Expert Systems With Applications 164 (2021) 113981

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

A k-NN method for lung cancer prognosis with the use of a genetic algorithm for feature selection

Negar Maleki [a], Yasser Zeinali [b], Seyed Taghi Akhavan Niaki [b,*,1]

[a] Department of Industrial Engineering, Faculty of Engineering, University of Tehran, Iran
[b] Department of Industrial Engineering, Sharif University of Technology, Tehran, Iran

ARTICLE INFO

Keywords:
Lung cancer
Cancer staging diagnosis
Data mining
Genetic algorithm
Feature selection
k-NN technique

ABSTRACT

Lung cancer is one of the most common diseases for human beings everywhere throughout the world. Early identification of this disease is the main conceivable approach to enhance the possibility of patients' survival. In this paper, a k-Nearest-Neighbors technique, for which a genetic algorithm is applied for the efficient feature selection to reduce the dataset dimensions and enhance the classifier pace, is employed for diagnosing the stage of patients' disease. To improve the accuracy of the proposed algorithm, the best value for k is determined using an experimental procedure. The implementation of the proposed approach on a lung cancer database reveals 100% accuracy. This implies that one could use the algorithm to find a correlation between the clinical information and data mining techniques to support lung cancer staging diagnosis efficiently.

1. Introduction

Diagnosing a disease is an extremely complex assignment and numerous tests are usually required on the patients to reach a precise conclusion. This can lead us to use analytic devices, planned to help the doctors in their decisions. Early determination lessens the treatment time and may save lives. One of these diseases is the lung malignant growth, which happens when the cells in tissues of the lung develop in an uncontrolled way. This growth can spread beyond the lung by the process of metastasis into nearby tissue or other parts of the body. The vast majority (85%) of cases of lung cancer are due to long-term tobacco smoking and about 10–15% of the cases occur in people who have never smoked (Thun et al., 2008). These cases are often caused by a combination of genetic factors and exposure to radon gas, asbestos, second-hand smoke, or other forms of air pollution. Lung cancer may be seen on chest radiographs and computed tomography (CT) scans. The diagnosis is confirmed by biopsy which is usually performed by bronchoscopy or CT-guidance. Lung malignancy is a one-of-its-sort of disease that prompts 1.61 million death in the world every year (Li et al., 2018). Lung malignant growth is situated second among guys and tenth among females (Naresh & Shettar, 2014). The survival rate is usually higher if the malignancy is analyzed at the starting stages. That is why the early disclosure of lung malignant growth is of significant importance, based on which approximately 80% of the patients are analyzed successfully just at the inside or moved period of the disease (Wutsqa & Mandadara, 2017).

Machine learning utilizes scientific algorithms to distinguish patterns in extensive datasets and to iteratively enhance in playing out this recognizable proof with extra information. These algorithms are generally used in various spaces and different applications, for instance, commercial, protection, fund, internet-based life, and misrepresentation discovery, getting to different types of information gathered continuously and over numerous sources. As patient information is inaccessible for open investigation most of the time, utilizing these strategies to assess illness results can be a challenging task (Lynch et al., 2017).

In this paper, a machine learning method is applied to investigate information regarding lung malignancy, to assess the prescient intensity of these systems. To this aim, a k-Nearest-Neighbors (k-NN) algorithm is first developed to predict lung cancer in its early stage. As the feature selection algorithm can affect the performance of the kNN model, a genetic algorithm (GA) is utilized to optimize the model used to predict. This enables the model to achieve better accuracy in the prediction and prognosis stages. Besides, the value of the parameter k in the kNN algorithm is determined experimentally using an iterative approach. In the end, the performance of the proposed algorithm is assessed when it applies to a lung-cancer database.

* Corresponding author.
E-mail addresses: maleki.negar@ut.ac.ir (N. Maleki), yasser.zeinali@gmail.com (Y. Zeinali), Niaki@sharif.edu (S.T.A. Niaki).
[1] P.O. Box 11155-9414 Azadi Ave., Tehran 1458889694, Iran.

https://doi.org/10.1016/j.eswa.2020.113981
Received 6 July 2019; Received in revised form 4 September 2020; Accepted 7 September 2020
Available online 11 September 2020
0957-4174/© 2020 Elsevier Ltd. All rights reserved.

