Cancer miRNA biomarkers classification using a new representation algorithm and evolutionary deep learning

Article *in* Soft Computing · February 2021

DOI: 10.1007/s00500-020-05366-w

CITATIONS
8

READS
220

4 authors, including:

Mohammad Saniee Abadeh
Tarbiat Modares University
124 PUBLICATIONS **2,018** CITATIONS
SEE PROFILE

Saeed Sarbaziazad
Tarbiat Modares University
7 PUBLICATIONS **84** CITATIONS
SEE PROFILE

Najmeh Sadat Jaddi
Tarbiat Modares University
30 PUBLICATIONS **663** CITATIONS
SEE PROFILE

Soft Computing
https://doi.org/10.1007/s00500-020-05366-w

METHODOLOGIES AND APPLICATION

Cancer miRNA biomarkers classification using a new representation algorithm and evolutionary deep learning

Niousha Bagheri Khoulenjani[1] · Mohammad Saniee Abadeh[1] · Saeed Sarbazi-Azad[1] · Najmeh Sadat Jaddi[1]

Abstract

The diagnosis of cancer is presently undergoing a change of paradigm for the diagnostic panel using molecular biomarkers. MicroRNA (miRNA) is one of the most important genomic datasets presenting the genome sequences. Since several studies have shown the relationship between miRNAs and cancers, data mining and machine learning methods can be incorporated to extract a large amount of knowledge from cancer genomic datasets. However, previous research works on the identification of cancers from miRNAs have made it possible to diagnose cancer, and the accuracy of some classes is not quite satisfactory. Therefore, this research is aimed at promoting a super-class (meta-label) approach and deep learning in a three-phase method to diagnose cancers from miRNAs. The steps in the first phase of the proposed method, named Representation learning, are partitioning data into super-classes, meta-data creation and super-classes classification. This phase helps data to be split into some subsets to improve classification accuracy. In other words, the first phase groups labels based on the separability of classes into a meta-label, and then a multi-label learner is built to predict these meta-labels. In the second phase, a feature selection to reduce the dimensions of the problem is applied to each super-class to help to focus the attention of an induction algorithm in those features that are more important to predict the target concept. In the third phase of the proposed method, an evolutionary deep neural network for the classification of labels in each super-class is performed. The last two phases are done separately for each subset in which five super-classes and subsequently five deep neural networks are trained. The experimental results reveal that the proposed method achieved more efficient results than 19 recent machine learning methods. Despite the fact that evaluating the dataset which consists of 29 types of cancers provides a more complicated situation for the convolutional neural network to be learned, the performance of the method is noticeably better than other existing methods. The other success which can be considered here is a significant reduction in running time comparing to other methods.

Keywords MicroRNA · Meta-label · Super-class · Intelligent reasoning · Feature selection · Optimized convolutional neural network

1 Introduction

Diagnosing and predicting cancers from its genome sequences is one of the most conducted research works in the past decade. MicroRNA (miRNA) which presents this sequence is one of the big genomic cancer data sets (Peralta et al. 2015; Sabzehzari and Naghavi 2018). They are small non-coding RNA molecules (19–23 nt) that regulate gene expression by binding to miRNA response elements in messenger RNA (mRNA) at the post-transcription level (Yoon et al. 2019).

MiRNA is used as a regulator in the transcription of a target mRNA (messenger RNA: a kind of RNA which carries information from cell nucleus to cytoplasm) (Brown

Communicated by V. Loia.

✉ Mohammad Saniee Abadeh
saniee@modares.ac.ir

Niousha Bagheri Khoulenjani
newsha_bagheri@modares.ac.ir

Saeed Sarbazi-Azad
s.sarbaziazad@modares.ac.ir

Najmeh Sadat Jaddi
najmehjaddi@gmail.com

[1] Faculty of Electrical and Computer Engineering, Tarbiat Modares University, Jalal Al-e Ahmad, P.O. Box: 14115-194 Tehran, Iran

Published online: 12 October 2020

2007). MiRNAs downregulate or up-regulate translation, as downregulation is by linking to a special part of the target mRNA and silencing the translation process (Vasudevan et al. 2007). This regulation ability of MiRNAs consists of 20–30% of human genes. As miRNA may have different targets, several mRNAs might code a single gene; thus, miRNA has a complex system to regulate translation in cells (Jovanovic et al. 2010). According to recent studies, improvement in several factors has different effects on the level of expression of miRNAs including the organism type, the disease and the environment. Different types of illnesses lead to a different amount of miRNAs' expression such as cancer (Bartel 2004). The role of miRNAs in the regulation of cell metabolisms is considered as a great assistance to diagnose, prognosis and targeted therapies. Besides, the efficacy of cancer chemotherapy can be predicted by the level of miRNAs expression; thus, this level of expression can be mentioned as a biomarker itself (Ye et al. 2020). Also, scientists have benefited from the molecularly targeted drug function of miRNAs by interfering with the expression of some vital genes in the development of a disease (Barger and Nana-Sinkam 2015). There have been some experiments illustrated results that miRNAs are suitable biomarkers for cancer (Alevizos and Illei 2010).

miRNA expression profiles can be applied for diagnosis and classification of tumors, and thus could be developed as cancer biomarkers (Lu et al. 2005). Due to the high cost of traditional experimental methods, various methods using machine learning algorithms have been proposed to predict potential associations between miRNAs and cancers (Chen et al. 2018). Also, some studies showed that early diagnosing cancer can provide a better result in treatment. Besides, in diseases like cancer, molecular assessments are more accurate and cheaper than other symptoms. As recently many improvements have happened in detecting miRNA via blood, a simple test can reveal information about the cancer stage or the organ which struggles with the disease (Lopez-Rincon et al. 2018).

This study aims to introduce a framework in order to diagnose cancers from microRNA promoting a meta-label (super-class) approach and deep learning. One of the main purposes of the proposed method is to simplify the classification task of miRNA datasets by splitting data into some super-classes. Because of the existence of a high variety of cancers in miRNA, a super-class classification in the first phase is introduced. Thus, the data are divided into some super-classes to make the classification task easier. In this way, the labels of samples are mapped into a meta-label. Another issue with the data was the high amount of redundant features making data ambiguous. A feature selection method in order to reduce the number of features with the maintenance of classification performance is applied. Finally, a convolutional neural network (CNN) that its hyper-parameters are optimized by a GA is used for each super-class. The training time of deep learning algorithms is mostly high (because of having a high amount of neurons in each layer) and dependent on some factors such as the hyper-parameters, the implementation platform and the hardware. In order to make the process of classification by CNN faster, a graphical processing unit (GPU) is recommended for calculating the mathematic equation in a parallel way.

The rest of this paper is organized as follows: Sect. 2 provides a review of related works to this research. Section 3 describes the proposed method under two sub-sections: preliminaries which provide the fundamentals of the concepts of feature selection employed in this paper are provided in the first sub-section and the details of the genetic algorithm and convolutional neural network are presented in the second and third sub-section. Section 4 discusses the results of experiments to assess the performance of the proposed method. Discussion is given in Sect. 5, and Sect. 6 concludes the performance of the proposed method and the results obtained.

2 Related works

First, there is a need to have a review on previous research works which had been done about miRNA and neural networks. There are several points which should be considered about the classification of miRNA datasets and the classification ways which had been done and their pros and cons of each research. Besides, having a rough review about neural networks and optimization problems could provide an adequate amount of knowledge which is helpful in this part.

2.1 Classification of miRNA datasets

Table 1 is consisting of four different studies in the miRNAs classification. All of them have conducted a study on miRNAs as biomarkers in different cancers. Each of which has its own pros and cons which are mentioned in Table 1. All these methods have proved that miRNAs are good biomarkers in detecting cancers and in different types of cancers they might reveal vital information about many aspects of the disease. However, each method may have some constraints according to the results obtained.

Ye et al. (2020) have used SVM, random forest and logistic regression with the help of 10 miRNAs extracted as feature miRNAs, to evaluate and train the classification model. These 10 feature miRNAs are used to distinguishing cancerous tissues precisely. In the second study which was about breast cancer a tree-based machine learning

method was used, which is reported to be useful in extracting rules and easy to interpret clinically. Due to the fact that plenty of miRNAs are expressed in a different way in breast cancer, identifying the minimal set of miRNAs biomarkers for breast cancer classification was quite impossible. However, they claimed that they have tried to minimize the set of miRNAs and maintained the accuracy of the classification at the same time by using larger datasets, removing unwanted variation and merging breast cancer's subtypes to conclude a better result (Sherafatian 2018).

Zhang et al. (2020) claimed that RF was a better classifier than SVM in lung cancer. Besides, they have presented a novel computational approach and identified potential lung cancer biomarkers at the same time.

In another study constructing non-linear class separation boundaries between miRNAs and its target which is experimentally validated has been performed by support vector machines (SVMs). This can lead us to avoid false positives by taking experimentally validated miRNA-target interactions (MTIs) into account. By taking advantage of SVMs' ability, complex decision boundaries were concluded and non-linear or even embedded class relationships were found. In fact, this method is considered as a complementary method for previous ones, while it reveals that edge biomarkers contain more biological information. However, this method highly depends on the miRTBase (miRNAs-target interaction database) and newly added MTIs are not able to be detected (Pian et al. 2020).

