Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 4, 2021, Pages. 4741 - 4758
Received 05 March 2021; Accepted 01 April 2021.

Binomial Probability Distribution Based Principal Component Analysis with Imperialist Competitive Algorithm for Cancer Subtypes Diagnosis

[1]P. Avila Clemenshia, [2]Dr.B. Mukunthan

[1]Assistant Professor, Department of Computer Science, Nirmala College for Women, (Autonomous), Coimbatore.
[2]Associate Professor, Department of Computer Science, Sri Ramakrishna College of Arts and Science, (Autonomous), Coimbatore

Abstract

In recent days, human health is highly affected by cancer disease. For facilitating cancer therapy and diagnosis, it is highly important to identify cancer subtypes. Gene expression data based cancer subtype classification has keys to address fundamental issues related with drug discovery and cancer diagnosis. The research work is designed previously using an Artificial Bee Colony (ABC) with Deep Fuzzy Flexible Neural Forest (DFFNForest) approach for classifying cancer subtype. However, high-dimensionality challenge may rise when using the data directly to the classification of the cancer subtypes between samples. To solve this problem the proposed work is designed using a Binomial probability Distribution based Principal Component Analysis (BDPCA). The proposed cancer diagnosis methodology consists of the various stages like dimension reduction, feature selection and classification. Initially, gene expression datasets are taken as input. In first stage, dimensionality reduction is performed by using the Binomial probability Distribution based Principal Component Analysis (BDPCA). In the second stage, feature selection will perform by using Imperialist Competitive Algorithm (ICA) algorithm for reducing classifier's miss rate. Then selected feature will be implemented in Deep Fuzzy Flexible Neural Forest (DFFNForest). In DFFNForest, fuzzy is used to update the weight values of the classifier in the cancer subtype prediction. With respect to error, f-measure, recall, precision and accuracy, better performance is achieved by proposed system when compared to available systems as demonstrated in experimental results.

Keywords: Cancer subtype, Binomial probability Distribution based Principal Component Analysis (BDPCA), Imperialist Competitive Algorithm (ICA) and gene expression data

1. INTRODUCTION

Cancer research has undergone a steady transformation over the last few decades. Significant volumes of cancer evidence have been obtained and are accessible to the medical scientific community as a result of the use of emerging developments in the area of medicine. Scientists used various approaches, such as early-stage screening, to detect cancer forms before they cause symptoms [1]. Early detection and prognosis of a cancer subtype, on the other hand, has become a priority in cancer treatment because it can help with patient clinical management [2].

Accurate prediction of various cancer types will aid in improved detection and toxicity reduction for patients [3]. Microarray technology has allowed researchers to investigate the expression profiles of a wide number of genes under a variety of experimental conditions. The use of microarray-based gene expression profiling to forecast various cancer subtypes has shown considerable promise.

Tumor subtypes are identified using existing classification systems such as the World Health Organization's International Classification of Diseases, which includes codes to identify diseases as well as a broad range of signs, symptoms, adverse observations, grievances, social conditions, and external causes of injury or disease.

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 4, 2021, Pages. 4741 - 4758
Received 05 March 2021; Accepted 01 April 2021.

Different classification approaches from statistical and machine learning field have been extended to cancer classification, but there are several problems that make it a nontrivial challenge. The gene expression data were somewhat different from all of the data these approaches have previously worked with. To begin with, it has a high dimensionality, with thousands to tens of thousands of genes. Second, the amount of publicly accessible data is very limited, with much of it falling below 100.

Thirdly, for cancer distinction, majority of these genes are irrelevant. These genes are not effectively as well as efficiently handled in available classification techniques like Support Vector Machine(SVM)[4], Genetic Algorithm (GA), Fuzzy logic based, Rough set, Artificial Neural Network (ANN), Nearest Neighbour, Decision tree, Rule based and Bayesian Networks.

In some techniques, prior to classification of cancer, gene selection is done. Data size can be minimized using this gene selection and runtime is enhanced using this. Large amount of irrelevant genes which are minimizing classification accuracy are removed using this gene selection technique.

This paper is structured as, for cancer subtype classification, various recently used methods are summarized in section2, section 3 presents the proposed technique's details. Used dataset is described in section4 and obtained experimentation results also presented in that section. Proposed research work's contributions are concluded in section 5.

2. LITERATURE REVIEW

In cancer subtype identification, for using complex miRNA-TF-mRNA regulatory network information, a Weighted Similarity Network Fusion (WSNF), technique is proposed by Xu et al (2016). At first, regulatory network is build, where features are represented as nodes like messenger RNAs (mRNAs), Transcription Factors (TFs) and microRNAs (miRNAs). Interaction among features are indicated using edges. From different interatomic databases, retrieved the interactions.

For computing features weights, mRNAs, TFs and miRNAs expression data and network information is used and it represents features importance level. A network fusion technique is integrated with feature weight for clustering patients (samples) and identified cancer subtypes. Better performance is shown by WSNF as shown in experimental results and performance enhancement is achieved through information received from miRNA-TF-mRNA regulatory network [5].

For cancer subtypes classification, a Deep Flexible Neural Forest (DFNForest) model is designed by Xu et al (2019). For every forest, a multi-classification problem is transformed as various binary classification problems in designed DFNForest model, which makes the difference with conventional FNT model. Flexible natural tree model is deepened by exploring DFNForest's cascade structure. Without additional parameters, model's depth is enhanced.

For minimizing gene expression data's dimensionality, neighborhood rough set and fisher's combination is designed in addition with DFNForest model. This combination is used for obtaining high classification performance. With few genes, high accuracy can be produced using gene selection technique on RNA-seq gene expression data as illustrated in experimentation results. For cancer subtypes classification, better performance is shown by proposed DFNForest model [6].

For making cancer types explainable predictions, a new technique called OncoNetExplainer is designed by Karimet al (2019), which is based on Gene Expressions (GE) data. About 9,074 cancer patients genomic data is used in this system and this data cover 33 various cancer types from Pan-Cancer

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 4, 2021, Pages. 4741 - 4758
Received 05 March 2021; Accepted 01 April 2021.

Atlas. VGG16 and Convolutional neural networks (CNN) are trained using this data via guided-gradient class activation maps++, GradCAM++.

Further, for identifying significant biomarkers, a class-specific heat map is generated and with respect to mean absolute impact, feature importance is computed for ranking top genes among all cancer types. At cancer type prediction, high confidence is exhibited by both models as indicated in qualitative and quantitative analyses [7].

For cancer type prediction, a Convolutional Neural Network (CNN) model is presented by Mostavi et al (2020), which is based on gene expression. For classifying non-tumor and tumor samples as normal or its designated cancer type, unstructured gene expression is given as an input to Convolutional Neural Network (CNN) models.