The rest of this paper is structured as follows. Section 2 provides an overview of what had been done in the literature on lung cancer and which algorithms had been used for cancer diagnosis. Section 3 meticulously details the proposed techniques, and in Section 4 and 5 the performance of the proposed algorithm is analyzed. Finally, Section 6 concludes this work and recommends future works.

2. Literature review

Machine learning involves several algorithms such as k-Nearest Neighbors (kNN), support vector machine (SVM), Naive Bayes (NBs), classification tree (C4.5), gradient boosting machines (GBM), etc. While each of these algorithms processes data differently, in this section, a few recently proposed machine learning candidates in the area of malignant growth finding are reviewed chronologically.

Chen et al. (2013) presented a fuzzy system using kNN (FkNN) for Parkinson's disease (PD) diagnosis. Besides, they used the principal component analysis to find the most discriminative features on which the optimal FkNN model was built. They compared their system with the SVM algorithm and found that their proposed method performed better. The best classification accuracy of their FkNN reached to 96.07%.

Odajima & Pawlovsky (2014) declared that the precision of the kNN method changes with the number of neighbors and with the level of information utilized for classification. Meanwhile, they showed details about the variation of the maximum and the minimum values of the accuracy with the classification set sizes and the number of neighbors.

Lynch et al. (2017) applied some supervised learning classification techniques such as linear regression, decision trees, GBM, SVM, and a custom ensemble to the SEER database to order lung cancer patients regarding survival. The outcomes demonstrated that among the five individual models used, the most precise was GBM with a root mean square error (RMSE) value of 15.32. Septiani et al. (2017) compared the performances of C4.5, NBs, and kNN classification algorithms to detect breast cancer diagnosis on 670 data, each with 9 attributes. They showed that while NBs and kNN have the same accuracy of 98.51%, C4.5 is the worst with the accuracy equal to 91.79%. Hashi et al. (2017) employed decision tree and kNN algorithms to diagnose diabetes disease from the Pima Indians Dataset including 768 data, each with 8 attributes and attained 90.43% and 76.96% accuracy, respectively. This implies that the decision tree is the better-supervised method in terms of the classification accuracy in this case. This dataset had been also used in Iyer et al. (2015), Hayashi and Yukita (2016), Sa'di et al. (2015) and Huang et al. (2015) where they applied the decision tree method and attained 76.96%, 83.83%, 76.52%, and 62.17% accuracies, respectively. Khateeb and Usman (2017) used NB, kNN, J48, and bagging classifiers/ML classification techniques on a heart disease dataset consisting of 303 instances, each with 14 features. They divided their experimental outcomes into 6 cases and found the highest accuracy of 79.20% by the kNN classifier that utilizes all 14 attributes. Moreover, Tayeb et al. (2017) applied kNN as well to datasets compiled by the University of California to analyze two conditions (chronic kidney failure and heart disease) with an accuracy of roughly 90%.

Pradeep and Naveen (2018) used SVM, NBs, and C4.5 techniques on the North Central Cancer Treatment Group (NCCTG) lung cancer data set to help specialists for better conclusions for cancer survivability rate. The results show that C4.5 performs better in foreseeing lung malignancy with increment in the training data set. Alharbi (2018) employed a combined genetic-fuzzy algorithm to diagnose lung cancer. He applied the algorithm on 32 patients with 56 attributes without any reduction in dimensions and attained 97.5% accuracy with a 93% confidence. Cherif (2018) developed a new solution to accelerate the kNN algorithm dependent on clustering and attribute separating on the breast cancer database. He compared his proposed algorithm with other classification techniques such as SVM, Artificial Neural Network (ANN), NBs, and kNN. The dataset was isolated into 5 subsets of 113 occurrences, based on which the F-Measure of each technique was calculated five times to achieve an average F-Measure for each. The results demonstrated that while ANN performed the best, its execution time was 2.2 times higher than the proposed algorithm. Joshi and Mehta (2018) employed a well-known machine learning algorithm (kNN) to examine its execution on the Wisconsin diagnostic breast cancer dataset. The dataset involved 569 instances with 32 attributes and 2 classes. They used two essential dimensionality reduction strategies (principal component analysis (PCA) and linear discriminant analysis (LDA) and showed that kNN with LDA technique worked better than kNN and kNN with PCA with the accuracies 97.06%, 95.29%, and 95.88%, respectively. Akben (2018) utilized kNN, SVM, and NBs to pre-processed data in order to detect chronic kidney disease (CKD). They first used the methods on raw data and figured out that the classification was not accurate enough to encourage medicinal practitioners. As such, they employed the methods after the data was pre-processed by the k-means clustering approach. The outcomes demonstrated that the accuracy was increased significantly, especially, for the kNN classifier which reached 96%.

