chances of wrong prediction even if one dimension can exceed then ϑ(expansion coefficient) and the sum of all dimensions is under the expansion coefficient. This can lead to overlapping regions between different hyperboxes.

2. Hyperbox overlap test: The four existing cases for detecting the overlap between two different class hyperboxes are not sufficient. There are some inputs in which

overlapping regions are detected and the test assumes it is a nonoverlapping region and it stops the overlap test. So more conditions are added in the overlap test of EFMM.

3. Hyperbox contraction: In FMM the contraction is based on the hyperbox overlap test, but the overlap test phase can pass some undetected overlapping regions which creates problems in the contraction phase.

In EFMM they modified all these three phases to overcome these problems. The modified version improves the classification results.

5.3.2. EFMM Learning algorithm

The three heuristic rules which can overcome all the limitations of EFMM are:

1. Hyperbox expansion rule: To solve all expansion problems in FMM, a new equation is formulated. The q^{th} hyperbox is checked from all dimensions separately to see if it exceeds ϑ or not. This rule is only applicable if no dimension exceeds ϑ.

$$\boldsymbol{Max_n}(\boldsymbol{W_{qp}}, \boldsymbol{I_{hp}}) - \boldsymbol{Min_n}(\boldsymbol{V_{qp}}, \boldsymbol{I_{hp}}) \leq \boldsymbol{\vartheta} \qquad (5.8)$$

2. Hyperbox overlap test rule: In the original FMM, the four cases are insufficient for the hyperbox overlap test. In GFMM, they modified the test phase and included additional overlap testing cases, as observed from (5.7). Now there are total nine cases to detect possible overlap regions. And (5.9) and (5.10) are already there in FMM.

Initially, $\delta^{old} = 1$

$$case\ 1: V_{qp} < V_{rp} < W_{qp} < W_{rp}\ , \qquad \delta^{new} = min(W_{qp} - V_{rp}\ , \delta^{old}) \quad (5.9)$$

$$case\ 2: V_{rp} < V_{qp} < W_{rp} < W_{qp}\ , \qquad \delta^{new} = min(W_{rp} - V_{qp}\ , \delta^{old}) \quad (5.10)$$

$$case\ 3: V_{qp} = V_{rp} < W_{qp} < W_{rp}\ , \quad \delta^{new} = min(min(W_{qp} - V_{rp}\ , W_{rp} - V_{qp}\ ), \delta^{old}) \quad (5.11)$$

$$case\ 4: V_{qp} < V_{rp} < W_{qp} = W_{rp}\ , \quad \delta^{new} = min(min(W_{qp} - V_{rp}\ , W_{rp} - V_{qp}\ ), \delta^{old}) \quad (5.12)$$

$$case\ 5: V_{rp} = V_{qp} < W_{rp} < W_{qp}\ ,$$
$$\delta^{new} = min\big(min\big(W_{qp} - V_{rp}, W_{rp} - V_{qp}\big), \delta^{old}\big) \quad (5.13)$$

$$case\ 6: V_{rp} < V_{qp} < W_{rp} = W_{qp}\ ,$$
$$\delta^{new} = min\big(min\big(W_{qp} - V_{rp}, W_{rp} - V_{qp}\big), \delta^{old}\big) \quad (5.14)$$

$$case\ 7: V_{qp} < V_{rp} \leq W_{rp} < W_{qp}\ ,$$
$$\delta^{new} = min\big(min\big(W_{qp} - V_{rp}, W_{rp} - V_{qp}\big), \delta^{old}\big) \quad (5.15)$$

$$case\ 8: V_{rp} < V_{qp} \leq W_{qp} < W_{rp}\ ,$$
$$\delta^{new} = min\big(min\big(W_{qp} - V_{rp}, W_{rp} - V_{qp}\big), \delta^{old}\big) \quad (5.16)$$

$$case\ 9: V_{rp} = V_{qp} < W_{rp} = W_{qp}\ ,$$
$$\delta^{new} = min\Big(\big(W_{rp} - V_{qp}\big), \delta^{old}\Big) \quad (5.17)$$

When $\delta^{old} - \delta^{new} < 1$, then only overlapping region is detected. To check for the next dimension we have to initialize $\Delta = p\ and\ \delta^{old} = \delta^{new}$. And this loop ends when no more regions are detected.

3. Hyperbox contraction rule: For the contraction of the overlapping hyperboxes, EFMM introduces nine cases and, all these cases are totally based on the overlap test rules.

$$Case\ 1: V_{q\Delta} < V_{r\Delta} < W_{q\Delta} < W_{r\Delta}, W_{q\Delta}^{new} = V_{r\Delta}^{new} = \frac{W_{q\Delta}^{old} + V_{r\Delta}^{old}}{2} \quad (5.18)$$

$$Case\ 2: V_{r\Delta} < V_{q\Delta} < W_{r\Delta} < W_{q\Delta}, W_{r\Delta}^{new} = V_{q\Delta}^{new} = \frac{W_{r\Delta}^{old} + V_{q\Delta}^{old}}{2} \quad (5.19)$$

$$Case\ 3: V_{q\Delta} = V_{r\Delta} < W_{q\Delta} < W_{r\Delta}, V_{r\Delta}^{new} = W_{q\Delta}^{new} \quad (5.20)$$

$$Case\ 4: V_{q\Delta} < V_{r\Delta} < W_{q\Delta} = W_{r\Delta}, W_{q\Delta}^{new} = V_{r\Delta}^{new} \quad (5.21)$$

$$Case\ 5: V_{r\Delta} = V_{q\Delta} < W_{r\Delta} < W_{q\Delta}, V_{q\Delta}^{new} = W_{r\Delta}^{new} \quad (5.22)$$

$$Case\ 6: V_{r\Delta} < V_{q\Delta} < W_{r\Delta} = W_{q\Delta}, W_{r\Delta}^{new} = V_{q\Delta}^{new} \quad (5.23)$$

$$Case\ 7(a): V_{q\Delta} < V_{r\Delta} \leq W_{r\Delta} < W_{q\Delta}\ and\ \big(W_{r\Delta} - V_{q\Delta}\big)$$
$$< \big(W_{q\Delta} - V_{r\Delta}\big), V_{q\Delta}^{new} = W_{r\Delta}^{new} \quad (5.24)$$

$$Case\ 7(b): V_{q\Delta} < V_{r\Delta} \leq W_{r\Delta} < W_{q\Delta}\ and\ (W_{r\Delta} - V_{q\Delta}) > (W_{q\Delta} - V_{r\Delta}), W_{q\Delta}^{new} = V_{r\Delta}^{new} \tag{5.25}$$

$$Case\ 8(a): V_{r\Delta} < V_{q\Delta} \leq W_{q\Delta} < W_{r\Delta}\ and\ (W_{r\Delta} - V_{q\Delta}) < (W_{q\Delta} - V_{r\Delta}), W_{r\Delta}^{new} = V_{q\Delta}^{new} \tag{5.26}$$

$$Case\ 8(b): V_{r\Delta} < V_{q\Delta} \leq W_{q\Delta} < W_{r\Delta}\ and\ (W_{r\Delta} - V_{q\Delta}) > (W_{q\Delta} - V_{r\Delta}), V_{r\Delta}^{new} = W_{q\Delta}^{new} \tag{5.27}$$

$$Case\ 9(a): V_{q\Delta} = V_{r\Delta} < W_{q\Delta} = W_{r\Delta}, W_{q\Delta}^{new} = V_{r\Delta}^{new} = \frac{W_{q\Delta}^{old} + V_{r\Delta}^{old}}{2} \tag{5.28}$$

$$Case\ 9(b): V_{r\Delta} = V_{q\Delta} < W_{r\Delta} = W_{q\Delta}, W_{r\Delta}^{new} = V_{q\Delta}^{new} = \frac{W_{r\Delta}^{old} + V_{q\Delta}^{old}}{2} \tag{5.29}$$

These are the nine cases for hyperbox contraction in EFMM. These three heuristic rules are the main reason for the enhancement of the learning algorithm of EFMM over FMM.

5.4. METHODOLOGY

The task at hand in our current work is to identify lung cancer subtypes from the gene expression profiles pertaining to lung cancer data. The details of the dataset are given in Section 5.1. The process flow of the training and testing procedures for GFMM / EFMM is shown in Figure 5.2. The dataset is split into two equivalent halves using alternate samples for training and testing (50:50 train:test split ratio). For each classifier, we perform the validation (V) and cross-validation (CV) steps. In V step, the classifier is trained using the train set, and the trained model is used to classify the test set. The CV results are obtained by swapping the training and test sets. We followed the train : test split and validation procedure as in [30] LASSO feature selection is used to select significant and informative genes prior to the classification phase. With a reduced gene subset, the whole classification process becomes faster and the results are also impressive.