Three CNN models namely, 2D-Hybrid-CNN, 2D-Vanilla-CNNand 1D-CNN are implemented according to various convolution schemes and gene embedding design. On gene expression profiles, models are tested and trained. Excellent prediction accuracy around 93.9–95.0%, is achieved using this designed models [8].

New method for enhancing cancer subtype prediction is designed by Guo et al (2018),where heterogeneous biological networks and multi-sources transcriptome expression data are incorporated. In heterogeneous biological networks, according to regulatory associations, every genome element's multiple expression feature are extracted and in every expression data, between samples, similarities are predicted using a generalized matrix correlation technique.

In multiple-data views, based on various integration weights, similarity information is fused. Samples are clustered as various subtype groups, according to integrated similarity between them. Highly clinically meaningful cancer subtypes are identified using proposed technique as demonstrated in designed system when compared with other available techniques [9].

3. PROPOSED METHODOLOGY

The major contribution of the proposed work is to introduce a dimensionality reduction and classifier for evaluation of the cancer subtypes. The cancer diagnosis system consists of the following steps: dimensionality reduction, feature selection and classification. In the first stage, dimensionality reduction is performed by using the Binomial probability Distribution based Principal Component Analysis (BDPCA). In the second stage, feature selection will perform by using Imperialist Competitive Algorithm (ICA) algorithm for reducing classifier's miss rate. Then, selected feature will be implemented in Deep Fuzzy Flexible Neural Forest (DFFNForest). In DFFNForest, fuzzy is used to update weight values of the classifier in the cancer subtype prediction. The proposed work's flow diagram is illustrated in figure 1.

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 4, 2021, Pages. 4741 - 4758
Received 05 March 2021; Accepted 01 April 2021.

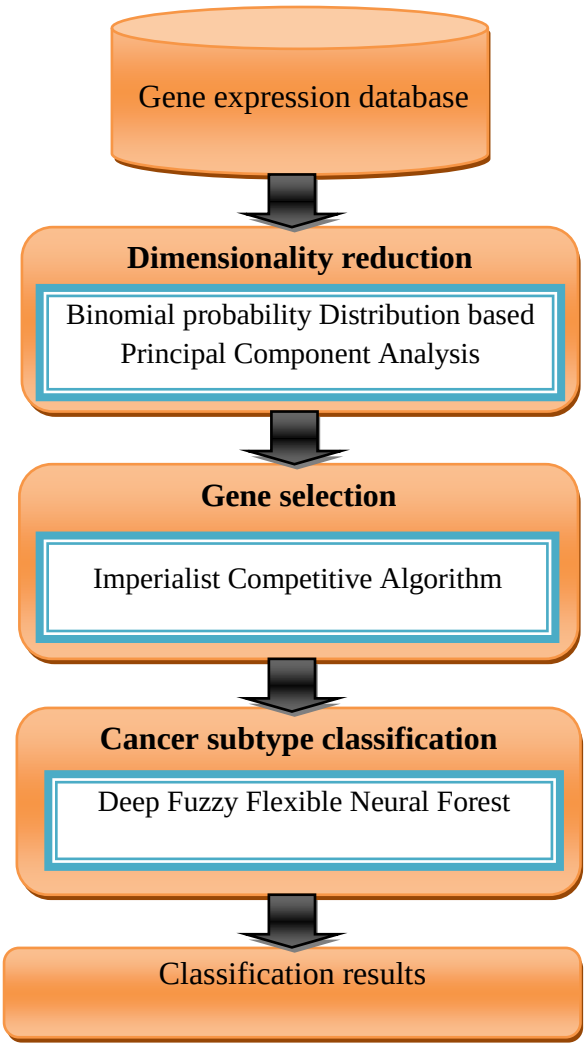

Figure 1: Flow diagram of the proposed work

3.1 Gene expression data

In general, there are huge amount of genes in gene expression data. However, there will be small amount of features available. Only few genes among this large amount of genes is having association with cancer subtypes. Other genes are assumed as noisy or redundant features. So, dimensionality reduction problem corresponds to gene selection where, important genes are selected and it maintains original genes classification accuracy.

3.2 Dimensionality reduction using Binomial probability Distribution based Principal Component Analysis (BDPCA)

High dimensional datasets are compacted spectrally using a popular technique called Principal Component Analysis (PCA) [10-12]. New variables set called principal components are generated using this technique. Original variable's linear combination makes every principal component [13]. In an orthogonal subspace, data is projected by PCA for capturing targeted dataset's variations. Orthogonal subspace has small dimension and highly effective techniques are represented using this.

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 4, 2021, Pages. 4741 - 4758
Received 05 March 2021; Accepted 01 April 2021.

In measured scales, while standardizing original dataset having large difference, it is necessary to avoid key information loss by PCA. A Binomial probability Distribution based Principal component analysis (BDPCA) is designed in this proposed system for solving this. Effective dimensionality reduction can be done using this technique. In a following manner, low-dimensional feature representation problem is stated, assume a n × N data matrix which is represented as X= $(x_1, x_2, \dots, x_n)$, where, feature vector with n dimension is represented as x_i.

Computing the Scatter Matrix

The scatter matrix is computed as:

$$S=\sum_{k=1}^{n}(x_k - m)(x_k - m)^T \qquad (1)$$

Where

m-mean vector

For enhancing dimension reduction probabilities, proposed system designed a binomial probability distribution for mean value computation.

$$m=n\,p \qquad (2)$$

n-denote the number of observations

p-probability of success

Computing eigenvectors and corresponding eigenvalues

While deriving Eigen values from covariance matrix, they are scaled by a factor. Computation of eigenvector-eigenvalue must be checked and it must satisfy the following condition.

$$\sum v = \boldsymbol{\lambda} v \qquad (3)$$

Where,

$$\sum -covariance\ matrix$$

V represents Eigen vector

$\boldsymbol{\lambda}$ represents Eigen value

Regarding data distribution, least information is beard by eigenvectors having low eigenvalues and they needs to be dropped. In general, eigenvectors are ranked to lowest value from highest value based on eigenvalue and top k eigenvectors are selected. In this, two eigenvectors with high eigenvalues are combined for constructing ×k-dimensional eigenvector matrix W .

In final stage, system computed 2×3-dimensional matrix W is used for transforming samples onto new subspace using expression y=W^T×x.

Binomial probability Distribution based Principal Component Analysis (BDPCA)

1. Entire dataset with d-dimensional samples are taken
2. D-dimensional mean vector is computed (i.e., in entire dataset, for every dimension, mean value is computed)
3. Entire dataset's covariance matrix is computed
4. Eigenvectors (e_1,e_2,...,,,,…,e_d) and respective eigenvalues ($\boldsymbol{\lambda}_1, \lambda_2, \dots, \lambda_d$) values are computed
5. Based on decreasing eigenvalues, eigenvectors are sorted and for forming d×k dimensional matrix W, k eigenvectors with high eigenvalues are selected (eigenvector is represented using every column)

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 4, 2021, Pages. 4741 - 4758
Received 05 March 2021; Accepted 01 April 2021.