Lakshmanaprabu et al. (2019) developed a hybrid algorithm involving an optimal deep neural network (ODNN) and a linear discriminate analysis (LDA) to classify lung nodules as either malignant or benign. In their work, the ODNN was first used to extract important features from computed tomography (CT) lung images. Then, LDA was applied to reduce the dimensionality of the features. Finally, a modified gravitational search algorithm was utilized to optimize the ODNN. The sensitivity, specificity, and accuracy of their algorithm were shown to be 96.2%, 94.2%, and 94.56%, respectively. Recently, Alirezaei et al. (2019) deployed four bi-objective *meta*-heuristic algorithms (multi-objective firefly (MOFA), multi-objective imperialist competitive algorithm (MOICA), non-dominated sorting genetic algorithm (NSGA-II), and multi-objective particle swarm optimization (MOPSO)) to determine the least number of attributes with the highest classification accuracy rate. Because of the importance of data quality, first of all, they utilized some preprocessing methods. Then SVM was used as a classifier. Among the above *meta*-heuristics, MOFA was the best with 95.12% accuracy.

As a supervised classifier, the K-Nearest Neighbor is used in this paper one more time on an available data set to predict lung cancer in its early stage. As Odajima and Pawlovsky (2014) showed that different values of the parameter k in the kNN algorithm affect the results significantly, a novel approach in the Python environment is developed to find the best value of k. Furthermore, a genetic algorithm (GA) is utilized to find the best features. In what comes the proposed approach is described in detail.

3. The proposed approach

The proposed methodology is an enhancement of the kNN method. This section briefly provides a background for the kNN method. We then demonstrate how GA can improve the accuracy of the kNN method.

3.1. k-Nearest-Neighbors Classifiers

The kNN classifier has been widely used in the area of pattern recognition. Nearest-neighbor classifiers depend on learning by relationship, that is, by contrasting a given test tuple and preparing tuples that are similar to it. The preparation tuples are portrayed by n traits. Each tuple refers to a point in a $n-$dimensional space; hence, all the preparation tuples are put away in a $n-$dimensional example space. At the point when given an obscure tuple, a k-closest neighbor classifier looks the example space for the K preparing tuples that are nearest to the obscure tuple. These k-preparing tuples are the k "closest neighbors" of the obscure tuple.

Closeness in the kNN algorithm is characterized by a separation metric, for example, Euclidean distance. The Minkowski distance between two tuples, say, $X_1 = (x_{11}, x_{12}, \cdots, x_{1n})$ and $X_2 = (x_{21}, x_{22}, \cdots, x_{2n})$, is:

$$dist(X_1, X_2) = \left(\sum_{i=1}^{n} |x_i - x_j|^p \right)^{1/p} \quad (1)$$

For each numeric characteristic of a data point, the distinction between the relating estimations of that characteristic in the tuple X_1 and X_2 is first realized by squaring the distance, and then adding them up for all characteristics. The square root is next taken on the aggregate gathered separation tally. Commonly, the estimation of each characteristic is normalized before utilizing Eq. (1). It will probably enhance the accuracy rate of the algorithm.