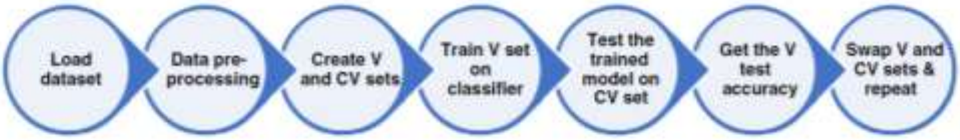

Load dataset
Data pre-processing
Create V and CV sets
Train V set on classifier
Test the trained model on CV set
Get the V test accuracy
Swap V and CV sets & repeat

Fig 5.2 Flow chart of the training and testing process.

The steps of the methodology are detailed below:

1. Load the Microarray gene expression dataset.

2. Normalize the dataset using min–max normalization and the range of the normalization is [0, 1]. In min–max normalization, all the minimum values are set to 0 and all the maximum values are set to 1. The values which lie between maximum and minimum values are set with a decimal value within a range of [0, 1].

3. In this step, we select the important features with LASSO and then perform the classification task.

    i. Directly go for 50:50 train: test split and then perform classification by GFMM, EFMM and other classifiers and models that are used for comparison.
    ii. Extract the important features from the training set of the lung cancer dataset. There is a requirement for this step because the lung cancer dataset has 12,600 genes (i.e. features) and not all the genes make an impact on the final result. We used the LASSO feature extraction technique for extracting the features from both the training and test sets. This step makes the whole classification process faster and more efficient. After the feature selection stage, the selected gene pool is used for training of the model. The trained model is applied obtain the test accuracies also known as Validation (V) accuracies.
    iii. Now swap the training and test sets and repeat step (ii) by performing feature selection using LASSO on the new training set. The test accuracies are compiled which are known as the cross-validation (CV) accuracy.

CHAPTER-6

EXPERIMENTAL RESULTS

6.1. FMM-LASSO RESULTS:

In fig 6.1, the bar graph compares the accuracy of all the classifiers used in the experiment. The fuzzy min-max classifier with LASSO feature extraction performs the best among all other methods with 95.08% accuracy on the microarray lung cancer dataset. The classes in the dataset are unbalanced, so we also conducted the f1-score test for all the classifiers [47]. As we see in fig 6.2, the fuzzy min-max classifier with LASSO gets the highest F1-score of 0.92 compared to other classifiers.

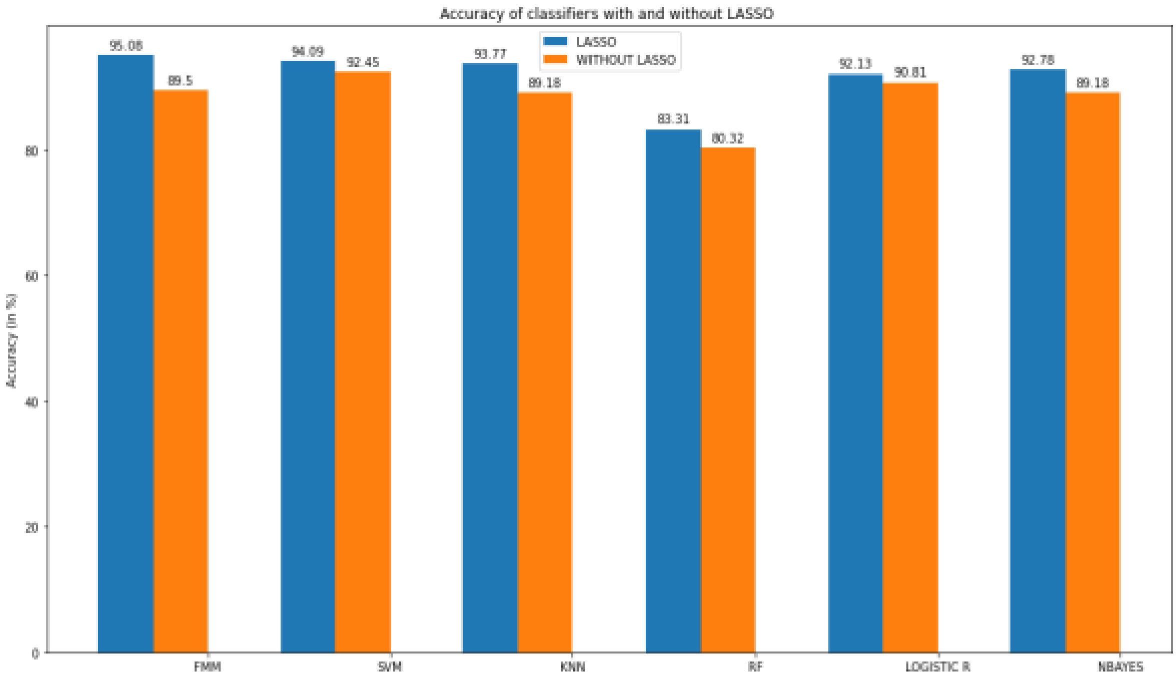


Fig 6.1. Test accuracies of Lung cancer classification

The final f1-score of every classifier is the average of the f1-scores that we got in all five iterations of classification. After analyzing the f1-score and accuracies we can say 9 that the Fuzzy min-max classifier performs best in the case of LASSO feature extraction and SVM performs better in the case of without feature extraction.

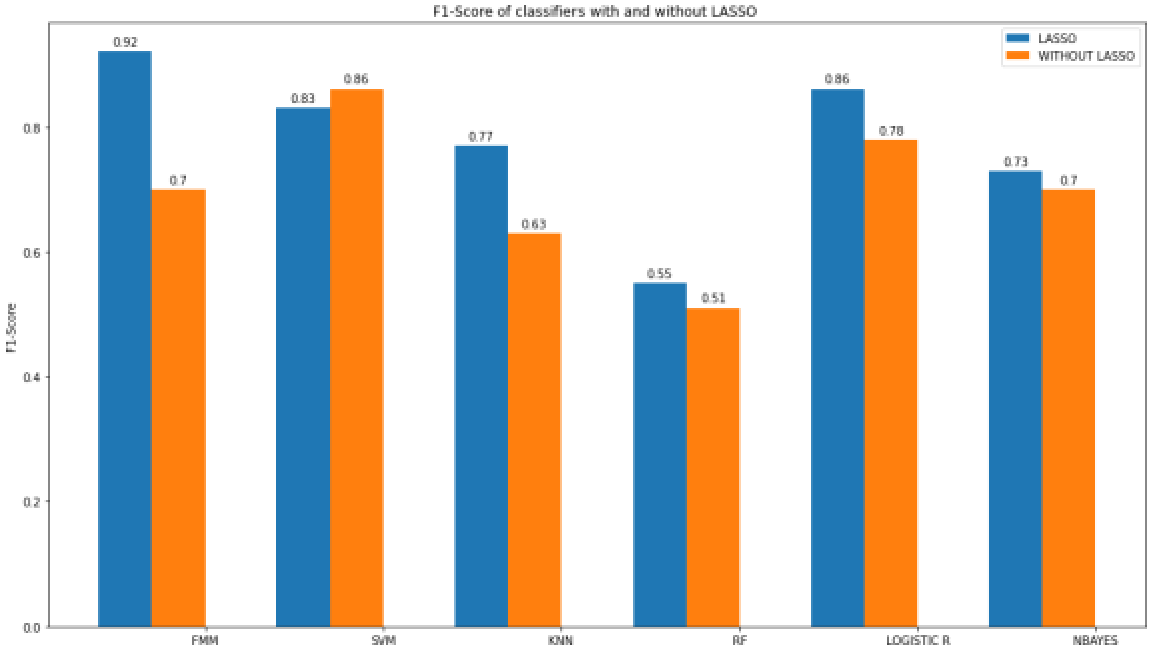


Fig 6.2. F1-Score of lung cancer classification

In table 6.1 we have shown the execution time comparison of all the six classifiers, and we have analyzed that fuzzy min-max takes more time to compare to SVM and other classifiers. LASSO make a major impact on everything like we can see in the bar graphs the accuracies are comparatively very less in without LASSO case, and the same observation for f1-score and the execution time.

Classifiers	**Time (in secs)**	
	LASSO	**Without LASSO**
Fuzzy Min Max Classifier	13.77	963.31
Support Vector Machine (SVM)	0.40	17.74
K- Nearest Neighbor	0.37	0.37
Random Forest	1.98	2.90
Logistic Regression	0.66	8.51
Naïve Bayes	0.32	1.39

Table 6.1. Execution Time of Classifiers

6.2. SMOTE-LASSO-DeepNet FRAMEWORK RESULTS:

All the results were calculated in a setting where the dataset is divided into two sets, one for odd-numbered points and the other for even-numbered points, as illustrated in the process flow in Fig. 4.1. In the validation (*V*) stage, the training and test sets were used as it is, while in the cross-validation (*CV*) stage, the training and test sets were interchanged. For both parts, LASSO was used for feature selection. For the performance analysis, we calculated the accuracy and F1-score of all methods. The classification results of all four datasets are given in Tables 6.2 to 6.5 for the Leukemia, Lung cancer, Brain cancer and Breast cancer datasets, respectively.