6. Samples are transformed to new subspace using this d×k eigenvector matrix. Following mathematical expression is used for summarizing this, $y=W^T \times x$, where, x represents d×1-dimensional vector and one sample is represented using this and in new subspace, transformed k×1dimensional sample is represented as y.

3.3 Gene selection using Imperialist Competitive Algorithm (ICA)

For selecting genes, Imperialist Competitive Algorithm (ICA) is used in this proposed work. For optimization, new evolutionary algorithm used is Imperialist Competitive Algorithm (ICA), which is inspired from imperialist competitive [14]. Imperialism forms base for this ICA's robustness. Beyond Government's own borders, rule and power extension policy is given by imperialism.

Genes count equalized initial population in this algorithm. Among population, some best genes are selected as imperialists genes [15]. Among mentioned imperialists, remaining population (genes) are split to form colonies. Then, among all empires, imperialistic competition is initiated. Power of weakest empire cannot be enhanced and in this competition, it will not get succeed and from competition, it will be eliminated.

As a result, along with competition between empires, all colonies called genes move towards its relevant genes called imperialists. At last, gene convergence is produced using collapse mechanism, where, there will be only one empire in world and all other countries will be the colonies of that one empire. Our solution corresponds to robust empire.

3.3.1. Generating initial empires

Optimization focuses on optimal genes computation. This proposed work assumes genes count as country, so called population. Genes count is a 1×Narray in N-dimensional problem and is defined as,

$$\text{Number of genes} = (x_1, x_2, ..., x_n),\ x_i \in R, 1 \le i \le N \quad (4)$$

Gene's classification accuracy is defined using gene's cost value. For classification, various classifiers are used. For classification, Deep Fuzzy Flexible Neural Forest (DFFNForest) is used in this proposed work. Cost function is defined as,

$$\text{Cost} = \frac{Number\ \ of\ correctly\ \ classified\ \ \ genes}{Total\ number\ \ of\ genes} \quad (5)$$

The N_{pop}should be designed using designed system. Every gene's cost is f(x)at variables $(x_1, x_2, ..., x_n)$. Then

$$cost = f(genes) = f(x1, x2, ..., xn) \quad (6)$$

For creating imperialist so called empires, most powerful genes N_{imp} are selected. Population's remaining N_{col} are assumed as colonies. An imperialist's normalized cost is defined as,

$$c_n = c_n - {}^{max}_{i}\{c_i\} \quad (7)$$

Where, n^{th} imperialist's cost is given by c_n and its normalized cost is expressed as c_n. Every imperialist's normalized power is computed as,

$$P_n = \left\{ \frac{C_n}{\sum_{i=1}^{N_{imp}} C_i} \right\} \quad (8)$$

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 4, 2021, Pages. 4741 - 4758
Received 05 March 2021; Accepted 01 April 2021.

So, empire's initial colonies count is computed as,

$$\text{No.}C_n = \text{round}(p_n . \text{N}_{col}) \quad (9)$$

Where, n^{th} empire's initial colonies count is expressed as No.C_n and all colonies count is expressed as N_{col}. Colonies No.C_n is selected randomly for dividing colonies for imperialists and are given to n^{th} empire.

3.3.2. Moving the colonies of an empire toward the imperialist

Every genes (colony) which move towards imperialist genes (imperialist) by x-units in direction is a vector from colony to imperialist. . A random variable is represented as x and it has uniform distribution. Then,

$$x \sim U(0, \beta \times D),\ \beta > 1 \quad (10)$$

Where, distance between imperialist and genes is represented as dis. Closeness between imperialist and colony is defined by β.

3.3.3 Revolution

In every iteration, in empire, genes count is replaced with same newly generated countries count. Some new countries are generated in this manner by system and empire's some colonies are replaced by this randomly. Empire's colonies count which needs to replaced with same newly generated genes count.

$$\text{N.R.C} = \text{round}\{\text{RevolutionRate} \times \text{No.(The colonies of } empire_n)\} \quad (11)$$

Where, revolutionary colonies count is represented as N.R.C. ICA's global convergence is enhanced using this and trapping to local minima is also avoided using this.

3.3.4 Exchanging positions of the imperialist and a colony

In colony movement, it may access a better position that its imperialist. So, to that position, imperialist is moved and vice versa.

3.3.5. Total power of an empire

Empire's total power is defined by its all own colonies as mentioned belwo:

$$\text{T}\cdot\text{Cn} = \text{cost}(imperialist_n) + \xi \cdot \text{mean}(\text{cost}(\text{colonies of } empire_n)) \quad (12)$$

Where, position coefficient is expressed as ξ.

3.3.6. Imperialistic competition

For taking other empire's colonies possession, all empires will compete with each other. As a result, weaker empires power is decreased gradually and powerful one's get increased. For the same, every empire's possession probability is computed according to its total power [16]. Normalized total cost is computed as,

$$\text{N}\cdot\text{T}\cdot C_n = \text{T}\cdot C_n - \max\{\text{T}\cdot C_i\} \quad (13)$$

Where, nth empire's total cost is represented as $\text{T}\cdot C_n$, normalized cost is represented as $\text{N}\cdot\text{T}\cdot C_n$. Now, every empire's possession probability is computed as,

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 4, 2021, Pages. 4741 - 4758
Received 05 March 2021; Accepted 01 April 2021.

$$P_{p_n} = \left\{ \frac{\mathrm{N} \cdot \mathrm{T} \cdot C_n}{\sum_{i=1}^{N_{imp}} \mathrm{N} \cdot \mathrm{T} \cdot C_i} \right\} \quad (14)$$

Among empires, mentioned colonies are split according to its possession probability. Vector P is formulated as,

$$\mathrm{P} = [\mathrm{p}_{\mathrm{p}1}, \mathrm{p}_{\mathrm{p}2}, \mathrm{p}_{\mathrm{p}3}, \ldots, p_{p_{\mathrm{Nimp}}}] \quad (15)$$

and also vector R is having uniform element distribution.

$$\mathrm{R} = [\mathrm{r}_1, \mathrm{r}_2, \mathrm{r}_3, \ldots, r_{N_{imp}}] \; \mathrm{r}_1, \mathrm{r}_2, \mathrm{r}_3, \ldots, r_{N_{imp}} \; \mathrm{p} \sim \mathrm{U}(0,1) \quad (16)$$

At last, vector D is expressed as,

$$\mathrm{D} = \mathrm{P} - \mathrm{R} = [\mathrm{p}_{\mathrm{p}1} - \mathrm{r}_1, \mathrm{p}_{\mathrm{p}2} - \mathrm{r}_2, \mathrm{p}_{\mathrm{p}3} - \mathrm{r}_3, \ldots, p_{p_{\mathrm{Nimp}}} - r_{N_{imp}}] \quad (17)$$

Mentioned colonies are handed to an empire using D's elements. In D, empire's relevant index is high.