2.2 Evolutionary-based optimization

Optimization problems are dividing into four groups: constrained or unconstrained problems, continuous or discrete problems, single-objective or multi-objective problems, static or dynamic problems (Abualigah 2020). A local-based search algorithm runs iteratively with one candidate solution until reaching the termination criteria. B-hill climbing (Abualigah et al. 2018), and hill climbing (Abualigah 2017) are some examples of this category. The second category consists of a collection of randomly generated solutions which is called population and in each iteration these solutions mix and produce new solutions. The optimal solution according to the fitness function generated by mixture in population for example cuckoo optimization algorithm (Rajabioun 2011), artificial bee colony algorithm (Karaboga and Akay 2009) and genetic algorithm (Abualigah and Hanandeh 2015). Swarm-based algorithm also starts with a population however, in each iteration solutions are produced based on historical information which is obtained in the previous generations. Multi-verse optimizer algorithm (Mirjalili et al. 2016), krill herd algorithm (Abualigah 2016) and flower pollination algorithm (Yang 2012) are some examples in the third category. The last category is a combination of two

Table 1 miRNA classification methods

Method	Type of cancer	Advantages	Disadvantages
Usage of machine learning to identify the feature miRNAs, which can be reliably used as biomarkers for diagnosis LUSC (Ye et al. 2020)	Lung squamous cell carcinoma (LUSC)	Finding consistent microRNAs, distinguishing cancerous tissues precisely	Some confusing factors such as patient age, gender, ethnicity, tumor stage and technical factors have not been evaluated
Tree-based machine learning (Sherafatian 2018)	Breast cancer	Rule extraction by identifying minimal set of biomarkers and presenting the most important miRNAs in this cancer classification. Besides, the data are clinically interpretable by normalizing unwanted variation factors and removing them from the clustering process, distinct classes can be obtained	Batch effect influences PC1 and PC2 noticeably, Cancer stage cannot be used as a differentiation factor in PCA plots even after normalization
Supervised classifier adopting RF with two-step feature selection method (Zhang et al. 2020)	Lung cancer	Constructing an efficient classifier RF was better than SVM in this case, distinguishing potential lung cancer biomarkers according to the classification results	Biomarkers are mutual between several types of cancers; Therefore, it is just used for lung cancer and in different cancers may lead to different results.
Discovering DM-miRNAs by using a Support vector machine (Pian et al. 2020)	Breast and kidney cancers	Finding complex boundaries and non-linear relationships in classes, This method can identify cancer types accurately	Newly gained miRNAs targeted interactions (MTIs) are not detectable as it has not been recorded in miRTarBase

algorithms for example hybrid whale optimization algorithm with local search algorithm (Abdel-Basset et al. 2018), the hybrid genetic wind-driven heuristic optimization algorithm (Javaid et al. 2017) and the hybrid firefly and particle swarm optimization algorithm (Aydilek 2018).

Optimization of hyper-parameters of neural networks has been evaluated through several studies (Young 2015; Chin et al. 2017; Soon et al. 2017; Wang et al. 2019). Since hyper-parameters are defined differently, optimization aims might vary in different studies. In some studies, hyper-parameters are considered important just in neural network layers thus its total architecture remains unchanged while it is tuned in an excellent way (Wang et al. 2019). While in other studies other factors has been changed such as number or order of layers, learning rate and, so on. This makes a totally new neural network based on what have been planned before. Thus, the CNN hyper-parameter which is fine-tuned is considered an optimization problem with a various mode function. CNN's performance might become disappointing due to the fact that an insensible selection strategy causes unfavorable hyper-parameter configuration. CNN hyper-parameters can be implemented by integer programming via a vast range of heuristic algorithms (Young 2015). genetic algorithm (GA), which is a common SI algorithm, is definitely a good choice for inherently supporting integer optimization. GA had been used for optimization in CNNs' hyper-parameters (Lopez-Rincon et al. 2018). The performance of GA on CNN hyper-parameter is limited due to the fact that the cost of CNN training is noticeably high and GA has an ability to produce a premature solution. Numerous iterations for adjusting weights and biases are among the main parts of requirements for training a deep CNN. This makes the evaluation of hyper-parameter's quality costly (Wang et al. 2019).

3 Proposed method

The proposed method is presented in the following two sub-sections. First, a brief explanation of basic concepts used in this research as a basis is described. In the second subsection, the details of the proposed method are presented.

3.1 Preliminaries

In this sub-section in order to introduce the scope of the research, basic concepts including feature selection, evolutionary algorithm and convolutional neural network are described.

3.1.1 Feature selection

Data pre-processing is often neglected, but it is definitely a significant step in the data mining process. The purpose of data pre-processing methods and more specifically data reduction models is to clean up and simplify input data (Peralta et al. 2015). Some data, such as gene expression microarrays, contain a very high number of features corresponding to the number of samples. In these data, the number of samples is mostly lower than 400 where the number of features is at least 4000 (Sarbazi-Azad and Abadeh 2018). Selecting significant features from feature-set is important because in some datasets most of the features cannot help or contribute in class prediction and may cause to overfit or a biased model creation. In other words, feature selection is the process of ignoring those excessive features and selecting features that can better describe the data (Potharaju and Sreedevi 2019; Bolón-Canedo et al. 2014). Considering too many features might lead to a decline in the model's prediction, therefore selecting relevant features is an important step through each study. There are four important positive points about selecting the suitable feature for predictive models: (1) omitting confusing variables, (2) constructing simpler models which are not disposed to overfitting, (3) increasing the efficiency of the predicted model and a reduction of data collecting costs in the future and (4) putting more values on variables which can be tested (Torres and Judson-Torres 2019). The number of features in the miRNA data is higher corresponding to the number of samples, so the feature selection can help in the construction of a model with enhanced performance in terms of classification accuracy. Recent researches have shown that data complexity measures are effective tools to select significant genes from data (Morán-Fernández et al. 2017).

Feature selection methods are divided into two groups: supervised and unsupervised learning. The first group needs a prior knowledge about the classes' labels. On the other hand, unsupervised labels are not requisite for the feature selection process. This makes unsupervised feature selection suitable for clustering techniques (Abualigah 2019).

One of the well-known data complexity measures to quantify the complexity of data is the fisher discriminant ratio (Ho and Basu 2000). This is one of the measures that the classification accuracy is highly dependent on it (Ghasemzadeh et al. 2019). In other words, the fisher discriminant ratio measures how separated are the classes according to specific features. The results in Sarbazi-Azad et al. (2018) show that the features with a high fisher rank can be used to create a high-performance model. Fisher is the ratio of the between-class and within-class scatter matrix. By maximizing the distance between classes and

minimizing distance within classes, the discriminant will be better and more efficient. In this research, the fisher has been applied for feature selection. The fisher discriminant ratio (F1) for *i*th feature in a multiclass problem is calculated as follows:

$$F1 = \frac{\sum_{c_j=1}^{C} \sum_{c_k=c_j+1}^{C} p_{cj} p_{ck} \left(\mu_{c_j}^{f_i} - \mu_{c_k}^{f_i}\right)^2}{\sum_{c_j=1}^{c} p_{c_j} \left(\sigma_{c_j}^{f_i}\right)^2} \quad (1)$$

where p_{cj}, $\mu_{c_j}^{f_i}$ and $\sigma_{c_j}^{f_i}$ are the number of classes, the ratio of *j*th class(the number of samples in class *j* over the total number of samples), mean and variance of feature *i* over *j*th class, respectively. This process has a major impact on classes separability and consequently the classification accuracy, as the model can only concentrate on important features and ignore the others (Peralta et al. 2015; Potharaju and Sreedevi 2019; Bolón-Canedo et al. 2014; Ho and Basu 2000; Sarbazi-Azad et al. 2018; Lewis et al. 2006). Thus, the rank of each feature is calculated from (1) and high-rank ones are selected significant genes.

3.1.2 Genetic algorithm

By inspiration from natural animal instincts, evolutionary algorithms became able to intelligently explore in the enormous research area. Intelligent exploration means avoidance of searching completely a special part of a large search space, which has the possibility of having the desired result. This is because of their capability of escaping local optima in the search area. One of the most popular evolutionary algorithms is the standard GA introduced by Holland (Holland 1992). Many variants of genetic algorithms have been developed and applied to a wide domain of problems.

Figure 1 demonstrates the steps of GA. As depicted, first an initial population containing chromosomes or individuals is created. Each chromosome should be evaluated that how suitable for our purpose is, and therefore, a fitness function is defined. The fitness of the produced chromosomes is calculated by the fitness function. A population (chromosomes) of parents having high fitness is selected. Regarding the population of parents and recombination operator, new offspring is produced. Recombination is the process of transmitting some pieces in two chromosomes, which is called crossover (Garzelli et al. 2008). For example, a chromosome can be produced by a combination of the first and the second half part of two chromosomes by crossover operation. This step triggers the transmission of different genes from parents to offspring. After performing crossover on chromosomes, the mutation function is applied to produce some other offspring. The mutation is the process of mutating the value of a random set of indexes from a chromosome. The number of genes being mutated in each iteration may be one or more. Finally, the offspring having good enough fitness based on the defined function is considered as the new generation to be added to the population. This process is iterated till reaching predefined criteria such as the number of iterations or till a special value of fitness is achieved.

3.1.3 Convolutional neural networks

CNNs are inspired by biological neural network constituting the animal brain. Researches being done in the 1950s to 1960s by D.H Hubel and T.N Wiesel on the brain of mammals suggested a new model for how the mammals perceive the world visually (Hubel and Wiesel 1959, 1960). Such systems learn to do tasks generally without being programmed with any task-specific rules, by considering examples. Figure 2 shows the architecture of the algorithm.