An appropriate value for k can be obtained experimentally. Beginning with k = 1, a test set is utilized in Python to evaluate the error rate of the classifier. This procedure can be repeated each time by augmenting k to include one more neighbor. The k esteem that gives the best base error rate is chosen (Han et al., 2011).

3.2. Genetic algorithm implementation

GA is a heuristic search method. It can be utilized to search for an optimal solution into spaces that are excessively expansive to be comprehensively looked at. This algorithm is a method for solving both constrained and unconstrained optimization problems that are based on natural selection, the process that drives biological evolution. It has numerous applications in natural sciences, mathematics, computer science, finance and economics, industry, management, and engineering, among others. It can mirror the procedure of characteristic determination in the kNN algorithm. There are five phases in a genetic algorithm:

1. Initial population
2. Fitness function
3. Selection
4. Crossover
5. Mutation

The GA technique is an iterative method that includes a population communicating to a look space to find answers for an issue by a limited series of images, called the genome, gathered in a chromosome (solution). The fundamental GA continues as pursues: an underlying population of chromosomes is produced indiscriminately or heuristically. In each developmental advance (generation), the chromosomes in the population are decoded and assessed by a fitness function that portrays the streamlining issue in the search space. To shape another population (the next generation), chromosomes are chosen by their fitness. Here, numerous choices are available, one of the least complex being the fitness proportionate choice, where chromosomes are chosen with a likelihood corresponding to their relative fitness. This guarantees the normal number of times a selected individual is around corresponding to its relative performance in the population. Therefore, high-fitness chromosomes stand a superior opportunity to recreate and convey new individuals to the population, while low-fitness chromosomes will not.

New chromosomes are brought into the population by hereditary operations called crossover and mutation. The crossover operation is performed with a likelihood between two chosen individuals (parents) trading parts of their genomes to shape two new chromosomes (offspring). Meanwhile, the mutation operation averts untimely union to nearby optima by randomly examining new focuses in the hunt space; it is performed by flipping bits at arbitrary, with some low likelihood. GA is a stochastic iterative process, which is not ensured to find the optimum point. Moreover, the stopping condition might be indicated as a maximal number of generations or a desired value of the fitness.

3.3. Performance criteria

Accuracy is one of the performance criteria with several meanings in different areas. In the classification methods, however, accuracy is defined as a statistical measure of how well a binary classification test correctly identifies or excludes a condition. That is, the accuracy is the proportion of true results (both true positives and true negatives) among the total number of cases examined in the experiment. Eq. (2) is used to quantify binary accuracy:

$$Accuracy = (TP + TN)/(TP + TN + FP + FN) \quad (2)$$

where, TP = True positive, FP = False positive, TN = True negative, FN = False negative.

All of the above quantities can be extracted using the confusion matrix; a table that is often used to describe the performance of a classification model on a set of test data for which the true values are known (Melamed et al., 2003).

Other performance criteria are “sensitivity” and “specificity”, also known in statistics as a classification function, which are widely used in medicine and bioinformatics studies. The sensitivity or the recall measures the proportion of true positives that are correctly identified and the specificity also measures the proportion of true negatives in experiments. Eqs. (3) and (4) define these measures.

$$Sensitivity = TP/(TP + FN) \quad (3)$$

$$Specificity = TN/(TN + FP) \quad (4)$$

3.4. Feature selection

The first goal in the proposed feature selection method is to reach at least the same accuracy rate as the whole features provide. The second goal is to improve the accuracy rate. Here, not only gathering extensive information on the features costs too much in terms of both the time and money, but also extra information results in wastage of time in classifying and diagnosis. As such, it is better to reduce the dimension in terms of the number of features to get a better response and to find a better correlation between the features and the outcomes.