3.3.7. The eliminated empire

If all colonies of empires are lost, it will get collapsed and becomes a normal colony.

3.3.8. Convergence

At last, there will be a highly powerful empire than other competitor and this unique empire will control all colonies. So, cost of all colonies and unique empire are same. This for ensuring no difference between genes (colonies) and its unique empire.

Algorithm 2: Imperialist Competitive Algorithm (ICA)

Input: Number of genes in Gene expression data
Output: Optimal genes

Step 1: Define classification accuracy as objective function: f(x), x=$(x_1, x_2, .., x_d)$
Step 2: Gene's count is initialized
Step 3: In search space, random solution is generated and initial empires are created.
Step 3: Assimilation: In directions, in different way, towards imperialist states, genes are moved.
Step 4: Revolution: In some countries characteristics, random changes are made.
Step 5: Between an imperialist genes and genes, positions are exchanged. Empire control is taken by the genes having better position than imperialist genes, where, existing imperialist genes are replaced.
Step 6: Imperialistic competition: There will be competition among all imperialist for taking colonies possession of each other.
Step 7: Powerless empires are eliminated. There will be a gradual lose in weak empires power and at last, they will be eliminated.
Step 8: Terminate if stop condition is satisfied, else move to step 2.
Step 9: End

3.4 Deep Fuzzy Flexible Neural Forest (DFFNForest) based classification

Then the selected feature will be implemented in the Deep Fuzzy Flexible Neural Forest(DFFNForest). Here Fuzzy function is introduced for enhancing DFNForest classifier's

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 4, 2021, Pages. 4741 - 4758
Received 05 March 2021; Accepted 01 April 2021.

classification results. In the DFFNForest, fuzzy is used to update the weight values of the classifier in the cancer subtype prediction.

FLEXIBLE NEURAL TREE

The FNT model is generated using terminal instruction set T and function set F and is given by,

$$S=F \cup T =\{+_2, +_3, \ldots\ldots,,, +_N\} \cup \{x_1, \ldots., x_n\} \quad (18)$$

Where, non-leaf node's instruction with *I* parameters is represented as $+_i$(*i*= 2, 3, 4,…,*N*), non-leaf node's instruction without parameters are represented as x_1,x_2…,x_n. A non terminal instruction $+_i$(*i*=2, 3, 4,….,*N*) is assumed for generating a flexible neural tree, where, for connecting weights between children and non-leaf node, *i* values are generated randomly. For flexible neural tree, assumed the following flexible activation function.

$$f(x)=(1 + e^{-x})^{-1} \quad (19)$$

Flexible neuron's output$+_n$is expressed as.

$$\sum_n = \sum_{j=1}^{n} w_j * x_j \quad (20)$$

Where, inputs are represented as x_j(*j* = 1, 2,…,*n*). Node's output$+_n$is expressed as

$$out_n = f(sum_n) = ((1 + e^{-sumn})^{-1} \quad (21)$$

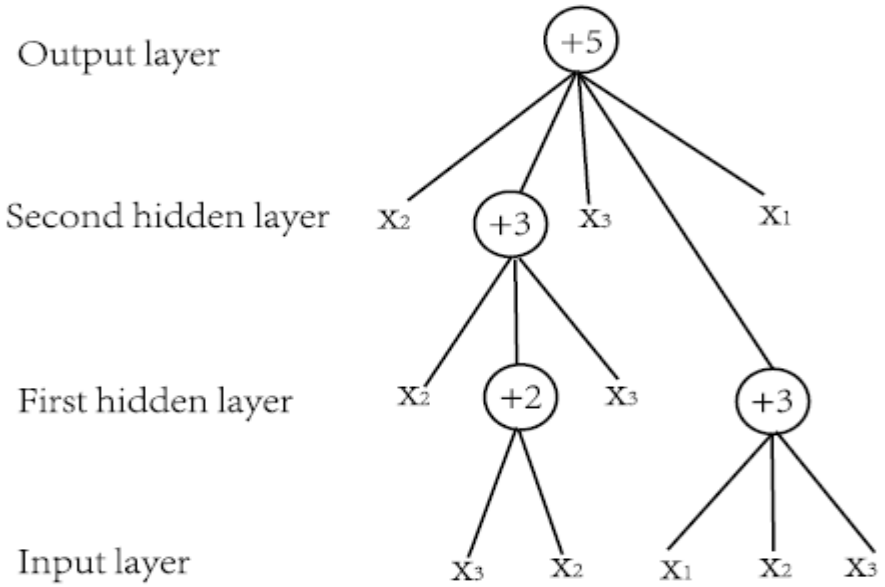

Figure 2: A typical representation of the FNT with function instruction set F=$\{+_2, +_3, +_4, +_5\}$, and terminal instruction set T =$\{x_1, x_2, x_3\}$.

Figure 2 shows typical FNT representation. Total flexible neural tree's output is computed recursively from left-to right using depth-first technique. Over-layer connections are allowed by flexible neural tree model and structure is selected automatically. This model is a sparse one and better generalization performance is exhibited using this model. There are two major steps in this FNT optimization process, namely, tree structure optimization and parameter optimization.

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 4, 2021, Pages. 4741 - 4758
Received 05 March 2021; Accepted 01 April 2021.

Weight value updation using fuzzy function

In the DFFNForest, fuzzy is used to update the weight values of the classifier in the cancer subtype prediction. A fuzzy if-then system is used in simplest classifier which is based on fuzzy rule and it is having high correlation with fuzzy control system. Assume an example having 3 classes. Classification rules are specified for constructing feature or genes weight values. e.g.,

IF x_i is moderate AND x_j is less THEN class is 1
IF x_i is moderate AND x_j is high THEN class is 2
IF x_i is high ANDx_j is minimum THEN class is 2
IF x_i is minimum ANDx_j is high THEN class is 3

The x_j and two features are represented using numerical values with its weights. Linguistic values are used by rules. For every feature, if there exist M possible linguistic values and n features in problem, possible various if-then rules of conjunction type (AND) is M^n. A membership function is used for representing every linguistic values.