CNN contains four main layers including convolution layer, pooling layer, activation layer and fully connected layer. Convolution, pooling and activation layers are considered as the hidden layer. The convolution is performed by filters or kernels (these terms are used interchangeably) to produce a feature map from the input data. After feeding input into the network, a series of convolutions will be done by the filters considered in each layer. Just like any other neural network, an activation function is used in order to get the output of the convolution layer. The activation function considered in this study is the *rectified linear unit* (ReLU). It is common to use a pooling layer after the activation. The pooling reduces the number of parameters and computation and also avoids the overfitting function of pooling by reducing the dimensionality of feature maps (Aghdam and Heravi 2017; Liu et al. 2019). Max pooling is the most frequent type of pooling, which takes the maximum value in each window. After the first part (convolution and pooling layers), there is a classification part including a few fully connected layers. Neurons in a fully connected layer have full connections to all the activations in the previous layer. The fully connected layer contains the softmax function for classification. The softmax function is calculated as Eq. (2), where z is a vector of the inputs to the output layer and j indexes the output units ($j = 1, 2, \ldots, K$)

$$\sigma(z)_j = \frac{e^{z_j}}{\sum_{h=1}^{k} e^{z_h}} \quad (2)$$

Training a CNN works in the same way as a regular neural network, using backpropagation. Backpropagation technique is the most common one for training neural networks. The underlying simplicity of the technique and its relative power are the reasons for the popularity. In

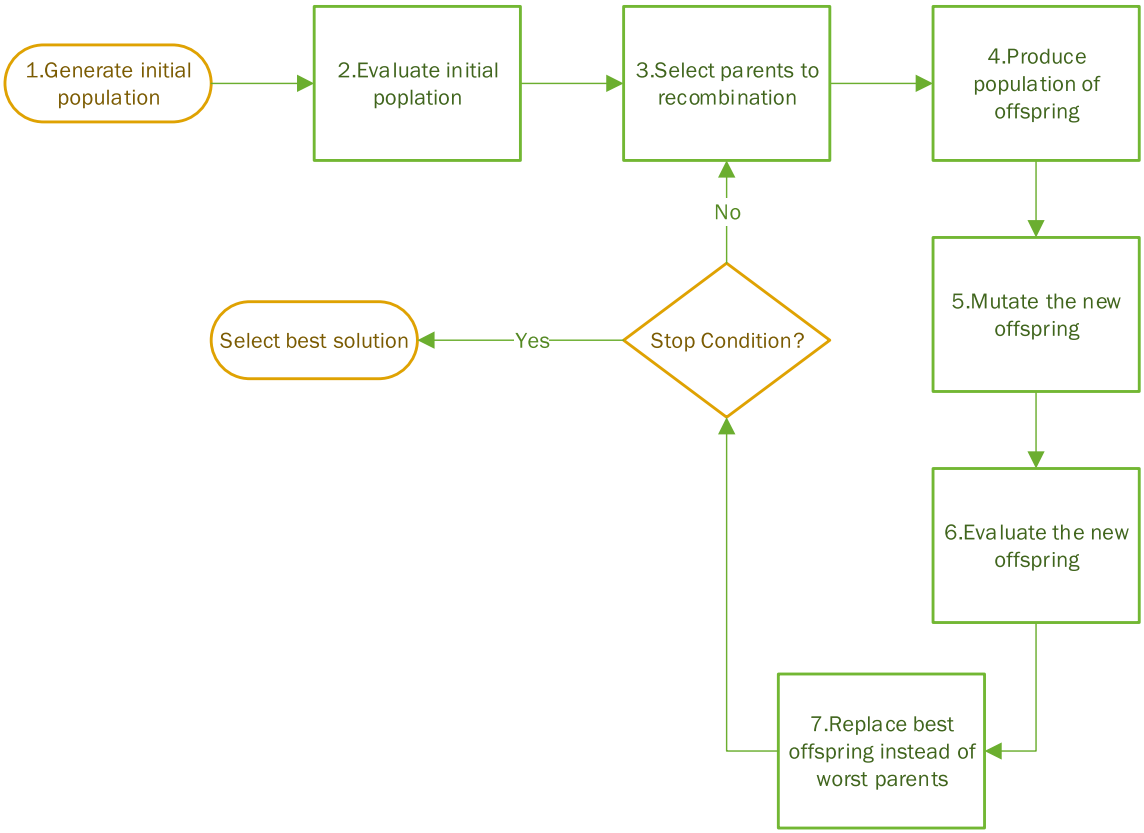

Fig. 1 The steps of the GA

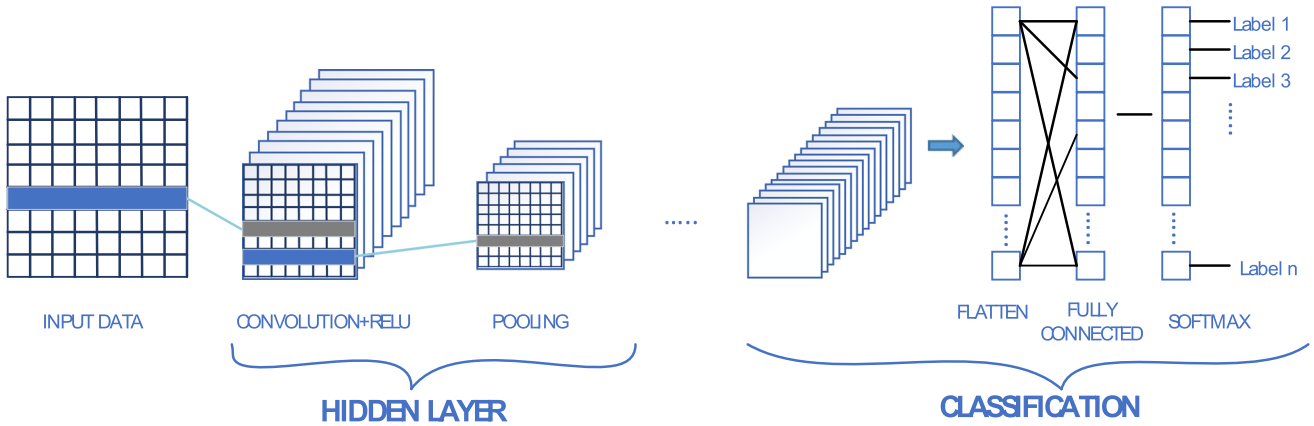

Fig. 2 Architecture of a CNN including three parts: input layer, hidden layers and classification

backpropagation, the weights are updated based on the loss function. The loss function is used to measure the inconsistency between the predicted value and the actual label. It can be said that the neural network model is trained by minimizing the loss function by optimization algorithms such as stochastic gradient descent. In this paper, cross-entropy is used as the loss function. There is also a dropout operation which is common to put it immediately after an activation layer in order to avoid overfitting issue (Aghdam and Heravi 2017).

To train a CNN model, three important hyper-parameters must be assigned:

- The size of the convolution window (kernel size)
- The number of filters (that is, how many filters do we want to use)
- The Max pooling window

The convolution window is one of the important hyper-parameters in CNN structure which can automatically learn low-level and high-level features and it can be useful for

cancer analysis (Han 2018; Öztürk 2018). Also, the number of filters is directly related to the complexity of the model (Montavon et al. 2011) and a different kind of Max pooling window can dramatically change the representation (Scherer et al. 2010).

3.2 New representation algorithm and evolutionary deep learning

The dataset applied in this study for evaluation can be accessed from the atlas.[1] 29 different types of cancers are considered in this data.

Table 2 demonstrates the name of cancer classes, the number of samples for each cancer. The super-classes and the cancers in each one are also determined. As it can be seen, 5 super-classes have been considered that each super-class is created in a way that the cancers in the same super-class would have a maximum of separability and minimum of class overlap with the cancers from other super-classes.

Cancer is a complex genetic disease involving anomalies in the structure and expression of coding and non-coding genes resulting from a number of inherited and somatic defects in some critical genes. Nowadays it has become apparent that the cancer cell's genomic complexity is much higher than anticipated. miRNA is one of the important big genomic datasets presenting the genome sequences. miRNA gene regulation plays an important role in cell proliferation, death in the cell, tumorigenesis and the growth of mammalian cells (Edgar 2016).

In this section, a novel method for the classification of miRNA data is presented. As mentioned in Sect. 2.2, the nature of classifying the data is a challenging task because of having various types of cancers and high amount of features corresponding to the number of samples in each class. Therefore, it is tried to pre-process the dataset (in terms of the complexity of data) to more separable data.

The first step in every data analyzing method is normalizing input data. In order to normalize the features range, *Z*-score is applied to the input data. Then the normalized data divided into two groups: train and test set, respectively. Training set went under a specific process which its outcome is a set of models trained for each super-class of the training dataset. As can be seen in the Fig. 3, the training process can be totally divided into three main phases, namely representation, feature selection and classification. These three phases guide us to detect a proper model as a classifier which can evaluate each test dataset and returns a suitable class for them.

In the representation learning an important procedure followed and lead to distinguishing several super-classes among training datasets which is named intelligent reasoning. Intelligent reasoning is also used in the test dataset to first determine which super-class the test data belongs to and then evaluation takes place according to the proposed model for the determined super-class. The next step is to select important features among all of them to have a better classifier according to the features. The last part of the training process is finding a suitable parameter for the CNN model in each super-class which was obtained in the intelligent reasoning in the representation learning process.

All in all, by following training procedure a set of models will be obtained for each super-class. After detecting super-classes testing data can go through the procedure and after passing intelligent reasoning and defining a super-class for test data the appropriate model which had been trained by the training process can be used

Table 2 All super-classes, their classes and the number of samples in each class

Datasets	Class	Number of samples
Super-class0	BLCA	415
	KIRC	261
	SKCM	452
	UCEC	418
	UVM	80
Super-class1	ACC	80
	BRCA	778
	CHOL	35
	DLBC	47
	ESCA	200
	LIHC	374
	PCPG	184
Super-class2	HNSC	488
	KIRP	292
	LGG	530
	LUSC	341
	MESO	87
	TGCT	155
Super-class3	CESC	308
	KICH	66
	LUAD	458
	PAAD	179
	PRAD	500
	THYM	124
Super-class4	FPPP	45
	SARC	262
	UCS	57
	STAD	399
	THCA	514

[1] http://cancergenome.nih.gov/.