Methods	**Accuracy (in%)**		**F1-Score**	
	Validation	**Cross-validation**	**Validation**	**Cross- Validation**
SMOTE-LASSO-DeepNet	**98.13**	**100**	**0.98**	**1.0**
LASSO-SMOTE-DeepNet	96.25	**100**	0.95	**1.0**
LASSO-DeepNet	96.25	98.125	0.94	0.98
FMM-LASSO	90.62	93.75	0.92	0.94
SVM-LASSO	96.87	96.87	0.95	0.96
RF-LASSO	96.25	96.25	0.96	0.95
KNN-LASSO	96.87	**100**	0.95	**1.0**
NB-LASSO	81.25	84.37	0.75	0.82
LR-LASSO	96.87	**100**	0.95	**1.0**

Table 6.2. Results for Leukemia

Methods	**Accuracy (in %)**		**F1-Score**	
	Validation	**Cross-validation**	**Validation**	**Cross- Validation**
SMOTE-LASSO-DeepNet	**95.68**	**94.06**	**0.91**	**0.85**
LASSO-SMOTE-DeepNet	95.49	91.28	**0.91**	0.78
LASSO-DeepNet	94.31	89.9	0.87	0.76
FMM-LASSO	90.19	93	0.86	**0.85**
SVM-LASSO	93.13	92.07	0.74	0.72
RF-LASSO	91.76	85.34	0.7	0.59
KNN-LASSO	94.11	89.1	0.83	0.68
NB-LASSO	94.11	84.15	0.75	0.6
LR-LASSO	95	89.1	0.85	0.74

Table 6.3. Results for Lung Cancer

In Table 6.2, as we observe in the case of the Leukemia dataset, SMOTE-LASSO-DeepNet performs best among all the classifiers. KNN-LASSO, LR-LASSO, and LASSO-SMOTE-DeepNet also achieve 100% accuracy in cross-validation, but in the validation case, the highest accuracy is 98.13% with an F1-score of 0.98 for the proposed method. In the Lung cancer results shown in Table 5, SMOTE-LASSO-DeepNet achieves the highest accuracies of 95.68% and 94.06% (for *V* and *CV* respectively). However, in case of the Brain cancer results shown in Table 6.4, the SVM-LASSO and LR-LASSO perform slightly better in the *V* and *CV* cases, respectively.

Methods	**Accuracy (in %)**		**F1-Score**	
	Validation	**Cross-validation**	**Validation**	**Cross- Validation**
SMOTE-LASSO-DeepNet	91.3	83.3	0.89	0.83
LASSO-SMOTE-DeepNet	90.7	84	0.89	0.83
LASSO-DeepNet	90.7	83.6	0.89	0.82
FMM-LASSO	89.23	84.6	0.86	0.82
SVM-LASSO	**92.3**	84.6	**0.9**	0.82
RF-LASSO	90.15	86.15	0.87	0.84
KNN-LASSO	89.23	78.46	0.86	0.79
NB-LASSO	87.69	80	0.85	0.77
LR-LASSO	90.76	**87.69**	0.88	**0.88**

Table 6.4. Brain Cancer Results

Methods	**Accuracy (in %)**		**F1-Score**	
	Validation	**Cross-validation**	**Validation**	**Cross- Validation**
SMOTE-LASSO-DeepNet	84.47	**91.46**	0.84	**0.92**
LASSO-SMOTE-DeepNet	81.31	**91.46**	0.83	0.91
LASSO-DeepNet	81.31	90.93	0.82	0.91
FMM-LASSO	72.36	81.33	0.76	0.81
SVM-LASSO	85.52	84	0.83	0.77
RF-LASSO	78.15	**91.46**	0.77	0.79
KNN-LASSO	80.26	81.33	0.81	0.77
NB-LASSO	84.21	89.33	0.84	0.77
LR-LASSO	**86.84**	88	**0.86**	0.87

Table 6.5. Breast Cancer Results

SMOTE-LASSO-DeepNet achieves a consistently good performance for both *V* and *CV* cases, unlike some other classifiers in Table 6.5 for which a dip in performance was noted when cross-validating the results. In case of the Breast cancer results shown in Table 7, LR-LASSO achieves the highest accuracy of 86.84% (F1-score=0.86) for the validation case, but in cross-validation, SMOTE-LASSO-DeepNet achieves the highest accuracy of 91.46% (F1-score=0.92).

As an overall observation, we can say that classifiers other than the proposed method, especially SVM-LASSO and LR-LASSO have also performed well, but their performance was not consistent across all four datasets. Our proposed method SMOTE-LASSO-DeepNet performs consistently best in case of all four datasets for both *V* and *CV* cases. Apart from the proposed method, we also study two DeepNet variations of our framework (same architecture of DeepNet is maintained): LASSO-SMOTE-DeepNet and LASSO-DeepNet; we find from Tables 6.2 to 6.5 that their performances were not at par with that of the proposed SMOTE-LASSO-DeepNet framework.

6.3. GFMM & EFMM RESULTS:

This dataset has 203 samples and 12,600 features (genes). There are five classes indicating five subtypes of lung cancer[38].The class distribution is highly imbalanced. The different cancer subtypes and their class populations are: lung adenocarcinomas (139), squamous cell lung carcinomas (21), lung carcinoids (20), small cell lung carcinomas (6), and normal samples (17). Under such a scenario, defining accurate class boundaries is an obvious challenge. We propose to counter this challenge using the FMM classifiers: GFMM and EFMM. We first normalize the dataset using min–max normalization and the range of the normalization is set to [0, 1]. After obtaining the train and test sets, we extract the selected features by implementing LASSO on the train set. The reduced feature set is used to train the model and compute the test accuracy.

6.3.1. GFMM Results

The General Fuzzy min–max model hyperbox visualization after training is complete is shown in fig 6.3. The five colors indicate the five classes. As observed from the hyperbox visualization in fig 6.3, the majority class namely, lung adenocarcinoma, is segregated well from the other classes which indicate a good classification performance. The minority classes are also distinctly separated from each other. For the GFMM learning algorithm, the value we choose for the hyper-box expansion coefficient is 0.5 and the sensitivity value is 1. The classification results are shown in Table 6.6 (accuracy) and 6.7 (execution time) for all methods. General Fuzzy min–max classifier gives the best result among all the classifiers, as observed from Table 6.6. The accuracy achieved with LASSO is 98.04% and 94.06% for validation and cross-validation, respectively, and this is the best among all the classifiers that we have used for this microarray dataset. For GFMM, we observe that the execution time of the classification process in case of selected features is 4.57s (with hyperbox visualization) which is faster as compared to all the other fuzzy models. The reason for the accurately defined class boundaries of GFMM is the simple operations involved for which even the few samples in the training set is sufficient, thereby reducing the overlap between classes to a great extent.

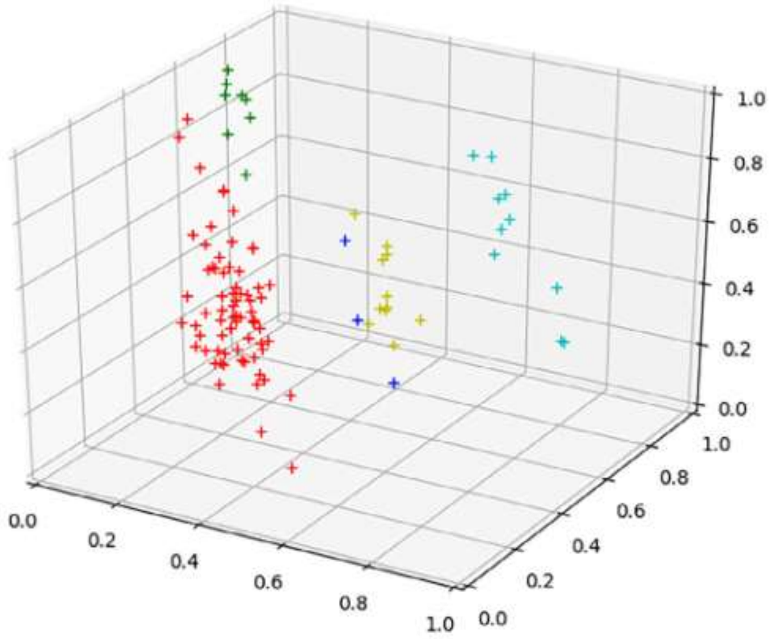


Fig 6.3. GFMM hyperbox constructed after training is complete.