The genetic algorithm is a technique to select the best features. In this technique, a binary random vector *VectorS* consisting of the features is first generated using Eq. (5) (Pawlovsky & Hiroki, 2017):

$$Vector_{(s_j)}: \quad s_j = Y_i; \quad Y_i = \begin{cases} 1 \ ; & if \ Vector_{s_j} \ contains \ feature \ i \\ 0 \ ; & otherwise \end{cases} \quad (5)$$

Then, an objective function based on the misclassification performance criterion is defined for any selected combination of the features. This objective function works as a penalty function that should be minimized to find the best combination of the features. Here, the misclassification rate (mcr) is simply $mcr = 1 - accuracyrate$ and is obtained using Eq. (6), where m is the number of classification-targets and a_{ij} is the number of cases the target i is classified as the target j using the classification method. The a_{ij} elements construct a matrix in (7) called the confusion matrix that depends on the problem as well as the dataset (Pawlovsky & Hiroki, 2017):

$$mcr = \frac{\sum a_{ij} - \left[\sum a_{ij}; (i = j) \right]}{\sum a_{ij}}; \quad i, j = 1, 2, \cdots, m \quad (6)$$

$$\begin{pmatrix} a_{11} & \cdots & a_{1m} \\ \vdots & \ddots & \vdots \\ a_{m1} & \cdots & a_{mm} \end{pmatrix}_{m \times m} \quad (7)$$

Now, the objective function to be minimized is a weighted sum of the mcr and n_f (number of selected features) defined as

$$MinZ = w_1 * mcr + w_2 * n_f \quad (8)$$

Dividing the right-hand-side of Eq. (8) byw_1, we have:

$$MinZ = mcr + w_2/w_1 * n_f \quad (9)$$

Table 1
Attributes (features) involved in the dataset.

Age (1)	Gender (2)	Air Pollution (3)	Alcohol use (4)	Dust Allergy (5)	Occupational Hazards (6)
Genetic Risk (7)	Coughing of Blood (8)	Fatigue (9)	Weight Loss (10)	Smoking (11)	Wheezing (12)
Chest pain (13)	Chronic Lung Disease (14)	Balanced Diet (15)	Obesity (16)	Shortness of Breath (17)	Passive Smoker (18)
Swallowing Difficulty (19)	Clubbing of Finger Nails (20)	Frequent Cold (21)	Dry Cough (22)	Snoring (23)	

Assuming $w_2/w_1 = W$, the objective function becomes:

$$MinZ = mcr + W^*n_f \tag{10}$$

Now, W can be defined as:

$$\begin{matrix} W \propto mcr \rightarrow \\ W = \beta^* mcr \rightarrow \\ MinZ = mcr + \beta^* mcr^* n_f \end{matrix} \tag{11}$$

This leads to:

$$MinZ = mcr(1 + \beta^* n_f) \tag{12}$$

where β can be defined as a penalty for having an additional feature ($0 \le \beta \le 1$).

Using this objective function, GA tries to find the best combination of the features with the minimum number of features that minimize both the cost and the misclassification rate. Here, the stopping criterion to end the iterations in GA is chosen to be a predefined number of iterations.

3.5. Data description

The box shown in this figure is "Dataset". The importance of the dataset is an undeniable part of the research because it affects the final result. The lung cancer dataset under consideration that is obtained from the Data world site (https://data.world/cancerdatahp/lung-cancer-data) contains 1000 samples, each with 23 features shown in Table 1. The targets in this dataset are the risk levels of the lung cancer suffering that are classified in 3 levels of Low, Medium, and High (see Table 2).

4. The framework of the proposed lung cancer diagnosis procedure

The general structure of the proposed diagnosing procedure is depicted in Fig. 1. Having the dataset, the next box in Fig. 2 is to check whether or not any pre-processing is needed to remove missing values or substituting them with appropriate data. The rows of datasets who were imperfect could have been deleted but we decided to automatically fill the missing values using the software function for utilizing the mean of the other values for them. Then, in the next box, GA applies to the clean dataset to find the best combination of the features that provide the highest correlation between the features and the targets. To this aim, the vector *VectorS* is obtained in Fig. 3.