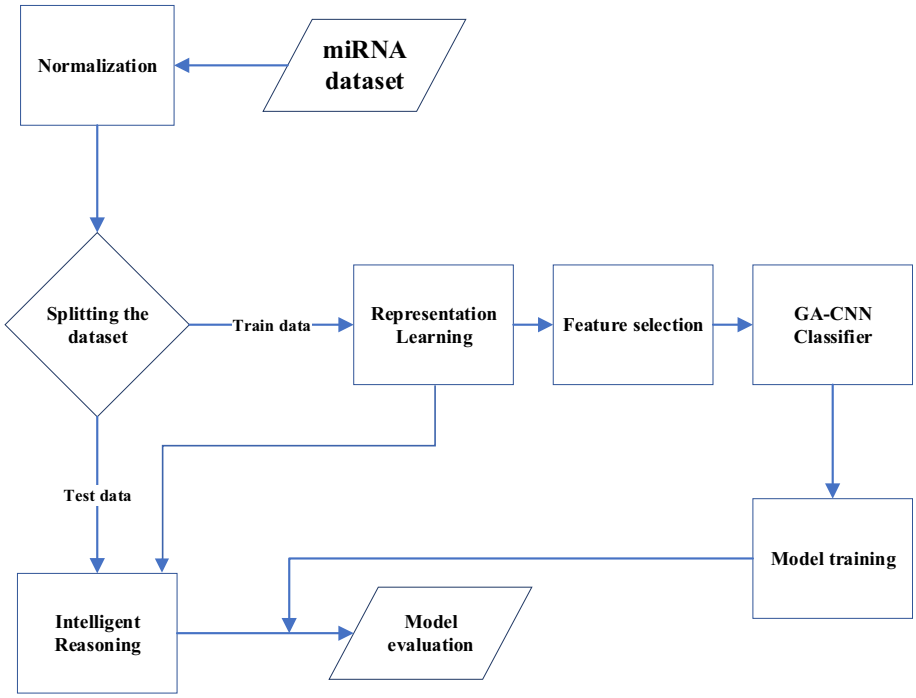

Fig. 3 The flowchart of proposed method

to classify the test data. In fact, intelligent reasoning is a part of both train and test procedure.

The structure of the first phase is shown in Fig. 4. As demonstrated, first an SVM classifier from the total data is made. Then, the confusion matrix from the normalized dataset was achieved by utilizing the SVM classifier (Lewis et al. 2006). The confusion matrix is a good way to report results in M-class classification problems because the relationship between the classifier outputs and the true ones can be observed (Salem et al. 2017). Therefore, the confusion matrix results from the classifier indicating how correct the samples are classified. For example, the number in row i and column j shows the number of samples being in class i and miss-classified in class j. Based on the results from the confusion matrix, the data are divided into five super-classes. Each super-class contains classes having the least overlapping with classes in other ones. The main goal of this way is to maximize inter-class separability, where the intra-class separability may be low or high. The inter-class and intra-class separability are the distinguishability between super-classes and classes of each super-class, respectively. Thus, each sample is classified in one of the super-classes. In other words, a meta-label for the data is considered. It is obvious that the number of labels in meta-label is the same as the number of super-classes. In the next step, a classifier for classifying the super-classes should be trained. It is important to predict the super-classes label with high accuracy because the final performance of the proposed method is highly dependent on the performance of the meta-label prediction model. After splitting data into five subsets (super-classes), in order to accurately classify the samples into the correct super-class a meta-data approach is considered. This step is shown as step 2 in Fig. 4. As demonstrated a meta-data is made that its sample size is the same as the input data and each column indicates the prediction of one of the considered classifiers. The classification approaches considered to fill meta-data cells are One Vs One (OVO) and One Vs Rest (OVR) (Bishop 2006). The first one is a popular approach for class binarization considering every pair of classes as a data subset. In the latter one k (number of super-classes) classifiers are made that for each one the samples belong to the category of interest are labeled as positive and the others have negative labels. The number of classifiers in the two mentioned approaches is a combination of (k, 2) and k, respectively, where k is the number of super-classes. After the features of the meta-data are made, the meta-label from the meta-label generator is concatenated to the features as the last column of meta-data. In fact, this is an ensembling method to create new data that each column is the vote of one of the classifiers to the samples. The last step in this phase is classifying samples based on the created meta-data (each row of meta-data is a map of the corresponding sample). Therefore, a classifier is trained to predict the meta-label of the created meta-data.

Super-classes which are demonstrated in Table 2 have several classes and samples with specific features for each one. The total number of features for these classes is 1046. It is very high corresponding to the number of samples and the number of classes in the data. Figure 4 indicates the

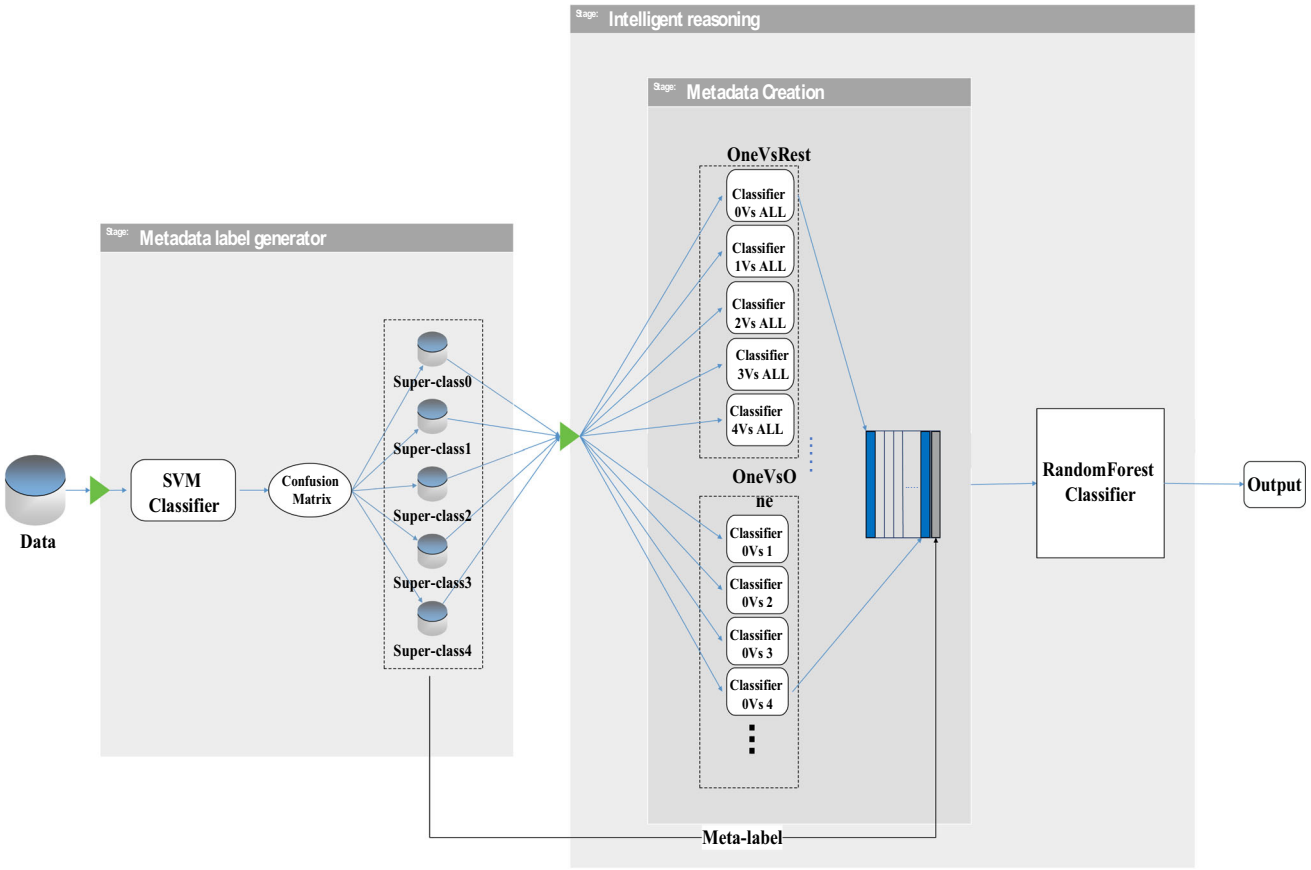

Fig. 4 Structure of meta-data classification (representation)

process of creating meta-data to super-classes classification. As demonstrated, this phase has a two-stage namely meta-label generator and intelligent reasoning. The first stage is the process of dividing the data into some super-classes based on the confusion matrix, and the second one is the procedure of creating a meta-data for super-classes classification. Finally, a model for meta-label classification is constructed.

In the second phase, a feature selection algorithm proposed in Sarbazi-Azad et al. (2018) is applied to each super-class in order to reduce the number of features. Feature selection is important both to speed up learning and to improve the quality of concept. Another reason for dimensionality reduction is that the distribution of data was not normal. It means that the number of samples in some classes was high and others was low. This issue can cause to overfit when we have many redundant features or genes having no additional information for class prediction. The dimensionality reduction is done after classifying the super-classes because the classes located in each super-class may have special characteristics that it can cause to feature different roles and importance in each subset. As it is expected, the feature-set for each super-class was different; therefore, a subset of features, making classes more separable, was maintained, and the rest of the features are ignored for each super-class. The feature selection method applied in this phase is more described in Sect. 3.1.1. In the dimensionality reduction process, the input having 1046 features was fed into the feature selection method and 300 features were selected from each subset having the same meta-label.