6.3.2. EFMM Results

The enhanced Fuzzy min–max classifier does not give good results for the small sample dataset in our experiments. The accuracy we achieved with LASSO is 90.2% and 93.07% for Validation and cross-validation, respectively, as observed from Table 6.6.

The EFMM hyperbox visualization is shown in Figure 6.4, that is, obtained after the training process is complete. Comparing the hyperbox visualizations of GFMM in Figure 6.3 with that of EFMM in Figure 6.4, we observe a better segregation of classes in case of GFMM in Figure 6.3, indicating that the fuzzy membership functions for the five classes are more well-defined in case of GFMM. The hyperbox visualization shows some degree of overlap between the majority class and few minority classes, and also among the minority classes. This implies that EFMM is incapable of learning from small sample datasets such as gene expression datasets due to the intricate learning procedures involved that require sufficient samples to learn from.

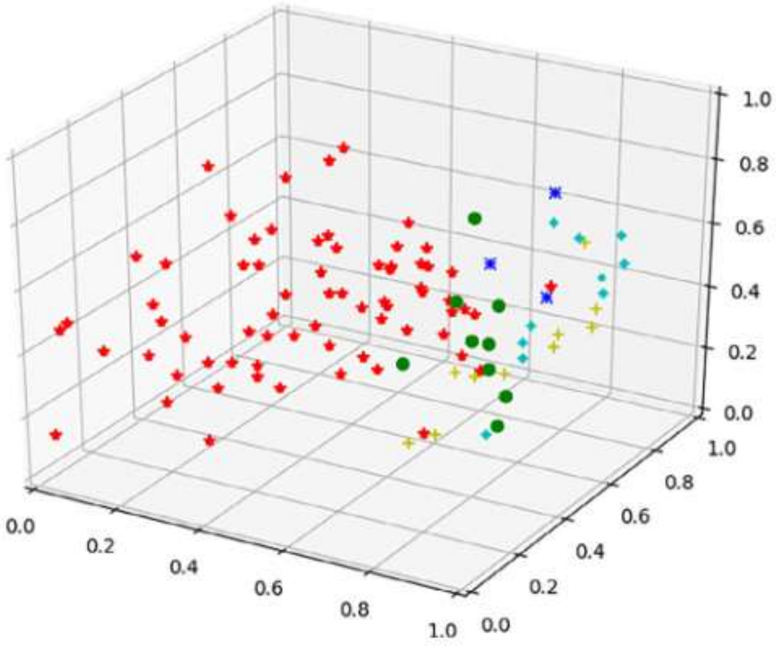


Fig 6.4. EFMM hyperbox constructed after training is complete.

6.3.3. Results comparison

In Table 6.6 we can see the test accuracy comparison of all the classification algorithms. Other than GFMM and EFMM, we have compared the results to the machine learning models Support Vector Machine (SVM), K-Nearest Neighbor, Logistic Regression, Naïve Bayes and Random Forest and to some of the existing works on cancer gene expression classification:

Algorithms	Accuracy (in %)	
	Validation(V)	Cross-Validation
General Fuzzy Min-Max (Ours)	98.04	94.06
Enhanced Fuzzy Min-Max (Ours)	90.2	93.07
Fuzzy Min-Max	90.19	93
SMOTE and DNN	95.68	94.06
K-Nearest Neighbor	94.11	89.1
Logistic Regression	95	89.1
Naive Bayes	94.11	84.15
Random Forest	91.76	85.34
Support Vector Machine (SVM)	93.13	92.07
1D-CNN [55]	94.11	90.09
DNN[56]	94.31	89.9

Table 6.6 Test accuracies of different models for lung cancer classification with LASSO as feature selector.

Algorithms	Execution Time (in seconds)	
	With LASSO	Without LASSO
General Fuzzy Min-Max	4.57	87.52
Enhanced Fuzzy Min-Max	40.21	41.54
Fuzzy Min-Max	13.77	963.31
SMOTE, LASSO, and DNN	7.03	-
K-Nearest Neighbor	0.37	0.37
Logistic Regression	0.66	8.51
Naive Bayes	0.32	1.39
Random Forest	1.98	2.90
Support Vector Machine (SVM)	0.40	17.74
1D-CNN	5.93	325
DNN	6.12	-

Table 6.7 Average execution time of classification algorithms

FMM, 1D-CNN, DNN, SMOTE and DNN. For the hyperparameter settings of 11,12,14,30 we have referred to the original articles. LASSO is applied for feature selection in all cases. Out of the 12,600 features, 98 features are extracted during validation, and 95 features are extracted during cross-validation, LASSO being applied only on the training set in each case. LASSO thus results in a reduced gene pool which is applied for training purpose. From these results, we analyze that GFMM stands out among all the algorithms in terms of the accuracy obtained (Validation accuracy=98.04%, Cross-validation accuracy=94.06%). The second-best performance is that of SMOTE, LASSO, and DNN [57] (Validation accuracy=95.68%, Cross-validation accuracy = 94.06%). Figure 5.5 shows the comparison of the performance of GFMM and EFMM with and without LASSO. The application of LASSO creates a reduced and optimized gene pool, hence the system is faster and performance is boosted, as observed from Figure5.5 which compares the test accuracy scores. We note the following observations from Figure5.5 A, B showing the validation and cross-validation accuracies, respectively.

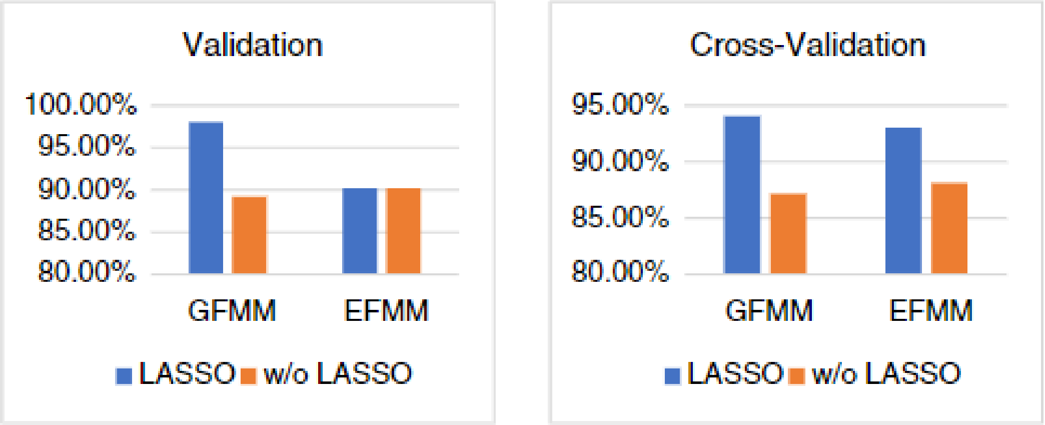


Fig 6.5 Performance of GFMM and EFMM models with and without LASSO

i. Feature selection by LASSO significantly boosts the performance of the FMM models, especially GFMM, due to the inclusion of the most informative features in the selected gene pool.

ii. In the absence of feature selection, EFMM marginally outperforms GFMM, signifying that for efficient operation of GFMM, informative features are required.

From the execution times of all models summarized in Table 6.7, we make a general observation that FMM-based methods take more time to execute as compared to other machine learning algorithms. GFMM is the fastest with feature selection as compared to other fuzzy algorithms, and EFMM takes approximately same time for both with and without feature selection. The execution time of GFMM is 4.57s with online or progressive hyperbox visualization and around0.32s without online or progressive hyperbox visualization [58].

After analyzing all the comparison methods, we can say that the General Fuzzy min–max classifier is a very suitable option for the classification of microarray data, and it performs the best when used with feature selection for selecting the most optimal gene set that identifies the cancer subtype accurately.

CONCLUSION

In this thesis, we have presented three distinct approaches for the classification of microarray gene expression data. Each approach employs different techniques and algorithms to address the challenges of small sample sizes and high-dimensional datasets.

The initial approach synergizes the Fuzzy Min-Max (FMM) classifier with the LASSO feature selection technique to effectively decrease dataset dimensions while preserving relevant gene information. This method outperforms alternative classifiers, showcasing its effectiveness in accuracy and F1-score. Future research will prioritize investigating more efficient computational architectures for FMM to improve the efficiency of the automated cancer diagnosis system.

In the second approach, we introduce a deep learning framework called SMOTE-LASSO-DeepNet. This framework utilizes SMOTE for minority class augmentation and LASSO for feature selection, followed by training a DeepNet model. The results show that our method consistently outperforms other approaches across multiple benchmark cancer datasets. As a future extension, we plan to investigate deep ensemble frameworks for further improving the classification of gene expression data.