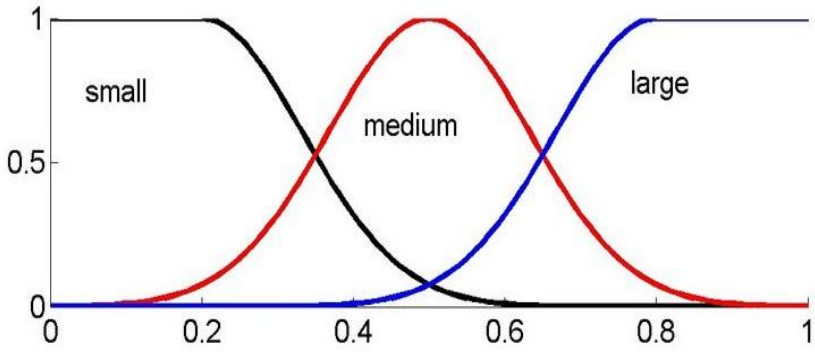

Figure 3: Membership functions for weight values for $gene\ x_i$

THE PROPOSED DEEP FLEXIBLE NEURAL FOREST MODEL

A special type of neural network is flexible neural tree. Parameters and structures optimized automatically using this. However, there exist various problems. At first, there will be one root node which is assumed as output node. Multi classification problems are not able to be dealt with this. Then, model needs to be deepened for obtaining better performance.

However, parameters count will be increased because of this. Parameter optimization algorithm's cost will also be enhanced. For solving flexible neural tree's problems, a deep flexible neural forest (DFNForest) is proposed. Cancer subtypes are classified using this novel flexible neural tree ensemble technique.

In speech and visual recognition tasks, such great success is achieved by deep neural network due to model's complexity an representation learning. Layer by layer feature processing is referred as representation learning. Without additional parameters, cascade forest structure is adopted in designed

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 4, 2021, Pages. 4741 - 4758
Received 05 March 2021; Accepted 01 April 2021.

system which makes FNT deeper. Layer by layer feature processing produces new features as shown in figure 4. For next layer, new features and original features are passed as input.

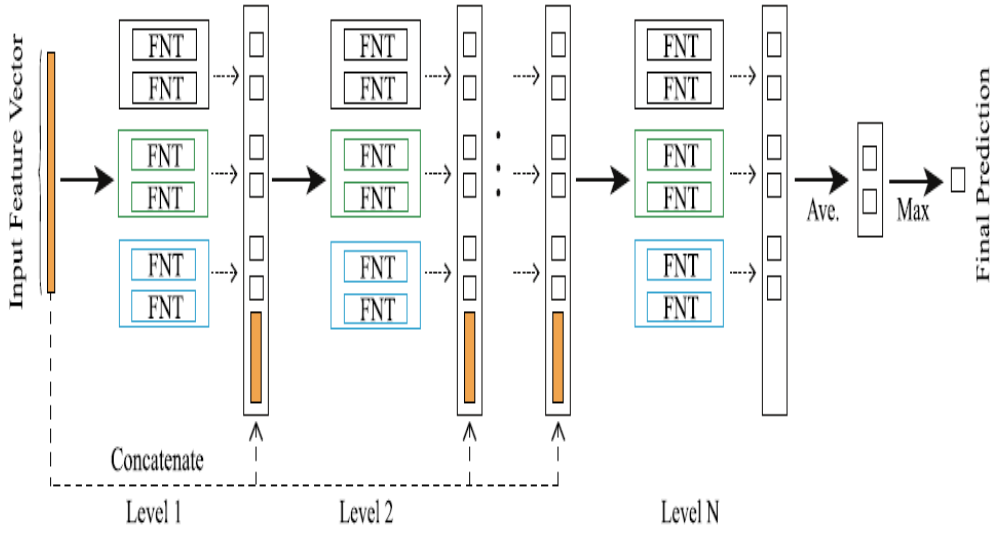

Figure 4: Illustration of the cascade forest structure

In proposed model, every level is a FNT's ensemble. In multi-grain cascade forest(gcForest), decision tree is utilized. Continuous data cannot be applied directly with decision tree, which is a major disadvantage of it. This requires data discretization. Information may be lost because of this. Continuous nature is exhibited by gene expression data. So, FNT is used as base classifier. Following FNT advantages are maintained in this proposed technique.

A sparse model is FNT and cross-layer connections are allowed by this. Better generalization performance is achieved using this while avoiding over-fitting. Parameters and structure are optimized automatically using FNT. In addition, through various FNTs, overall performance is enhanced using proposed ensemble learning. Through various grammars, various FNT structures are generated by system for enhancing ensemble learning's diversity.

For simplicity, assume two FNTs and three forests in every forest. As shown in Figure 5, function set F of $\{+_2, +_3, +_4\}$ is used by first forest, $\{+_2, +_4, +_5\}$ is used by second forest and $\{+_3, +_4, +_5\}$ is used by third forest. The M-ary technique is used for solving FNTs problem in dealing with multi-classification problems. In this, in a forest, multi-class problem is transformed as various two-class problems.

For instance, there is a need to have $k = \log_2 4 = 2$ FNTs in every forest, if it is for-class problem. In forest, classification problem defines trees count as illustrated.

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 4, 2021, Pages. 4741 - 4758
Received 05 March 2021; Accepted 01 April 2021.

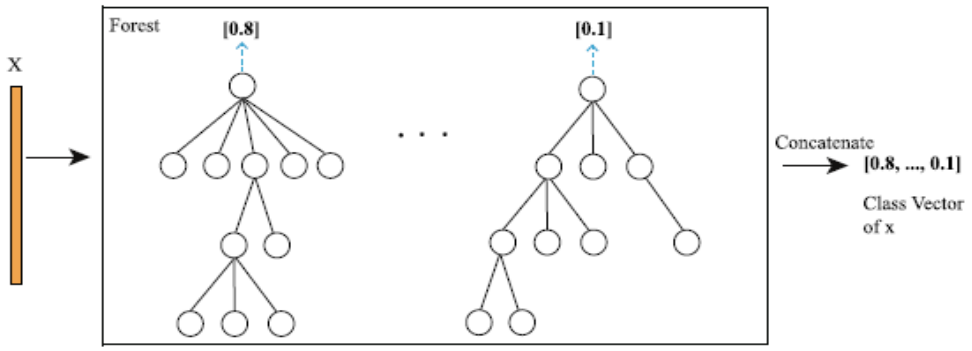

Figure 5: Illustration of class vector generation. Each FNT will generate an estimated value and then concatenate together

For an example, an estimate value is generated by every FNT as demonstrated in Figure 5. A class vector having concatenation with original input feature vector is formed using this estimated value and it is given as an input to next level. For instance, if there are four classes, then a two-dimensional class vector is produced by every forest. Thus 6 (2 ×3) augmented features are received by cascade's next level. Two parts are formed by dividing training set. One is used in validation and another one is used in training.