As in Fig. 5, the final phase of the proposed method is the classification of the samples of each super-class. Each subset resulted from gene selection including 300 features, is fed into a CNN classifier (expressed in the background section). So, the number of the trained model is the same as the number of meta-labels determined in the first phase. This process is done in order to optimize the CNN algorithm. The process of this phase is what is presented in Abualigah (2019) but it is described here briefly. As demonstrated to predict a test sample (the below branch) first the meta-label of the sample must be determined. It is done by intelligent reasoning from Fig. 4. Then, based on the meta-label, the features are selected (the ids of significant features for each meta-label is assigned in the training phase). After the reduction in dimensions, the appropriate CNN model is used to predict the actual label of the sample.

The trained CNN includes six layers in the last phase of the proposed method. First, the input vectors of length 300 were fed into the CNN. The hidden layers considered were pooling process and convolution. After the convolution layer, there are classification layers including fully connected layers and the output result (a probability vector showing the probability of being the sample classified in each class). As mentioned in Sect. 3.1.3, each layer of this algorithm has hyper-parameters that each set of hyper-parameters indicates a CNN classifier parameter. Hyper-parameters of layer "i" include a number of filters (NF_i), convolution window (CW_i) and Max Pulling Window (PW_i). In the last step, fully the connected layer has a number of filter parameters (NF_4). In order to optimize CNN classifier, a genetic algorithm is utilized (Lopez-Rincon et al. 2018). Each set of hyper-parameters in a CNN is considered as an individual for a genetic algorithm according to Fig. 6. During GA, there are steps in which chromosomes are quantified, and due to the limitation of the neural network, individuals (hyper-parameters) have to take value in valid boundaries. Therefore, the maximum and minimum values for each hyper-parameter must be specified. Also, individuals or solutions during mutation and crossover may violate the limitations and become invalid solutions. Some algorithms eliminate these solutions from the population, while others modify and turn them into a new valid solution. In this paper, a repair algorithm in order to rectify the offspring is added to GA. The limitations of hyper-parameters and the repair algorithm are explained in Lopez-Rincon et al. (2018).

The fitness of the proposed method is the average accuracy of tenfold cross-validation of the CNN classifier that its hyper-parameters were assigned as a solution in the genetic algorithm. In K-fold cross-validation, the dataset is divided into K parts, where the number of samples is $\frac{\text{number of samples}}{K}$ in each fold and the same number of

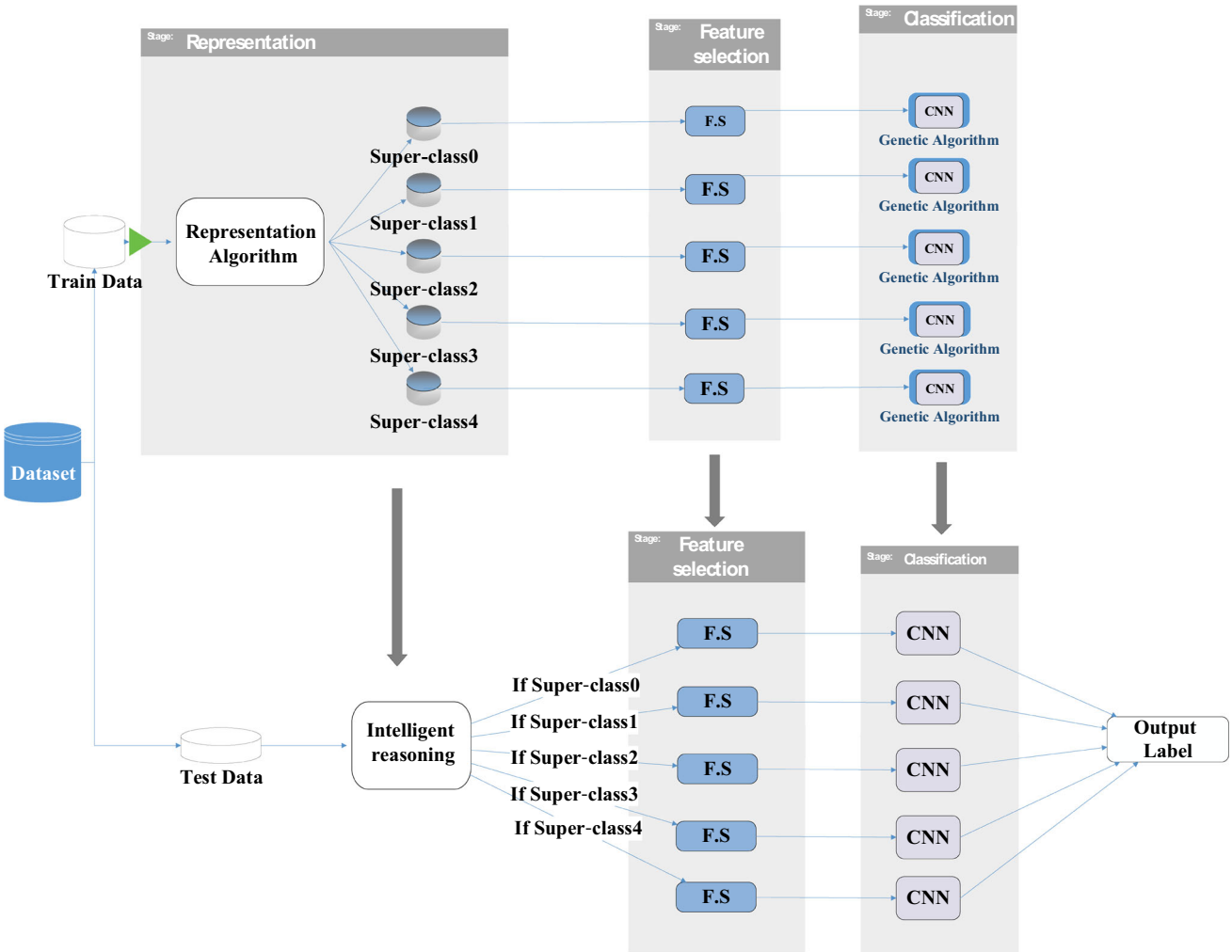

Fig. 5 An overview of the proposed method, including three phases; meta-data creation, feature selection and classification (genetic algorithm and convolutional neural network)

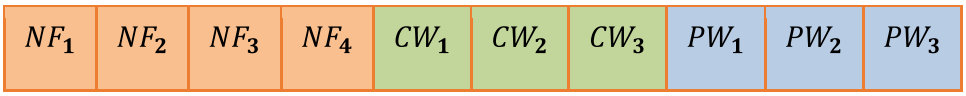

Fig. 6 A sample of individuals in the genetic algorithm which can find optimized CNN

features as the whole dataset. In each iteration, one of the K folds is selected as the test and others as the train set. It is obvious that the number of iterations in this way is K. In this case, all of the samples are tested once by this approach. The accuracy of this approach is considered as the mean of accuracy over the folds.

4 Experimental results

In this section, the evaluation of the method and the most notable results will be discussed. The proposed framework is verified with a miRNA dataset containing 29 types of Cancer. All codes and algorithms are implemented in the Python programming language. The implementation of the CNN architecture was on a system having GTX1080 TI GPU.

4.1 Performance evaluation

The performance of the proposed Representation learning is evaluated in a conventional way via fivefold cross-validation. The meta-data for super-classes prediction was created by the votes of One Vs Rest and One Vs One classifiers to the samples. Five super-classes were considered in the experiment, each one containing samples of some classes. After so many experiments, the classifiers considered for meta-data creation and super-classes prediction were gradient boosting and random forest, respectively. The average accuracies of fivefold for meta-data classification are 100%, indicating all samples are correctly classified in its own super-class. In other words, the meta-label for each sample is correctly predicted.

The classification accuracy of the proposed method for each super-class is reported in Table 3. All the results reported are the mean of tenfold cross-validation. The four widely used criteria including accuracy (Acc), precision (Prec), recall (Rec) and F1 score (Fujino et al. 2008) are calculated as follows:

$$\text{Acc} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{2}$$

$$\text{Prec} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{3}$$

$$\text{Rec} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{4}$$

$$\text{Fsc} = 2 * \frac{\text{Prec} * \text{Rec}}{\text{Prec} + \text{Rec}} \tag{5}$$

These evaluation metrics are used in order to compare the performance of the proposed method with state-of-the-art researches. Table 4 shows the F1 score and the precision of each super-class. Also, the Loss function and the accuracy of the test and train during backpropagation are illustrated in Fig. 7. The vertical and horizontal axis shows the accuracy of each super-class and iteration numbers, respectively. According to the charts, the classification accuracy has a high growth even in the first 50 iterations and it is because of the feature selection done in the pre-processing phase to select important features helping the model to concentrate on the significant features and ignore others. In this paper, 300 features are selected in each super-classes that are available at "https://github.com/NioushaBagheri/selected-feature/blob/master/Selected%20feature.csv." The total accuracy of the proposed method is higher compared to the (Lopez-Rincon et al. 2018) method (97.6% compared to 96.6% reported in Table 4).

Table 5 shows the mean of tenfold cross-validation test accuracy for each of 29 classes compared with the method recently proposed in Lopez-Rincon et al. (2018) namely EA-Optimized CNN. The classes with more accuracy resulted are highlighted in bold. Despite a low number of samples in CHOL, DLBC and FPPP classes and being counted as hard ones to classify in Lopez-Rincon et al. (2018), the achieved accuracy of these classes is 100%. As seen in Table 1, the CHOL, DLBC and ESCA classes are in the same super-class and the accuracies resulted are outperformed according to reports. It is because some classes can be better distinguished compared with some special classes (the super-class classification approach), while it is hard to discern the right label when being identified by some other classes. In other words, the model can be fit better if we can make the data easier and this is done by grouping samples of each class into a super-classes where the number of super-classes is lower than the number of classes. Consequently, the rules of the classifier can be extracted easier and the prediction will be more efficient.