Finally, we explore the application of neuro-fuzzy systems, specifically the GFMM and EFMM models, for cancer diagnosis from gene expression data. Our experiments reveal that GFMM is well-suited for classifying small sample datasets due to its efficient computation of fuzzy memberships. GFMM, in conjunction with LASSO, achieves outstanding accuracy and execution time compared to other machine learning algorithms. Additionally, GFMM outperforms existing works on cancer gene expression classification. The hyperbox visualizations and decision boundaries of GFMM demonstrate its superior performance in accurately representing the different classes.

Collectively, these approaches contribute to the advancement of cancer diagnosis using gene expression data. The findings highlight the potential of neuro-fuzzy systems, deep learning frameworks, and their combinations with feature selection techniques for classifying small sample, high-dimensional gene expression datasets. Future research directions include further customization of neuro-fuzzy models to incorporate feature selection and imbalance treatment, aiming to develop more efficient and accurate cancer classification systems.

REFERENCES

[1] Schena, Mark, Dari Shalon, Ronald W. Davis, and Patrick O. Brown. "Quantitative monitoring of gene expression patterns with a complementary DNA microarray." *Science* 270, no. 5235 (1995): 467-470.

[2] Brazma, Alvis, Pascal Hingamp, John Quackenbush, Gavin Sherlock, Paul Spellman, Chris Stoeckert, John Aach et al. "Minimum information about a microarray experiment (MIAME)—toward standards for microarray data." *Nature genetics* 29, no. 4 (2001): 365-371.

[3] Varadhachary, Gauri, and James L. Abbruzzese. "Carcinoma of unknown primary." In *Abeloff's Clinical oncology*, pp. 1694-1702. Elsevier, 2020.

[4] Torre, Lindsey A., Britton Trabert, Carol E. DeSantis, Kimberly D. Miller, Goli Samimi, Carolyn D. Runowicz, Mia M. Gaudet, Ahmedin Jemal, and Rebecca L. Siegel. "Ovarian cancer statistics, 2018." CA: a cancer journal for clinicians 68, no. 4 (2018): 284-296.

[5] Lu, Ying, and Jiawei Han. "Cancer classification using gene expression data." Information Systems 28, no. 4 (2003): 243-268.

[6] Susan, Seba, and Amitesh Kumar. "The balancing trick: Optimized sampling of imbalanced datasets—A brief survey of the recent State of the Art." *Engineering Reports* 3, no. 4 (2021): e12298

[7] Wang, Lipo, Yaoli Wang, and Qing Chang. "Feature selection methods for big data bioinformatics: a survey from the search perspective." Methods 111 (2016): 21-31.

[8] Jain, Anmol, Aishwary Kumar, and Seba Susan. "Evaluating Deep Neural Network Ensembles by Majority Voting Cum Meta-Learning Scheme." In Soft Computing and Signal Processing, pp. 29-37. Springer, Singapore, 2022.

[9] Susan, Seba, and Jatin Malhotra. "Recognising devanagari script by deep structure learning of image quadrants." DESIDOC Journal of Library & Information Technology 40, no. 5 (2020): 268-271.

[10] S. Mitra and Y. Hayashi, "Bioinformatics with soft computing," IEEE Trans. Systems, Man and Cybernetics, Part C, vol.36, pp.616 -635, 2006.

[11] Chen, Yifei, Yi Li, Rajiv Narayan, Aravind Subramanian, and Xiaohui Xie. "Gene expression inference with deep learning." Bioinformatics 32, no. 12 (2016): 1832-1839.

[12] Tabares-Soto, Reinel, Simon Orozco-Arias, Victor Romero-Cano, Vanesa Segovia Bucheli, José Luis Rodríguez-Sotelo, and Cristian Felipe Jiménez-Varón. "A comparative study of machine learning and deep learning algorithms to classify cancer types based on microarray gene expression data." PeerJ Computer Science 6 (2020): e270.

[13] Lyu, Boyu, and Anamul Haque. "Deep learning based tumor type classification using gene expression data." In Proceedings of the 2018 ACM international conference on bioinformatics, computational biology, and health informatics, pp. 89-96. 2018.

[14] Guillen, Pablo, and Jerry Ebalunode. "Cancer classification based on microarray gene expression data using deep learning." In *2016 International Conference on Computational Science and Computational Intelligence (CSCI)*, pp. 1403-1405. IEEE, 2016.

[15] Mohammed, Mohanad, Henry Mwambi, Innocent B. Mboya, Murtada K. Elbashir, and Bernard Omolo. "A stacking ensemble deep learning approach to cancer type classification based on TCGA data." Scientific reports 11, no. 1 (2021): 1-22.

[16] Mostavi, Milad, Yu-Chiao Chiu, Yufei Huang, and Yidong Chen. "Convolutional neural network models for cancer type prediction based on gene expression." BMC medical genomics 13, no. 5 (2020): 1-13.

[17] Abd-Elnaby, Muhammed, Marco Alfonse, and Mohamed Roushdy. "Classification of breast cancer using microarray gene expression data: a survey." *Journal of Biomedical Informatics* 117 (2021): 103764.

[18] Shahbeig, Saleh, Akbar Rahideh, Mohammad Sadegh Helfroush, and Kamran Kazemi. "Gene selection from large-scale gene expression data based on fuzzy interactive multi-objective binary optimization for medical diagnosis." *Biocybernetics and Biomedical Engineering* 38, no. 2 (2018): 313-328.

[19] Hu, Hong, Jiuyong Li, Ashley Plank, Hua Wang, and Grant Daggard. "A comparative study of classification methods for microarray data analysis." In *Proceedings of the 5th Australasian Data Mining Conference (AusDM 2006): Data Mining and Analytics 2006*, pp. 33-37. ACS Press, 2006.

[20] Lee, Jae Won, Jung Bok Lee, Mira Park, and Seuck Heun Song. "An extensive comparison of recent classification tools applied to microarray data." *Computational Statistics & Data Analysis* 48, no. 4 (2005): 869-885.

[21] Arunkumar, C., and S. Ramakrishnan. "Attribute selection using fuzzy roughset based customized similarity measure for lung cancer microarray gene

expression data." *Future Computing and Informatics Journal* 3, no. 1 (2018): 131-142.

[22] Susan, Seba, and Srishti Sharma. "A fuzzy nearest neighbor classifier for speaker identification." In *2012 fourth international conference on computational intelligence and communication networks*, pp. 842-845. IEEE, 2012.

[23] Susan, Seba, and Seema Chandna. "Object recognition from color images by fuzzy classification of gabor wavelet features." In *2013 5th International Conference and Computational Intelligence and Communication Networks*, pp. 301-305. IEEE, 2013.

[24] Ohno-Machado, Lucila, Staal Vinterbo, and Griffin Weber. "Classification of gene expression data using fuzzy logic." *Journal of Intelligent & Fuzzy Systems* 12, no. 1 (2002): 19-24.

[25] Chu, Feng, and Lipo Wang. "Applications of support vector machines to cancer classification with microarray data." International journal of neural systems 15, no. 06 (2005): 475-484.

[26] Wang, Lipo, Feng Chu, and Wei Xie. "Accurate cancer classification using expressions of very few genes." IEEE/ACM Transactions on computational biology and bioinformatics 4, no. 1 (2007): 40-53.

[27] Tibshirani, R. J. "Regression shrinkage and selection via the lasso." (2011): 273-282.

[28] Hira, Zena M., and Duncan F. Gillies. "A review of feature selection and feature extraction methods applied on microarray data." *Advances in bioinformatics* 2015 (2015).

[29] GÜÇKIRAN, Kıvanç, İsmail Cantürk, and Lale ÖZYILMAZ. "DNA microarray gene expression data classification using SVM, MLP, and RF with feature selection methods relief and LASSO." *Süleyman Demirel Üniversitesi Fen Bilimleri Enstitüsü Dergisi* 23, no. 1 (2019): 126-132.

[30] Susan, Seba, and Madasu Hanmandlu. "Smaller feature subset selection for real-world datasets using a new mutual information with Gaussian gain." *Multidimensional Systems and Signal Processing* 30, no. 3 (2019): 1469-1488.

[31] Urda, Daniel, Julio Montes-Torres, Fernando Moreno, Leonardo Franco, and José M. Jerez. "Deep learning to analyze RNA-seq gene expression data." In *International work-conference on artificial neural networks,* pp. 50-59. Springer, Cham, 2017.

[32] Blagus, Rok, and Lara Lusa. "SMOTE for high-dimensional class-imbalanced data." *BMC bioinformatics* 14, no. 1 (2013): 1-16.

[33] Hamzeh, Osama, Abedalrhman Alkhateeb, Julia Zheng, Srinath Kandalam, and Luis Rueda. "Prediction of tumor location in prostate cancer tissue using a machine learning system on gene expression data." *BMC bioinformatics* 21, no. 2 (2020): 1-10.