During the new label addition, validation set is used for verifying entire cascade. Increase in level is stopped, if there no increase in accuracy. Automatic computation of cascade levels count is done in this manner. On datasets with various sizes, this can be used and for small-scale gene expression data, is is highly suitable.

An alternative for deep neural networks is provided using a novel deep learning model called DFNForest. Tree structure optimization algorithm is used for selecting FNT structure automatically in every forest. Adaptive determination of cascade levels are done. An FNT's ensemble is DFNForest. In dealing with multi- classification problems, FNT's shortcomings are resolved using this, where multi-classification problems are converted as various binary classification problems in every forest. Without additional parameters, model depth is enhanced using cascade structure.

4. EXPERIMENTAL RESULTS

The experimental analysis is carried out in MATLAB. Here conducted cancer subtype predictions using prostate cancer dataset which is downloaded from https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE15484. It contains 65 samples and two classes such as grade 8 and 6. The performance of the proposed Binomial probability Distribution based Principal Component Analysis (BDPCA) with Deep Fuzzy Flexible Neural Forest (DFFNForest) approach is compared with existing Deep Flexible Neural Forest (DFNForest) and Deep Fuzzy Flexible Neural Forest (DFFNForest) scheme with respect to F-measure, recall, precision, accuracy, and error. The performance comparison is shown in table 1.

Table 1: Performance comparison

Performance metrics in (%)	Methods		
	DFNForest	DFFNForest	BDPCA with DFFNForest
Accuracy	85	93.37	95.87
Precision	85.03	93.43	95.87
Recall	85	93.37	95.87

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 4, 2021, Pages. 4741 - 4758
Received 05 March 2021; Accepted 01 April 2021.

F-measure	85.01	93.40	95.87
Error	15	6.62	4.12

Performance metrics

4.1 Accuracy

Highly intuitive performance measure is accuracy. Ratio between correctly predicted observation and total observation defines accuracy value.

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + FN + TN} \quad (22)$$

where,

TP indicates True Positive
FNindicates False Negative
FPindicates False Positive
TN indicates True Negative

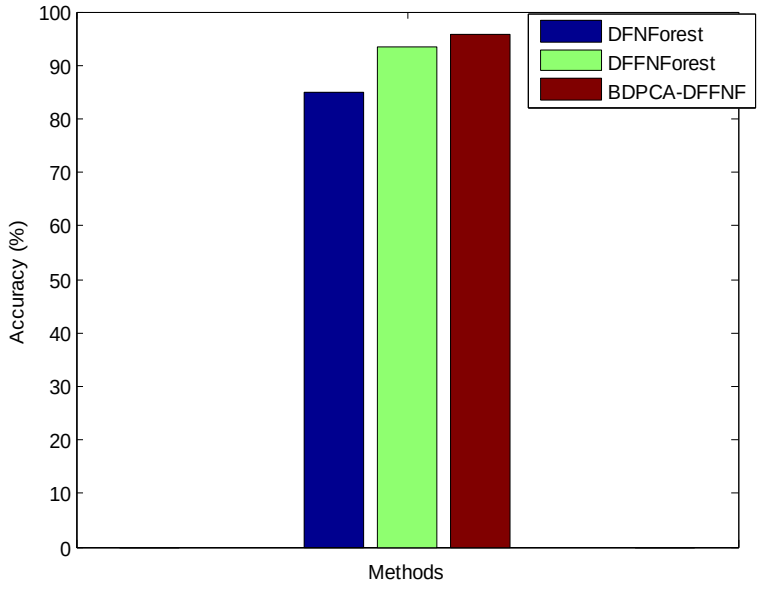

Figure 7: Accuracy comparison

Proposed BDPCA with DFFNForest's accuracy performance is compared with available DFNForest and DFFNForest methods and it isillustrated in figure 7. Various techniques are represented in x-axis and in y-axis, accuracy value is represented. In this proposed research work, optimal genes are selected by using ICA. It enhances accuracy rate. Proposed system achieves 95.87% of accuracy whereas other method such as DFNForest and DFFNForest attains 85% and 93.37% respectively as illustrated in experimental results.

4.2 Precision

Ratio between correctly predicted positive observations and total predicted positive observations defines precision values.

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 4, 2021, Pages. 4741 - 4758
Received 05 March 2021; Accepted 01 April 2021.

$$\text{Precision} = \frac{TP}{TP + FP} \quad (23)$$

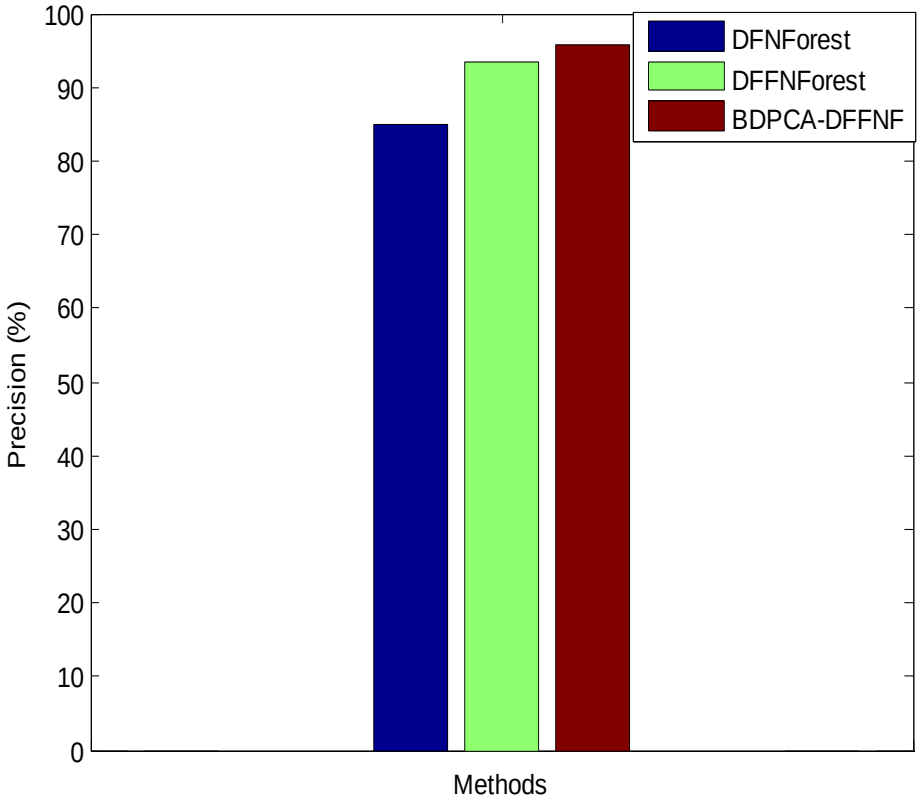

Figure 8: Precision comparison

The performance of the proposed BDPCA with DFFNForest scheme is compared with the existing DFNForest and DFFNForest methods in terms of precision. The precision comparison is shown in figure 8.Proposed BDPCA with DFFNForest achieves 95.87% of precision when existing DFNForest and DFFNForest method provides 85.03% and 93.43 % respectively as shown in experimental results.