We believe that the results can outperform other methods because of two important reasons; first, the feature selection and then the super-class approach. By feature selection, the features that cannot help to class prediction are removed. It helps CNN to regulate the classification pattern better. It is because the model can concentrate on

Table 3 The train and test accuracy and number of samples of each super-class compared to the EA-optimized CNN method

Super-classes	Train accuracy	Test accuracy	Test accuracy of EA-optimized CNN (Lopez-Rincon et al. 2018)	Number of samples
Super-class0	0.99375	0.975309	0.963066	1626
Super-class1	0.997041	0.990148	0.949477	1698
Super-class2	0.980952	0.973475	0.983234	1893
Super-class3	0.958333	0.9375	0.976452	1635
Super-class4	1	0.988235	0.963204	1277
Total	0.99	0.976	0.966	8129

Table 4 Tenfold cross-validation results on miRNA datasets

Super-classes	Precision	F1-score
Super-class0	0.9870	0.9863
Super-class1	0.9973	0.9971
Super-class2	0.9630	0.9626
Super-class3	0.9400	0.9382
Super-class4	0.9897	0.9885

the features having more importance and ability to label prediction. In the super-class approach, instead of classifications the whole data by a single one classifier, for each super-class, a deep neural network classifier is trained. One of the data complexities is tried to overcome by dividing data into sub-data. The mentioned data complexity is called data imbalance that can obviously be seen by the sample's distribution. As it can be seen, the number of samples in some classes such as DLBC, CHOL and FPPP is very low compared to others, and its effect can be observed in classification accuracy results in Lopez-Rincon et al. (2018). This problem is solved by the super-class approach. As reported in Lopez-Rincon et al. (2018), the process of the large search space of the genetic algorithm and complex calculation of CNN took 14 days with a 744 core system. It is while the runtime of the proposed method is only 3 days long. The reasons behind this are first splitting data into smaller subsets with a little number of classes and second applying feature selection in order to decrease the dataset's dimension from 1046 to 300 features. In addition, utilizing the GPU for CNN calculations helped so much in the runtime of the implemented method.

Finally, the proposed method is compared with 18 other methods from Breiman (1999, 2001), Friedman (2001), Cox (1958), Crammer et al. (2006), Tikhonov (1943), Hearst et al. (1998), Altman (1992), Tibshirani et al. (2002), Boser et al. (1992), Breiman et al. (1984), Geurts et al. (2006) and Hastie et al. (2009). In Fig. 8. The implementations are taken from the Scikit-Learn Python Packages (Pedregosa et al. 2011). The accuracies reported are the average of tenfold cross-validation. It can be concluded from the results that the proposed method has achieved the highest accuracy among other methods.

4.2 Parameter setting

In the proposed method, a genetic algorithm (GA) is used in order to find the optimal solution for hyper-parameters of the CNN classifier. Accuracy and 10 hyper-parameters of CNN are considered as the fitness function and individuals in GA, respectively. The GA is executed with various parameter settings, and the best results are obtained for the parameters as specified in Table 6. The initial population and the number of offspring are considered in each generation 50 and 25, respectively.

The founded hyper-parameters of CNN reported in the EA-optimized CNN (Karaboga and Akay 2009) were $NF_1 = 36$, $CW_1 = 206$, $PW_1 = 175$, $NF_2 = 251$, $CW_2 = 41$, $PW_2 = 4$, $NF_3 = 2$, $CW_3 = 133$, $PW_3 = 7$ and $NF_4 = 2$. In the proposed method, the hyper-parameters of the trained CNN models optimized by the GA are reported in Table 7. Each row indicates the hyper-parameters of one of the super-classes, respectively.

4.3 Statistical analysis

To statistically compare the performance of the proposed method with EA-optimized CNN (Karaboga and Akay 2009), Z-test is performed:

$$Z = \frac{\bar{X}_1 - \bar{X}_2}{\sqrt{\frac{\sigma_1^2}{n_1} - \frac{\sigma_2^2}{n_2}}} \tag{6}$$

Here, $\bar{X}_i$ and σ_i^2 are the mean and the standard division of all the accuracy of classes which obtain from the proposed method and EA-optimized CNN. Also, we used $n_1 = n_2 = 29$, because there are 29 values available for classes of the dataset. Therefore, the degrees of freedom for the Z-test is $29 \times 2 - 2 = 56$. The Z values and related p values are $Z = 3.4271$ and p value $= 0.000609$, respectively. The calculated p values for the corresponding

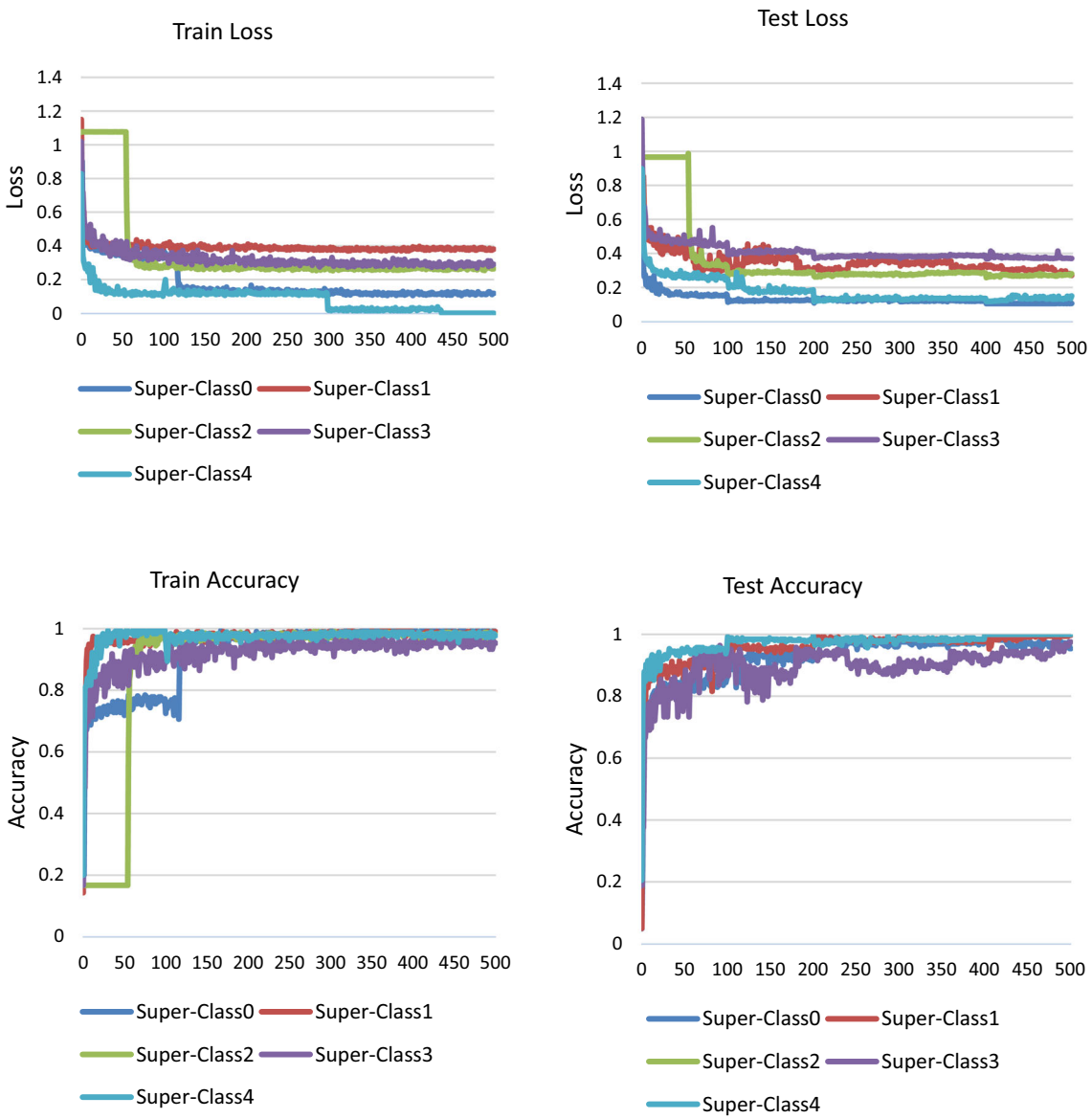

Fig. 7 Training and test accuracy, with training and test loss of the proposed method corresponding to each meta-label

Z values eventually reject the null hypothesis that there is no difference between the proposed method and other related methods (EA-optimized CNN) with a significance level of 0.05 for the miRNA dataset in favor of the alternative.

A diversity measure, namely the fitness standard division in a population in the genetic algorithm is compared and studied in this paper in order to illustrate the proportion of diversity in the populations and the search quality:

$$\text{stddev}\,(P) = \sqrt{\frac{\sum_{i=1}^{N}(f_i - \bar{f})^2}{N-1}} \tag{7}$$

Figure 9 demonstrates the natural evolution of standard deviation over 100 generations for each super-class. Although standard deviation shows great volatility, it displays a steep descent at about the 70th generation and quickly converges to close to zero. The standard divisions of stddevs are $\sigma_{\text{stddev0}} = 0.05792$, $\sigma_{\text{stddev1}} = 0.06822$, $\sigma_{\text{stddev2}} = 0.06891$, $\sigma_{\text{stddev3}} = 0.05777$, $\sigma_{\text{stddev0}} = 0.06205$.

Figure 10 shows the average fitness of the population of all five genetic algorithms during 100 generations. As the population evolves the average fitness increases while the width of the fitness range narrows (the fitness variance of the population is reduced Fig. 9). Therefore, we can conclude that genetic algorithms are converged to optimal solutions.