[34] Roy, Shikha, Rakesh Kumar, Vaibhav Mittal, and Dinesh Gupta. "Classification models for Invasive Ductal Carcinoma Progression, based on gene expression data-trained supervised machine learning." *Scientific reports* 10, no. 1 (2020): 1-15

[35] Simpson, Patrick K. "Fuzzy Min—MaX Neural NetWorks—Part 1: Classification." *IEEE Trans. on Neural Networks* 3, no. 5 (1992): 776-786.

[36] Grisci, Bruno Iochins, Bruno César Feltes, and Marcio Dorn. "Neuroevolution as a tool for microarray gene expression pattern identification in cancer research." *Journal of biomedical informatics* 89 (2019): 122-133.

[37] Feltes, Bruno Cesar, Eduardo Bassani Chandelier, Bruno Iochins Grisci, and Marcio Dorn. "Cumida: an extensively curated microarray database for benchmarking and testing of machine learning approaches in cancer research." *Journal of Computational Biology* 26, no. 4 (2019): 376-386.

[38] Bhattacharjee, Arindam, William G. Richards, Jane Staunton, Cheng Li, Stefano Monti, Priya Vasa, Christine Ladd et al. "Classification of human lung carcinomas by mRNA expression profiling reveals distinct adenocarcinoma subclasses." *Proceedings of the National Academy of Sciences* 98, no. 24 (2001): 13790-13795.

[39] Singh, Yashpal, and Seba Susan. "Optimal Gene Selection and Classification of Microarray Data Using Fuzzy Min-Max Neural Network with LASSO." In *International Conference on Intelligent and Fuzzy Systems*, pp. 777-784. Springer, Cham, 2022.

[40] Vapnik, Vladimir. *The nature of statistical learning theory*. Springer science & business media, 1999.

[41] Breiman, Leo. "Random forests." *Machine learning* 45, no. 1 (2001): 5-32.

[42] Peng, Chao-Ying Joanne, Kuk Lida Lee, and Gary M. Ingersoll. "An introduction to logistic regression analysis and reporting." *The journal of educational research* 96, no. 1 (2002): 3-14.

[43] Guo, Gongde, Hui Wang, David Bell, Yaxin Bi, and Kieran Greer. "KNN model-based approach in classification." In *OTM Confederated International Conferences" On the Move to Meaningful Internet Systems"*, pp. 986-996. Springer, Berlin, Heidelberg, 2003.

[44] Wickramasinghe, Indika, and Harsha Kalutarage. "Naive Bayes: applications, variations and vulnerabilities: a review of literature with code snippets for implementation." *Soft Computing* 25, no. 3 (2021): 2277-2293.

[45] Susan, Seba, and Jatin Malhotra. "Learning interpretable hidden state structures for handwritten numeral recognition." In *2020 4th International Conference on Computational Intelligence and Networks (CINE)*, pp. 1-6. IEEE, 2020.

[46] Saini, Manisha, and Seba Susan. "Deep transfer with minority data augmentation for imbalanced breast cancer dataset." *Applied Soft Computing* 97 (2020): 106759.

[47] Gabrys, Bogdan, and Andrzej Bargiela. "General fuzzy min-max neural network for clustering and classification." *IEEE transactions on neural networks* 11, no. 3 (2000): 769-783.

[48] Mohammed, Mohammed Falah, and Chee Peng Lim. "An enhanced fuzzy min–max neural network for pattern classification." *IEEE transactions on neural networks and learning systems* 26, no. 3 (2014): 417-429.

[49] Brown, Michael PS, William Noble Grundy, David Lin, Nello Cristianini, Charles Walsh Sugnet, Terrence S. Furey, Manuel Ares Jr, and David Haussler. "Knowledge-based analysis of microarray gene expression data by using support vector machines." *Proceedings of the National Academy of Sciences* 97, no. 1 (2000): 262-267.

[50] Ram, Malihe, Ali Najafi, and Mohammad Taghi Shakeri. "Classification and biomarker genes selection for cancer gene expression data using random forest." *Iranian journal of pathology* 12, no. 4 (2017): 339.

[51] Sartor, Maureen A., George D. Leikauf, and Mario Medvedovic. "LRpath: a logistic regression approach for identifying enriched biological groups in gene expression data." *Bioinformatics* 25, no. 2 (2009): 211-217.

[52] Chandra, B., and Manish Gupta. "Robust approach for estimating probabilities in Naïve–Bayes Classifier for gene expression data." *Expert Systems with Applications* 38, no. 3 (2011): 1293-1298.

[53] Khan, Javed, Jun S. Wei, Markus Ringner, Lao H. Saal, Marc Ladanyi, Frank Westermann, Frank Berthold et al. "Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks." *Nature medicine* 7, no. 6 (2001): 673-679.

[54] Ahmed O, Brifcani A. Gene expression classification based on deep learning.In 2019 4th Scientific International Conference Najaf (SICN).IEEE; 2019:145-149.

[55] Mohammed, Mohanad, Henry Mwambi, Innocent B. Mboya, Murtada K. Elbashir, and Bernard Omolo. "A stacking ensemble deep learning approach to cancer type classification based on TCGA data." *Scientific reports* 11, no. 1 (2021): 1-22.

[56] Urda, Daniel, Julio Montes-Torres, Fernando Moreno, Leonardo Franco, and José M. Jerez. "Deep learning to analyze RNA-seq gene expression data." In *Advances in Computational Intelligence: 14th International Work-Conference on Artificial Neural Networks, IWANN 2017, Cadiz, Spain, June 14-16, 2017, Proceedings, Part II 14*, pp. 50-59. Springer International Publishing, 2017.

[57] Singh, Yashpal, and Seba Susan. "SMOTE-LASSO-DeepNet Framework for Cancer Subtyping from Gene Expression Data." In *2022 15th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, pp. 1-6. IEEE, 2022.

[58] Singh, Y., & Susan, S. (2023). “Lung cancer subtyping from gene expression data using general and enhanced Fuzzy min-max neural networks”. *Engineering Reports, e12663. doi: 10.1002/eng2.126634.*

LIST OF PUBLICATIONS

1. Singh, Yashpal, and Seba Susan. "Optimal Gene Selection and Classification of Microarray Data Using Fuzzy Min-Max Neural Network with LASSO." In *International Conference on Intelligent and Fuzzy Systems*, pp. 777-784. Springer, Cham, 2022.

2. Singh, Yashpal, and Seba Susan. “SMOTE-LASSO-DeepNet Framework for Cancer Subtyping from Gene Expression Data.” In *2022 15th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, IEEE, 2022.

3. Singh, Y., & Susan, S. (2023). “Lung cancer subtyping from gene expression data using general and enhanced Fuzzy min-max neural networks”. *Engineering Reports, e12663. doi: 10.1002/eng2.126634.*