4.3 Recall

Ratio between correctly predicted positive observations and all observaions in actual class defines recall value.

$$\text{Recall} = \frac{TP}{TP + FN} \quad (24)$$

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 4, 2021, Pages. 4741 - 4758
Received 05 March 2021; Accepted 01 April 2021.

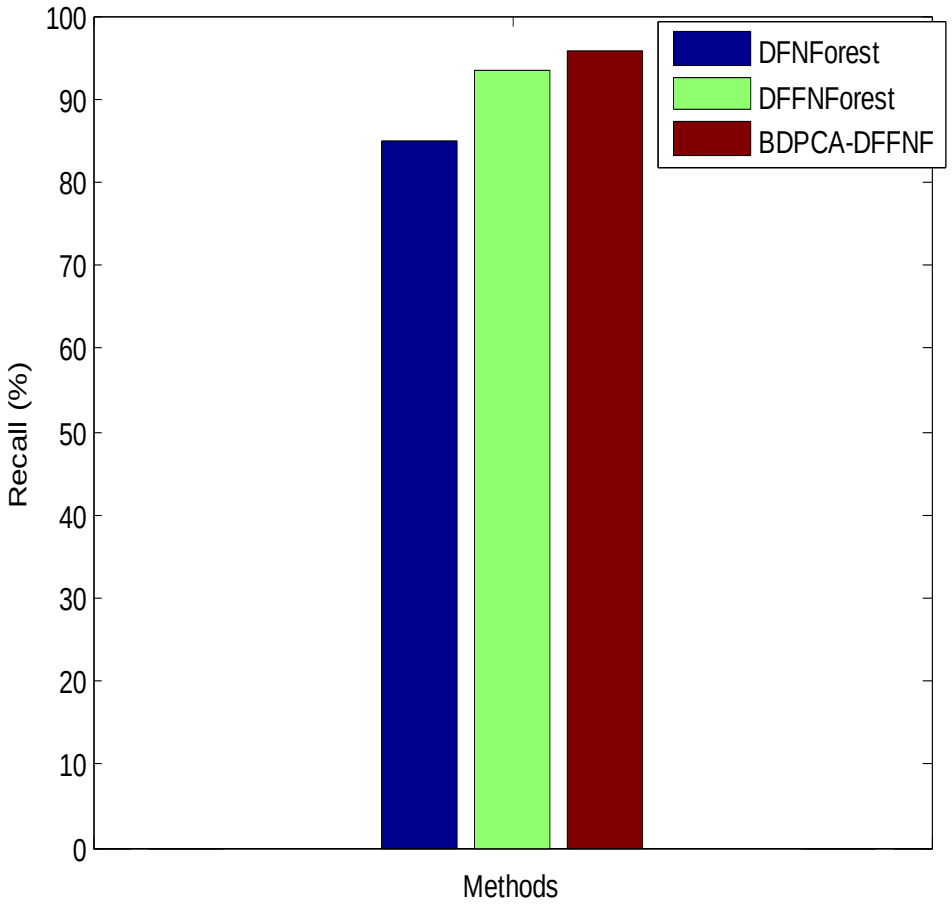

Figure 9: Recall comparison

The recall performance of the BDPCA with DFFNForest, DFFNForest and DFNForest methods are compared which is shown in figure 9. In this proposed research work, dimensionality reduction is performed by using the Binomial probability Distribution based Principal Component Analysis (BDPCA). It enhances recall rate. Proposed system attains 95.87% of recall where as DFNForest and DFFNForest achieves 85% and 93.37% respectively as indicated in this graph.

4.4 F-measure

Recall and Precision's weighted average value defined F1 score. Both false negatives and positives are considered for computation of this value.

$$\text{F-measure} = 2*\frac{(Recall * Precision)}{(Recall + Precision)} \quad (25)$$

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 4, 2021, Pages. 4741 - 4758
Received 05 March 2021; Accepted 01 April 2021.

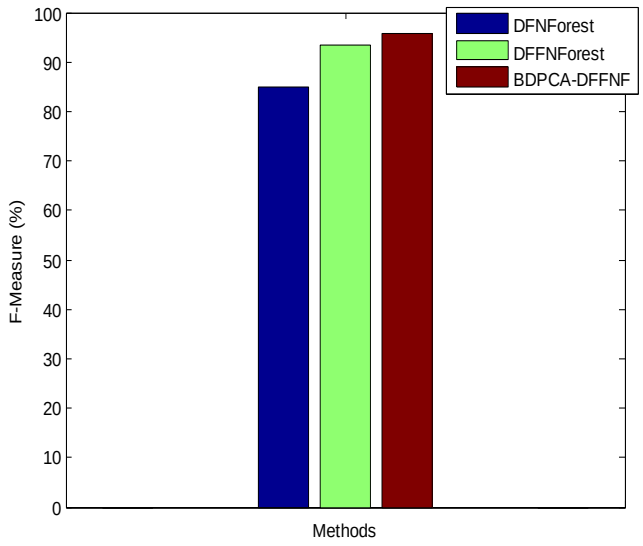

Figure 10: F-measure comparison

Figure 10 shows BDPCA with DFFNForest, DFFNForest and DFNForest method's f-measure performance. Various methods are represented in x-axis and in y-axis, f-measure value is given. The f-measure of the proposed system is 95.87% when DFNForest and DFFNForest method attains 85.01% and 93.40% respectively as shown in experimentation results.