Table 5 The classification accuracy of the proposed method and EA-optimized CNN for each class

Class	Number of samples	Accuracy of the proposed method	Accuracy of EA-optimized CNN (Lopez-Rincon et al. 2018)
ACC	**80**	**0.9375**	0.8889
BLCA	415	0.988	**1**
BRCA	**778**	**1**	0.9518
CESC	**308**	**0.9667**	0.875
CHOL	**35**	**1**	0.6
DLBC	**47**	**1**	0.8
ESCA	**200**	**1**	0.92
FPPP	**45**	**1**	0.8333
HNSC	488	0.9485	**0.9836**
KICH	66	0.9833	**1**
KIRC	**261**	**0.9808**	0.9655
KIRP	292	1	**1**
LGG	530	1	**1**
LIHC	374	1	**1**
LUAD	458	0.8667	**1**
LUSC	341	0.9412	**1**
MESO	**87**	**1**	0.7272
PAAD	179	0.95	**1**
PCPG	184	1	**1**
PRAD	500	1	**1**
SARC	262	1	**1**
SKCM	452	0.9889	**1**
STAD	**399**	**0.9875**	0.9268
TGCT	155	0.9677	**1**
THCA	514	0.9398	**0.98**
THYM	124	0.9833	**1**
UCEC	**418**	**1**	0.9459
UCS	57	0.9091	**1**
UVM	80	1	**1**

5 Discussion

In this article, the miRNA dataset of 29 types of cancers is used. Here, we proposed a Representation Algorithm, using confusion matrix, SVM, One Vs One and One Vs All classifiers to split the dataset into five sub-classes. Representation algorithm aims to help classifiers for making data more distinguishable and reducing the complexity and thereby, reducing running time. Also, in this way, a feature selection method is used for dimension reduction and finding significant genes in each sub-classes. The performance of the proposed method is found to be superior as compared with 18 state-of-art machine learning algorithms (Fig. 7) in terms of accuracy. From the tables, it is evident that the proposed method using Representation learning and feature selection performs better compared to the related existing method, EA-optimized CNN, as most of the classes' accuracy is higher and some of them reached to 100%. In this investigation, the proposed method provides a framework to utilize GA for finding hyper-parameters of CNN which is used for each super-classes. In addition, the running time of this method is 3 days by using GPU, while EA-optimized CNN takes 14 days by a very powerful system consisting of 744 cores (Lopez-Rincon et al. 2018). Our investigation is focused on maximizing classification accuracy with less features and reducing the running time at the same time and reached successful results.

6 Conclusion

The initiation and progression of cancer may involve microRNA which is grouped as small non-coding RNAs capable of regulating gene expression. The human malignancy can be characterized by their expression profiles. Extraction and analysis of miRNAs can give the

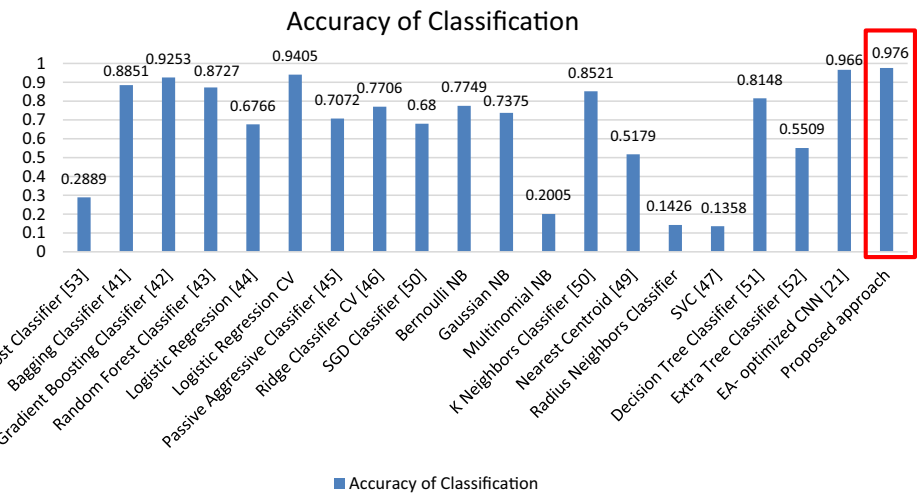

Fig. 8 The classification accuracy of the proposed method compared to other algorithms

Table 6 Values of different parameters for GA

Parameters	NP Population size	*K* Number of generations	MR Mutation rate	CR Crossover rate
Values	50	100	0.9	0.1

Table 7 The hyper-parameters of the CNN models trained for each super-class

Super-classes	NF_1	NF_2	NF_3	NF_4	CW_1	CW_2	CW_3	PW_1	PW_2	PW_3
0	3	80	3	4	154	383	6	60	18	7
1	2	2	50	2	6	277	18	15	14	7
2	5	4	14	3	6	116	44	6	35	35
3	2	20	9	5	87	303	5	23	15	10
4	2	2	34	3	5	309	22	20	31	5

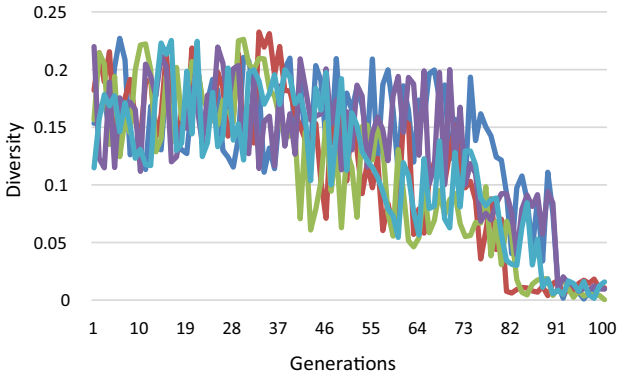

Fig. 9 Stddev versus generations

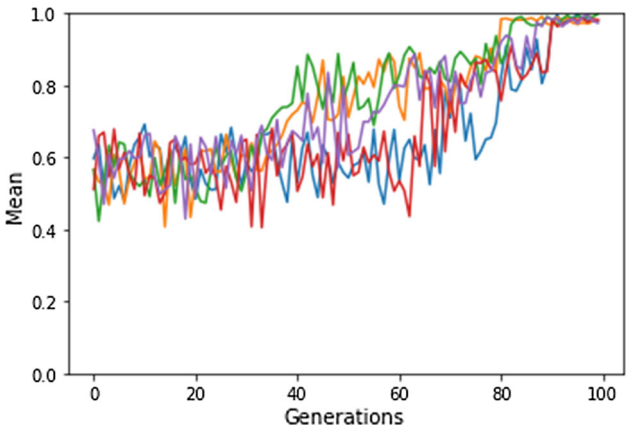

Fig. 10 The average fitness of population versus generations

knowledge about the candidate miRNA biomarkers of cancers. Therefore, classifying the miRNA dataset accurately and extracting insight from unstructured genomic data is of paramount importance. Because of having complexity in data and its characteristics, this is a challenging dataset for analysis. The number of features corresponding to the number of samples is high, besides the data suffer from imbalance. In this paper, the proposed method confirms the efficacy of some of the proposed cancer biomarkers and reveals more reliable candidate miRNAs that can be used as diagnostic or prognostic biomarkers or as therapeutic targets. In this way, a novel method for the classification of the miRNA data in which a three-phase method to classify this dataset was proposed. The first phase divides data into five sub-datasets in order to reduce the complexity of data and make the classes more separable. It was done using a super-class approach that splits data into some super-classes where the classes in each one have the most separability with classes in other ones. In the second phase, a feature selection method was applied to each super-class, selecting features having more ability to distinguish classes and eliminating obscure features. A CNN classifier for classification of cancer types was utilized in the last phase, which uses a GA to highlight optimized hyper-parameters of CNN. In this research to simplify the complication of calculating CNN and reduce the time of computations, GPU was recommended to do the operations in a parallel way.

According to the results, the proposed method outperformed the state-of-the-art research works in terms of classification accuracy. It is mostly because of the two tasks carried out in the data pre-processing phase, namely the super-class approach and feature selection. The first one causes to reduce the complexity of data by splitting it into some sub-datasets which are easier to classify while the latter one helps the convolutional neural network to concentrate on features having more importance and correlation with the class label. The feature selection and convolutional neural network algorithm were applied to each subset due to different characteristics of each subset and for each super-class, a model was trained. In other words, the proposed method introduces a two-step classification approach that each sample should be first classified into its corresponding super-class and then the appropriate CNN is used to predict the label of the sample. The

experimental results reveal that the proposed method can achieve more efficient results than 19 recent machine learning methods.

Choosing the number of layers in CNN is a common problem in many deep learning methods. An overly large number of layers may cause too much computational complexity. On the other hand, knowledge will not be fully extracted when the number of layers is too small. Therefore, an appropriate number of layers may impact the performance and running time of our method. Three layers for CNN are considered in this study after several tests. Other parameters of CNN are achieved by GA. Also, the number of features has a great impact on the CNN classifier. On the other hand, this study aims to classify cancers accurately with a minimum number of features. Due to several experimental tests, we found that with under 300 features, CNN cannot extract knowledge from the dataset and more than 300 features do not effect on the accuracy of CNN. Therefore, the most significant 300 features have been chosen from the dataset by the Fisher ratio algorithm.

In further studies, a number of factors which could influence the reliability of the data should be carefully considered, including confounding factors such as patient age, gender and ethnicity and also treatment options and differences in tumor staging and technical factors such as blood storage conditions. Also, in recent studies reports, exosomes have been attributed roles in the spread of miRNA as a contributing factor in the development of several diseases like cancer (Edgar 2016). As exosomes are implicated in cell–cell communication and the transmission of disease states, and explored as a means of drug discovery, the dataset can be expanded as miRNAs from the exosomes of cancer patients.

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest regarding this manuscript.