PAPER NAME

CANCER DIAGNOSIS FROM GENE EXPRE SSION DATA USING FUZZY CLASSIFIERS AND DEEP LEARNING

AUTHOR

Seba Susan

WORD COUNT

9682 Words

CHARACTER COUNT

53066 Characters

PAGE COUNT

40 Pages

FILE SIZE

1.2MB

SUBMISSION DATE

May 26, 2023 3:53 PM GMT+5:30

REPORT DATE

May 26, 2023 3:54 PM GMT+5:30

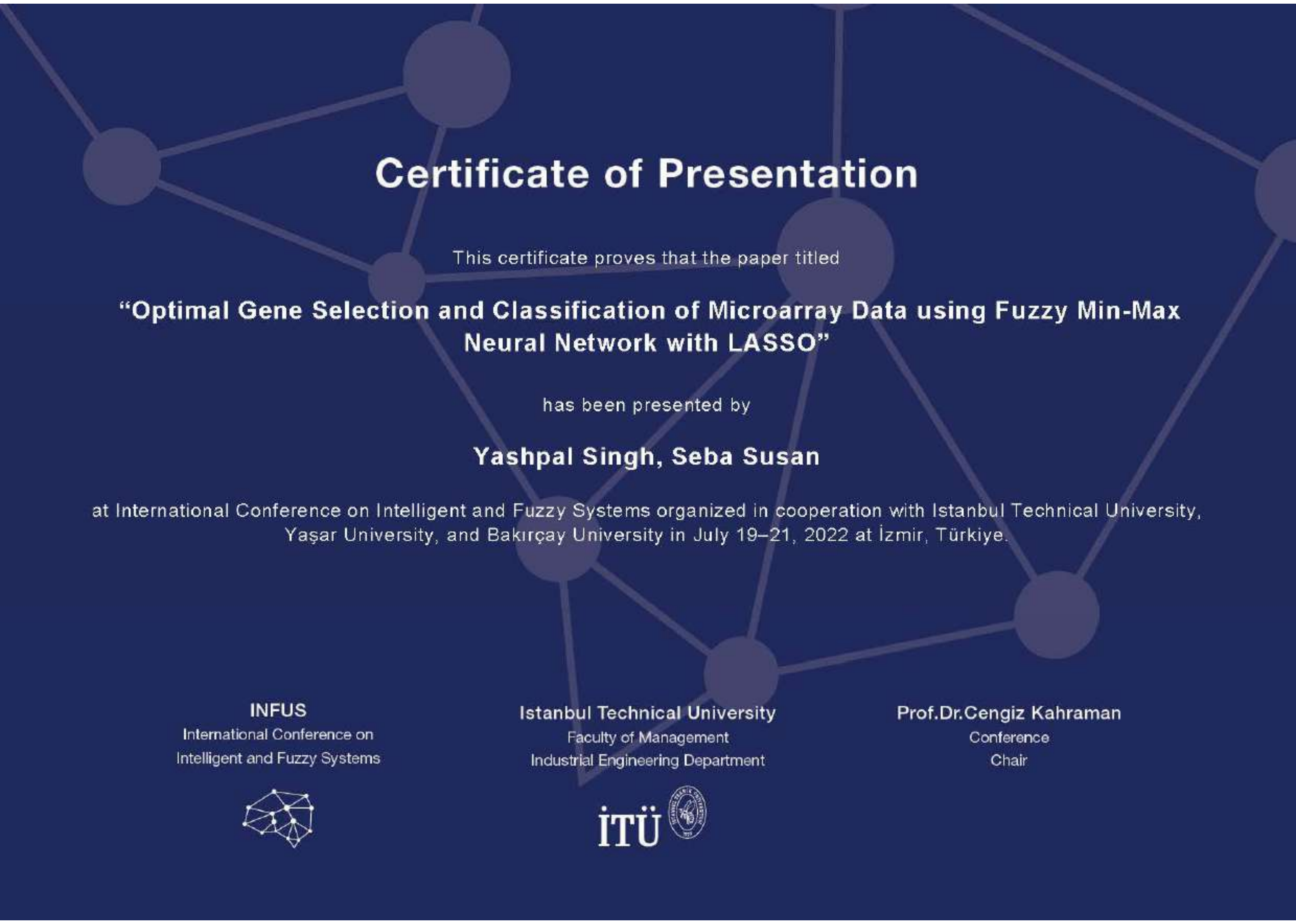
● 6% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

- 2% Internet database
- Crossref database
- 4% Submitted Works database
- 4% Publications database
- Crossref Posted Content database

● Excluded from Similarity Report

- Bibliographic material
- Manually excluded sources
- Small Matches (Less then 10 words)



Certificate of Presentation

This certificate proves that the paper titled

"Optimal Gene Selection and Classification of Microarray Data using Fuzzy Min-Max Neural Network with LASSO"

has been presented by

Yashpal Singh, Seba Susan

at International Conference on Intelligent and Fuzzy Systems organized in cooperation with Istanbul Technical University, Yaşar University, and Bakırçay University in July 19–21, 2022 at İzmir, Türkiye.

INFUS
International Conference on
Intelligent and Fuzzy Systems

Istanbul Technical University
Faculty of Management
Industrial Engineering Department

İTÜ

Prof.Dr.Cengiz Kahraman
Conference
Chair

CISP-BMEI
2022

CERTIFICATE

★ for Oral Presentation ★

Speaker:
Yashpal Singh

Authors: Yashpal Singh and Seba Susan

Paper Title: SMOTE-LASSO-DeepNet Framework for Cancer Subtyping from Gene Expression Data

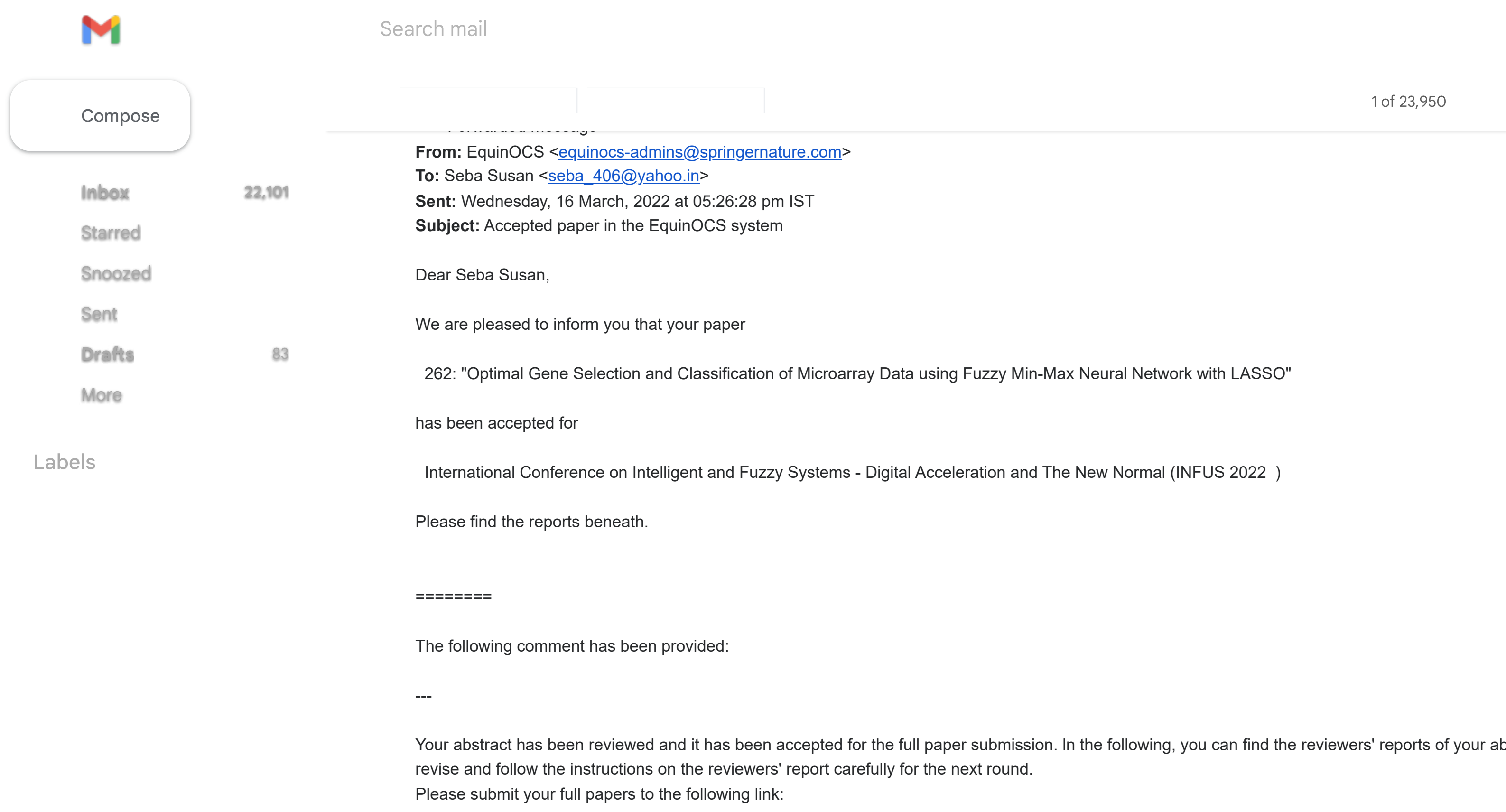
For your excellent oral presentation in the 15th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI 2022) held in Beijing, China, November 05-07, 2022.

General Chair: Lin Cao

Conference Committee
CISP-BMEI 2022

Image and Signal Processing BioMedical Engineering and Informatics Committe ★
CISP-BMEI
Regn. No.
#5[illegible]27[illegible]

IEEE EMB



Forwarded message

From: EquinOCS <equinocs-admins@springernature.com>
To: Seba Susan <seba_406@yahoo.in>
Sent: Wednesday, 16 March, 2022 at 05:26:28 pm IST
Subject: Accepted paper in the EquinOCS system

Dear Seba Susan,

We are pleased to inform you that your paper

262: "Optimal Gene Selection and Classification of Microarray Data using Fuzzy Min-Max Neural Network with LASSO"

has been accepted for

International Conference on Intelligent and Fuzzy Systems - Digital Acceleration and The New Normal (INFUS 2022)

Please find the reports beneath.