While the maximum number of iterations is set to 10, after the fourth iteration the cost function value converges to 0.53266 as it is shown in Fig. 4. Here, the population size is 20 and the probability of the cross-over operator is chosen to be 0.7, the mutation probability is set 0.02, based on which the number of offspring generated is 14 and the number of mutants is 6. Besides, the roulette wheel method selects the parents in all operations.

$$Number\ \ of\ \ offsprings = 2^*round\left(\frac{Crossover^*Pop}{2}\right) \tag{13}$$

$$Number\ \ of\ \ mutants = round(Mutation\ \ percentage^*Pop) \tag{14}$$

After applying GA, the kNN classifier is applied to the training dataset to learn how to recognize the targets. The kNN classifier needs data for the k-Nearest Neighbors to classify them to detect the targets correctly. As the parameter k affects the classification performance significantly, an iterative approach in Python is used to find an appropriate value of k. The appropriate value for k can be obtained experimentally as we said before. Beginning with k = 1, a test set is utilized in Python to evaluate the classifier. This procedure can be repeated each time by augmenting k to include one more neighbor. At last, the best k will be chosen to utilize it in the model. The outcomes are shown in Figs. 5 and 6.

It is clear from Figs. 5 and 6 that not only GA improves the accuracy rate of the kNN algorithm, but also k = 7 and k = 6 provide the maximum accuracies when GA is used on not used, respectively.

In the final box depicted in Fig. 2, the trained kNN accuracies when it applies to the testing datasets are compared to each other to evaluate the performance of the proposed methodology.

5. Comparison analysis

In this section, the performance of the proposed methodology in terms of the accuracy is compared to the ones of three different approaches including the decision tree, the kNN without the proposed feature-selection method (GA) and without configuring the k parameter, and the kNN with the proposed feature-selection approach that involves the k-parameter configuration. To this aim, 500 patients are chosen randomly from the lung cancer dataset to be used in all methods.

The confusion matrix of the decision trees approach is shown in Table 3. As seen in this table, the accuracy of the decision tree method is obtained as 95.2%.

The second method is the kNN without the use of neither the feature selection method nor the k-parameter configuration approach. The confusion matrix of this method is brought in Table 4.

To analyze the applicability domain of the experiment, one needs to partition the dataset into two different parts (train, and test), based on which the difference between the accuracy rates can be analyzed. We devoted 80% of the dataset to the training and 20% to the test set. This experiment resulted in an accuracy rate of 100 percent for the training set and an accuracy rate of 96.2 percent for the test set. These results are obtained by implementing the kNN with the K equal to 10. It is clear that the difference between these two accuracy rates is not significant; therefore, the model is applicable.

The results in Table 4 also indicate that the accuracy of the kNN approach without using the GA algorithm to select the best combination of the features is 96.2 percent when the k-parameter is set to 10. Although this accuracy is better than the one obtained using the decision tree method (95.2% in Table 3), it is further raised to 99.8%, when K is changed from 10 to 6. The confusion matrix of this approach is shown in Table 5.

Moreover, when GA is implemented the accuracy gets even better. Table 6 shows this conclusion. In other words, the accuracy rate of the kNN method with k = 6 neighbors that uses the feature selection algorithm is the highest.

We also implemented the 10 fold cross-validation for the training set to obtain the scores as follows.

1	1	1	1	0.96078431	1	0.95918367	0.97959184	1	0.97916667

Table 2
Parameters estimation of GA.