4.5 Error

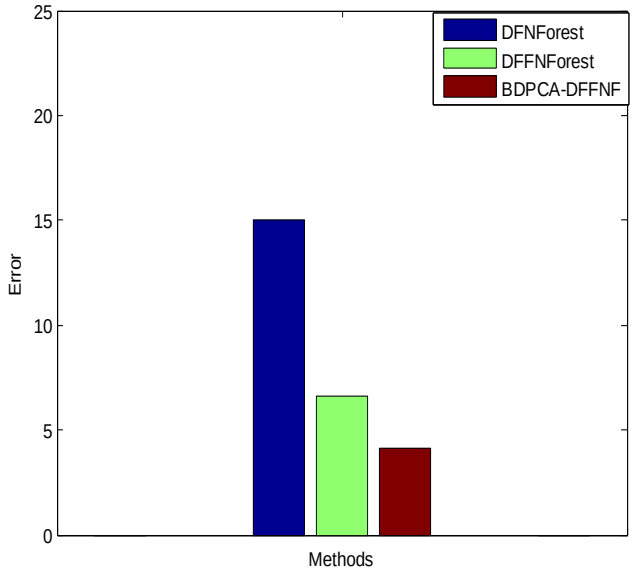

Figure 11: Error comparison

The performance of the BDPCA with DFFNForest, DFFNForest and DFNForest methods are compared in terms of error. In this proposed work, optimal gene is performed by using ICA for reducing

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 4, 2021, Pages. 4741 - 4758
Received 05 March 2021; Accepted 01 April 2021.

classifier's miss rate. Proposed ICA with DFFNForest system attains 4.12 % error rate whereas DFNForest and DFFNForest method achieves 15% and 6.62% respectively as indicated in above graph.

5. CONCLUSION

In this proposed research work, Binomial probability Distribution based Principal Component Analysis (BDPCA) with Deep Fuzzy Flexible Neural Forest (DFFNForest) is designed for accurate cancer subtypes diagnosis. In order to solve the high-dimensionality challenges, designed a Binomial probability Distribution based Principal Component Analysis (BDPCA). The optimal gene selection is performed by using Imperialist Competitive Algorithm (ICA) which enhances classifier's accuracy. According to the selected features, the classification is performed by using Deep Fuzzy Flexible Neural Forest (DFFNForest). With respect to error, f-measure, recall, precision and accuracy, better performance is achieved by proposed system when compared to available systems as demonstrated in experimental results.

References

1. Xiao, Y., Wu, J., Lin, Z., & Zhao, X. (2018). A deep learning-based multi-model ensemble method for cancer prediction. *Computer methods and programs in biomedicine*, *153*, 1-9.
2. Liu, Y., Wang, X. D., Qiu, M., & Zhao, H. (2019, October). Machine Learning for Cancer Subtype Prediction with FSA Method. In *International Conference on Smart Computing and Communication* (pp. 387-397). Springer, Cham.
3. Yang, B., Zhang, Y., Pang, S., Shang, X., Zhao, X., & Han, M. (2019). Integrating Multi-Omic Data with Deep Subspace Fusion Clustering for Cancer Subtype Prediction. *IEEE/ACM transactions on computational biology and bioinformatics*.
4. Kim, S. (2016). Weighted K-means support vector machine for cancer prediction. Springerplus, 5(1), 1-11.
5. Xu, T., Le, T. D., Liu, L., Wang, R., Sun, B., & Li, J. (2016). Identifying cancer subtypes from miRNA-TF-mRNA regulatory networks and expression data. *PloS one*, *11*(4).
6. Xu, J., Wu, P., Chen, Y., Meng, Q., Dawood, H., & Khan, M. M. (2019). A novel deep flexible neural forest model for classification of cancer subtypes based on gene expression data. *IEEE Access*, *7*, 22086-22095.
7. Karim, M., Cochez, M., Beyan, O., Decker, S., & Lange, C. (2019). Onconetexplainer: explainable predictions of cancer types based on gene expression data. *arXiv preprint arXiv:1909.04169*.
8. Mostavi, M., Chiu, Y. C., Huang, Y., & Chen, Y. (2020). Convolutional neural network models for cancer type prediction based on gene expression. *BMC Medical Genomics*, *13*(5), 1-13.
9. Guo, Y., Qi, Y., Li, Z., & Shang, X. (2018). Improvement of cancer subtype prediction by incorporating transcriptome expression data and heterogeneous biological networks. *BMC medical genomics*, *11*(6), 87-98.
10. Tharwat, A. (2016). Principal Component Analysis: An Overview. *Pattern Recognition*, *3*(3), 197-240.
11. Zou, H., & Xue, L. (2018). A selective overview of sparse principal component analysis. *Proceedings of the IEEE*, *106*(8), 1311-1320.
12. Jamal, A., Handayani, A., Septiandri, A. A., Ripmiatin, E., & Effendi, Y. (2018). Dimensionality Reduction using PCA and K-Means Clustering for Breast Cancer Prediction. *LONTAR KOMPUTER: Jurnal Ilmiah Teknologi Informasi*, 192-201.
13. Oh, J., & Kwak, N. (2016). Generalized mean for robust principal component analysis. *Pattern Recognition*, *54*, 116-127.
14. Mousavirad, S. J., Bidgoli, A. A., Komleh, H. E., & Schaefer, G. (2019). A memetic imperialist competitive algorithm with chaotic maps for multi-layer neural network training. *International Journal of Bio-Inspired Computation*, *14*(4), 227-236.

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 4, 2021, Pages. 4741 - 4758
Received 05 March 2021; Accepted 01 April 2021.

15. Majd, A., Sahebi, G., Daneshtalab, M., Plosila, J., Lotfi, S., & Tenhunen, H. (2018). Parallel imperialist competitive algorithms. *Concurrency and Computation: Practice and Experience*, *30*(7), e4393.
16. Kalteh, A. A., Zarbakhsh, P., Jirabadi, M., & Addeh, J. (2013). A research about breast cancer detection using different neural networks and K-MICA algorithm. *Journal of cancer research and therapeutics*, *9*(3), 456.

AUTHOR PROFILE

P. Avila Clemenshia received Bachelor of Science in Mathematics from Bharathiar University-Coimbatore, India in 2002 and Master of Computer Applications from from Bharathiar University in the year 2005 and M.Phil from Bharathiar University in the year 2014. She is currently working as a Assistant Professor, Department of Computer Science, Nirmala College for Women (Autonomous), Coimbatore, and her research work focuses on Big Data Analytics, Data Mining. She has five years of teaching experience. She also has two years of programming experience. Currently she is a regular part time Research Scholar in Department of Computer Science at Sri Ramakrishna College of Arts & Science, Coimbatore,Tamil Nadu, India.

Dr.B. Mukunthan pursed Bachelor of Science in Computer Science from Bharathiar University, India in 2004 and Master of Computer Applications from Bharathiar University in year 2007 and Ph.D from Anna University - Chennai in 2013. He is currently working as Associate Professor in Department of Computer Science,School of Computing, Sri Ramakrishna College of Arts and Science (Autonomous),Nava India,Coimbatore since 2017. He is a member of IEEE & IEEE computer society since 2009, a life member of the MISTE since 2010. He has published more than 25 research papers in reputed International journals. He is also Microsoft Certified Solution Developer. His main research work focuses on Algorithms, Bioinformatics, Big Data Analytics, Data Mining, IOT and Neural Networks. He also invented a Novel and Efficient online Bioinformatics Tool and filed for patent. He has 13 years of teaching experience and 11 years of Research Experience.