========

The following comment has been provided:

Your abstract has been reviewed and it has been accepted for the full paper submission. In the following, you can find the reviewers' reports of your ab
revise and follow the instructions on the reviewers' report carefully for the next round.
Please submit your full papers to the following link:

is:starred

Compose

Inbox 22,101

Starred

Snoozed

Sent

Drafts 83

More

Labels

12 of 130

Fw: CISP-BMEI 2022 P1118 Acceptance Notification

Inbox ×

Seba Susan <seba_406@yahoo.in>
to me

Sat, Aug 27, 2022, 9:15 PM

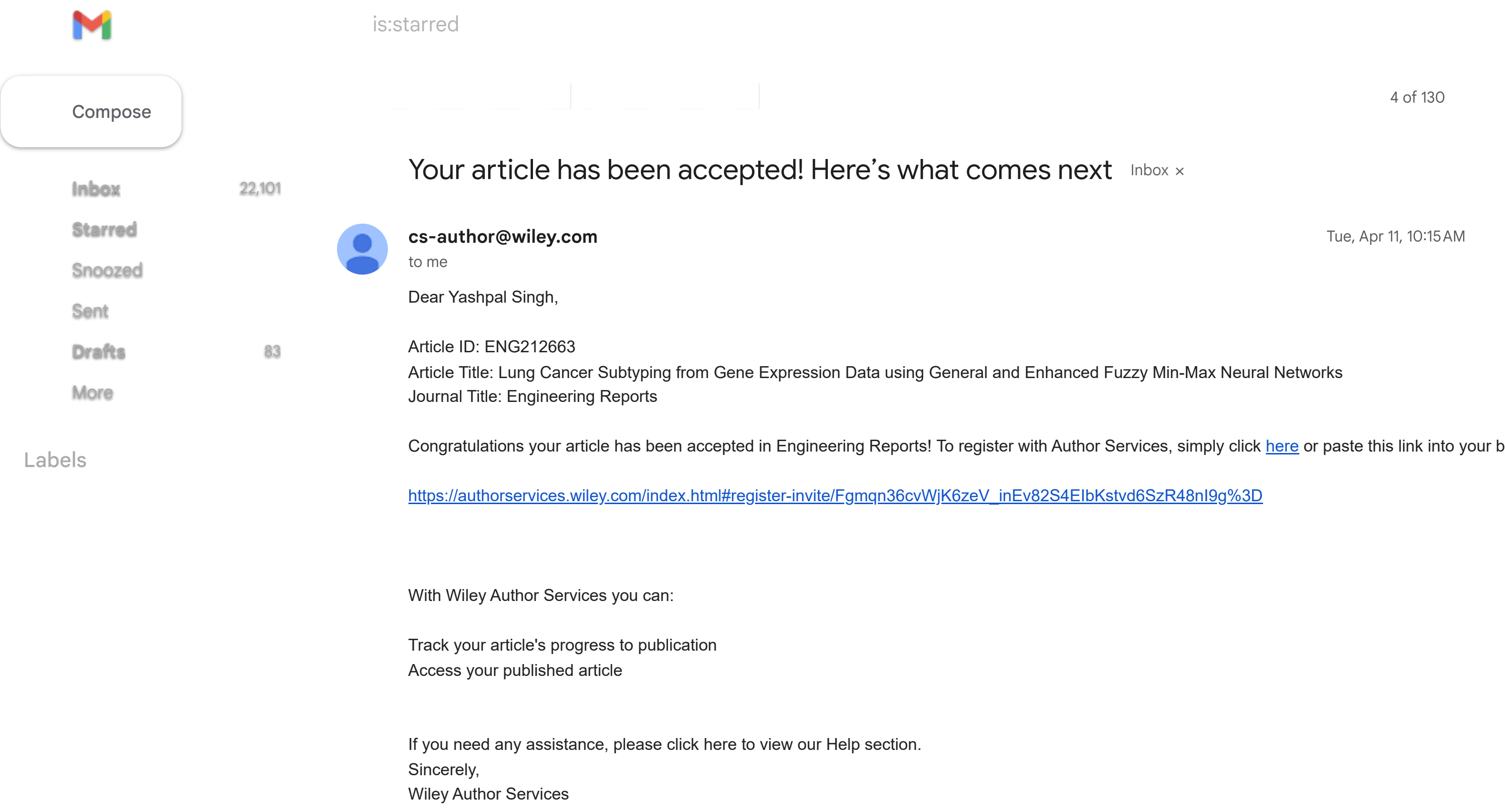
----- Forwarded message -----
From: Conference Committee <committee@bistu.edu.cn>
To: "seba_406@yahoo.in" <seba_406@yahoo.in>
Sent: Thursday, 18 August, 2022 at 08:03:18 pm IST
Subject: CISP-BMEI 2022 P1118 Acceptance Notification

Dear Seba Susan,

Paper ID : P1118
Paper Title : SMOTE-LASSO-DeepNet Framework for Cancer Subtyping from Gene Expression Data

(All Chinese characters in this email are intended for authors from China's mainland only. 内地作者请见下面中文录用通知。)

Congratulations! We are pleased to inform you that your above paper has been accepted for presentation at **the 2022 15th IEEE EMBS Regional Co**
Image and Signal Processing, BioMedical Engineering and Informatics (**CISP-BMEI 2022**) from 22-24 October 2022 in Beijing, China, subject to
below. After you complete the requirements below, your paper will appear in conference proceedings and will be submitted to EI Compendex, SCOPU
ISTP(CPCI), as well as the IEEE Xplore (CISP-BMEI 2008-2021 have all been indexed in EI Compendex etc.). Extended versions of best papers will a
CISP-BMEI special issue of an SCI-indexed journal. Only registered papers will be considered for the SCI journal special issue and only the selected a
notified by 30 October 2022.



Your article has been accepted! Here’s what comes next

cs-author@wiley.com

to me

Tue, Apr 11, 10:15 AM

Dear Yashpal Singh,

Article ID: ENG212663
Article Title: Lung Cancer Subtyping from Gene Expression Data using General and Enhanced Fuzzy Min-Max Neural Networks
Journal Title: Engineering Reports

Congratulations your article has been accepted in Engineering Reports! To register with Author Services, simply click here or paste this link into your b

https://authorservices.wiley.com/index.html#register-invite/Fgmqn36cvWjK6zeV_inEv82S4ElbKstvd6SzR48nI9g%3D

With Wiley Author Services you can:

Track your article's progress to publication
Access your published article

If you need any assistance, please click here to view our Help section.
Sincerely,
Wiley Author Services

Scopus Preview

This author profile is generated by Scopus. Learn more

Singh, Yashpal

ⓘ Delhi Technological University, New Delhi, India Ⓢ 57822625000 ⓘ iD Connect to ORCID

Is this you? Connect to Mendeley account

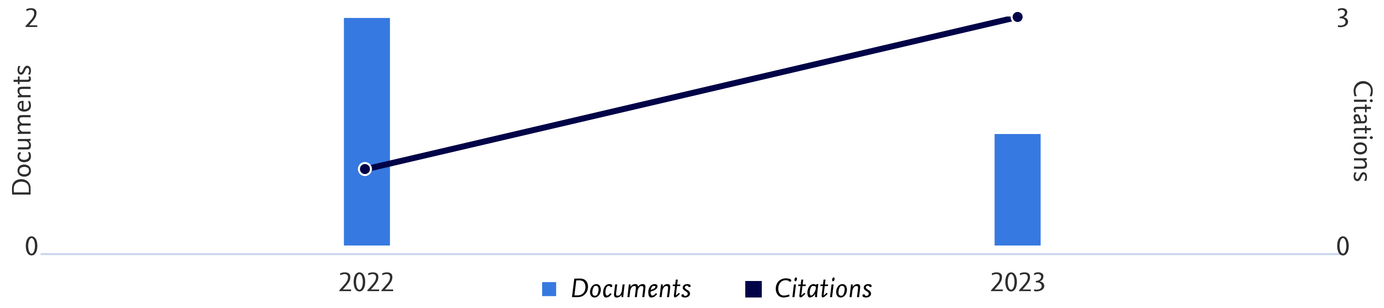
4
Citations by ***3 documents***

3
Documents

2
h-index View *h*-graph

Set alert Edit profile ••• More

Document & citation trends



Scopus Preview

Scopus Preview users can only view a limited set of features. Check your institution's access to view all documents and features.

Check access

3 Documents Cited by 3 documents 0 Preprints 1 Co-Author Topics
Beta
0 Awarded Grants

Note:

Scopus Preview users can only view an author's last 10 documents, while most other features are disabled. Do you have access through your institution? Check your institution's access to view all documents and features.

3 documents

Export all Save all to list

Sort by Date (...

Article • Article in Press • *Open access*

Lung cancer subtyping from gene expression data using general and enhanced Fuzzy min–max neural networks

0

Citations

Singh, Y., Susan, S.

Engineering Reports, 2023

Show abstract Related documents

Conference Paper

SMOTE-LASSO-DeepNet Framework for Cancer Subtyping from Gene Expression Data

2

Citations

Singh, Y., Susan, S.

Proceedings - 2022 15th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics, CISP-BMEI 2022, 2022

Show abstract Related documents

Conference Paper

Optimal Gene Selection and Classification of Microarray Data Using Fuzzy Min-Max Neural Network with LASSO

2

Citations

Singh, Y., Susan, S.

Lecture Notes in Networks and Systems, 2022, 504 LNNS, pp. 777–784

Show abstract Related documents

Back to top

View list in search results format

View references

Set document alert