References

Abdel-Basset M et al (2018) A hybrid whale optimization algorithm based on local search strategy for the permutation flow shop scheduling problem. Future Gener Comput Syst 85:129–145

Abualigah LM et al (2016) A krill herd algorithm for efficient text documents clustering. In: IEEE symposium on computer applications and industrial electronics (ISCAIE). IEEE

Abualigah LM et al (2017) β-hill climbing technique for the text document clustering. In: New Trends in Information Technology (NTIT)–2017, p 60

Abualigah LMQ (2019) Feature selection and enhanced krill herd algorithm for text document clustering. Springer, Berlin

Abualigah L (2020) Multi-verse optimizer algorithm: a comprehensive survey of its results, variants, and applications. Neural Comput Appl 32:12381–12401

Abualigah LMQ, Hanandeh ES (2015) Applying genetic algorithms to information retrieval using vector space model. Int J Comput Sci Eng Appl 5(1):19

Abualigah LM, Khader AT, Hanandeh ES (2018) A novel weighting scheme applied to improve the text document clustering techniques. In: Innovative computing, optimization and its applications. Springer, Berlin, pp 305–320

Aghdam HH, Heravi EJ (2017) Guide to convolutional neural networks, vol 10. Springer, New York, pp 978–983

Alevizos I, Illei GG (2010) MicroRNAs as biomarkers in rheumatic diseases. Nat Rev Rheumatol 6(7):391

Altman NS (1992) An introduction to kernel and nearest-neighbor nonparametric regression. Am Stat 46(3):175–185

Aydilek IB (2018) A hybrid firefly and particle swarm optimization algorithm for computationally expensive numerical problems. Appl Soft Comput 66:232–249

Barger JF, Nana-Sinkam SP (2015) MicroRNA as tools and therapeutics in lung cancer. Respir Med 109(7):803–812

Bartel DP (2004) MicroRNAs: genomics, biogenesis, mechanism, and function. Cell 116(2):281–297

Bishop CM (2006) Pattern recognition and machine learning. Springer, Berlin

Bolón-Canedo V, Sánchez-Marono N, Alonso-Betanzos A (2014) Data classification using an ensemble of filters. Neurocomputing 135:13–20

Boser BE, Guyon IM, Vapnik VN (1992) A training algorithm for optimal margin classifiers. In: Proceedings of the fifth annual workshop on computational learning theory. ACM

Breiman L (1999) Pasting small votes for classification in large databases and on-line. Mach Learn 36(1–2):85–103

Breiman L (2001) Random forests. Mach Learn 45(1):5–32

Breiman L et al (1984) Classification and regression trees. CRC Press, Boca Raton

Brown TA (2007) Genomes 3. Garland Science Pub., New York

Chen X et al (2018) Novel human miRNA-disease association inference based on random forest. Mol Therapy Nucleic Acids 13:568–579

Chin Y-H et al (2017) Music emotion recognition using PSO-based fuzzy hyper-rectangular composite neural networks. IET Signal Process 11(7):884–891

Cox DR (1958) The regression analysis of binary sequences. J R Stat Soc Ser B (Methodol) 20(2):215–232

Crammer K et al (2006) Online passive-aggressive algorithms. J Mach Learn Res 7(Mar):551–585

Edgar JR (2016) Q&A: what are exosomes, exactly? BMC Biol 14(1):46

Friedman JH (2001) Greedy function approximation: a gradient boosting machine. Ann Stat 29:1189–1232

Fujino A, Isozaki H, Suzuki J (2008) Multi-label text categorization with model combination based on f1-score maximization. In: Proceedings of the third international joint conference on natural language processing, vol II

Garzelli A, Capobianco L, Nencini F (2008) Fusion of multispectral and panchromatic images as an optimisation problem. In: Image fusion, p 223

Geurts P, Ernst D, Wehenkel L (2006) Extremely randomized trees. Mach Learn 63(1):3–42

Ghasemzadeh A, Azad SS, Esmaeili E (2019) Breast cancer detection based on Gabor-wavelet transform and machine learning methods. Int J Mach Learn Cybern 10(7):1603–1612

Han S et al (2018) Optimizing filter size in convolutional neural networks for facial action unit recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition

Hastie T et al (2009) Multi-class adaboost. Stat Interface 2(3):349–360
Hearst MA et al (1998) Support vector machines. IEEE Intell Syst Appl 13(4):18–28
Ho TK, Basu M (2000) Measuring the complexity of classification problems. In: Proceedings 15th international conference on pattern recognition, ICPR-2000. IEEE
Holland JH (1992) Adaptation in natural and artificial systems. MIT Press, Cambridge
Hubel DH, Wiesel TN (1959) Receptive fields of single neurones in the cat's striate cortex. J Physiol 148(3):574–591
Hubel D, Wiesel T (1960) Receptive fields of optic nerve fibres in the spider monkey. J Physiol 154(3):572–580
Javaid N et al (2017) A hybrid genetic wind driven heuristic optimization algorithm for demand side management in smart grid. Energies 10(3):319
Jovanovic M et al (2010) A quantitative targeted proteomics approach to validate predicted microRNA targets in *C. elegans*. Nat Methods 7(10):837–842
Karaboga D, Akay B (2009) A comparative study of artificial bee colony algorithm. Appl Math Comput 214(1):108–132
Lewis DP, Jebara T, Noble WS (2006) Support vector machine learning from heterogeneous data: an empirical analysis using protein sequence and structure. Bioinformatics 22(22):2753–2760
Liu X-Q et al (2019) Prediction of long non-coding RNAs based on deep learning. Genes 10(4):273
Lopez-Rincon A et al (2018) Evolutionary optimization of convolutional neural networks for cancer miRNA biomarkers classification. Appl Soft Comput 65:91–100
Lu J et al (2005) MicroRNA expression profiles classify human cancers. Nature 435(7043):834
Mirjalili S, Mirjalili SM, Hatamlou A (2016) Multi-verse optimizer: a nature-inspired algorithm for global optimization. Neural Comput Appl 27(2):495–513
Montavon G, Braun ML, MÃ‡ller K-R (2011) Kernel analysis of deep networks. J Mach Learn Res 12(Sep):2563–2581
Morán-Fernández L, Bolón-Canedo V, Alonso-Betanzos A (2017) Centralized vs. distributed feature selection methods based on data complexity measures. Knowl Based Syst 117:27–45
Öztürk Ş et al (2018) Convolution kernel size effect on convolutional neural network in histopathological image processing applications. In: International symposium on fundamentals of electrical engineering (ISFEE). IEEE
Pedregosa F et al (2011) Scikit-learn: machine learning in Python. J Mach Learn Res 12(Oct):2825–2830
Peralta D et al (2015) Evolutionary feature selection for big data classification: a MapReduce approach. Math Probl Eng. https://doi.org/10.1155/2015/246139
Pian C et al (2020) Discovering cancer-related miRNAs from miRNA-target interactions by support vector machines. Mol Therapy Nucleic Acids 19:1423–1433
Potharaju SP, Sreedevi M (2019) Distributed feature selection (DFS) strategy for microarray gene expression data to improve the classification performance. Clin Epidemiol Glob Health 7(2):171–176
Rajabioun R (2011) Cuckoo optimization algorithm. Appl Soft Comput 11(8):5508–5518
Sabzehzari M, Naghavi M (2018) Phyto-miRNA: a molecule with beneficial abilities for plant biotechnology. Gene 683:28–34
Salem H, Attiya G, El-Fishawy N (2017) Classification of human cancer diseases by gene expression profiles. Appl Soft Comput 50:124–134
Sarbazi-Azad S, Abadeh MS (2018) Gene selection for cancer classification from microarray data using data overlap measure. In: 25th National and 3rd international Iranian conference on biomedical engineering (ICBME). IEEE
Sarbazi-Azad S, Abadeh MS, Abadi MIN (2018) Feature selection in microarray gene expression data using fisher discriminant ratio. In: 8th International conference on computer and knowledge engineering (ICCKE). IEEE
Scherer D, Müller A, Behnke S (2010) Evaluation of pooling operations in convolutional architectures for object recognition. In: International conference on artificial neural networks. Springer, Berlin
Sherafatian M (2018) Tree-based machine learning algorithms identified minimal set of miRNA biomarkers for breast cancer diagnosis and molecular subtyping. Gene 677:111–118
Soon FC et al (2017) Hyper-parameters optimisation of deep CNN architecture for vehicle logo recognition. IET Intell Trans Syst 12(8):939–946
Tibshirani R et al (2002) Diagnosis of multiple cancer types by shrunken centroids of gene expression. Proc Natl Acad Sci 99(10):6567–6572
Tikhonov AN (1943) The stability of inverse problems. Dokl Akad Nauk SSSR 39:195–198
Torres R, Judson-Torres RL (2019) Research techniques made simple: feature selection for biomarker discovery. J Investig Dermatol 139(10):2068–2074
Vasudevan S, Tong Y, Steitz JA (2007) Switching from repression to activation: microRNAs can up-regulate translation. Science 318(5858):1931–1934
Wang Y, Zhang H, Zhang G (2019) cPSO-CNN: an efficient PSO-based algorithm for fine-tuning hyper-parameters of convolutional neural networks. Swarm Evol Comput 49:114–123
Yang X-S (2012) Flower pollination algorithm for global optimization. In: International conference on unconventional computing and natural computation. Springer, Berlin
Ye Z, Sun B, Xiao Z (2020) Machine learning identifies 10 feature miRNAs for Lung squamous cell carcinoma. Gene 749:144669
Yoon S et al (2019) Biclustering analysis of transcriptome big data identifies condition-specific microRNA targets. Nucleic Acids Res 47:e53
Young SR et al (2015) Optimizing deep learning hyper-parameters through an evolutionary algorithm. In: Proceedings of the workshop on machine learning in high-performance computing environments
Zhang Y-H et al (2020) Identifying circulating miRNA biomarkers for early diagnosis and monitoring of lung cancer. Biochim Biophys Acta (BBA) Mol Basis Dis 1866:165847