Max Iteration	Pop.	% Crossover	% Mutation	Time(s)	Cost function	Selected Vector
10	20	0.7	0.3	912.5702	0.51265	[2,7,10,16,17,19]
10	20	0.8	0.3	425.12316	0.54523	[10,11,15,16,17,19,20]
10	20	0.9	0.3	400.12805	0.54523	[10,11,15,16,17,19,20]
10	20	0.7	0.4	512.17854	0.56425	[2,10,13,14,15,16,17,19]
10	20	0.8	0.4	607.71827	0.56418	[2,10,13,14,15,16,17,18,19]
10	20	0.9	0.4	894.39541	0.56425	[2,10,13,14,15,16,17,19]
10	20	0.7	0.5	413.03148	0.56425	[2,10,13,14,15,16,17,19]
10	20	0.8	0.5	397.124976	0.56425	[2,10,13,14,15,16,17,19]
10	20	0.9	0.5	801.900148	0.56425	[2,10,13,14,15,16,17,19]
10	50	0.7	0.3	759.214019	0.5257	[2,10,15,16,17,19]
10	50	0.8	0.3	989.107872	0.56418	[2,10,13,14,15,16,17,18,19]
10	50	0.9	0.3	1251.439019	0.53421	[10,11,15,16,17,18,19]
10	50	0.7	0.4	1624.20197	0.56418	[2,10,13,14,15,16,17,18,19]
10	50	0.8	0.4	1724.219054	0.56418	[2,10,13,14,15,16,17,18,19]
10	50	0.9	0.4	2078.10536	0.55425	[2,10,13,14,15,16,17,19]
10	50	0.7	0.5	1954.028514	0.56418	[2,10,13,14,15,16,17,18,19]
10	50	0.8	0.5	1207.714546	0.56418	[2,10,13,14,15,16,17,18,19]
10	50	0.9	0.5	2007.167903	0.55425	[2,10,13,14,15,16,17,19]
10	80	0.7	0.3	1627.21883	0.56418	[2,10,13,14,15,16,17,18,19]
10	80	0.8	0.3	1405.15904	0.56418	[2,10,13,14,15,16,17,18,19]
10	80	0.9	0.3	2104.01791	0.56418	[2,10,13,14,15,16,17,18,19]
10	80	0.7	0.4	1721.8028	0.56418	[2,10,13,14,15,16,17,18,19]
10	80	0.8	0.4	2845.677454	0.56418	[2,10,13,14,15,16,17,18,19]
10	80	0.9	0.4	2157.01385	0.56418	[2,10,13,14,15,16,17,18,19]
10	80	0.7	0.5	2278.02138	0.56418	[2,10,13,14,15,16,17,18,19]
10	80	0.8	0.5	1984.98026	0.56418	[2,10,13,14,15,16,17,18,19]
10	80	0.9	0.5	2310.14806	0.56418	[2,10,13,14,15,16,17,18,19]

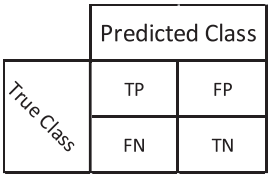

Fig. 1. Confusion Matrix.

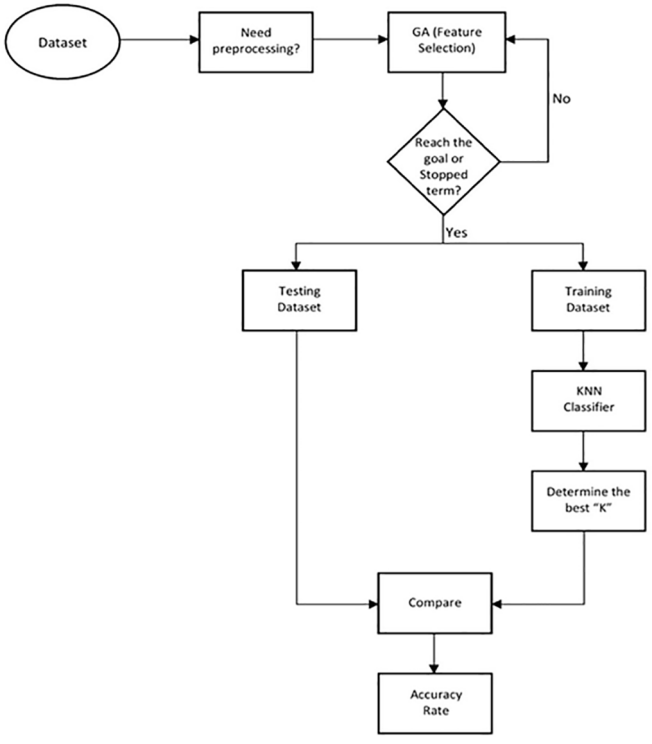

Fig. 2. The framework of the proposed lung cancer diagnosis procedure.

$$Vector\ S = [2, 7, 10, 16, 17, 19]; n_f = 6$$

Fig. 3. GA's features selection.

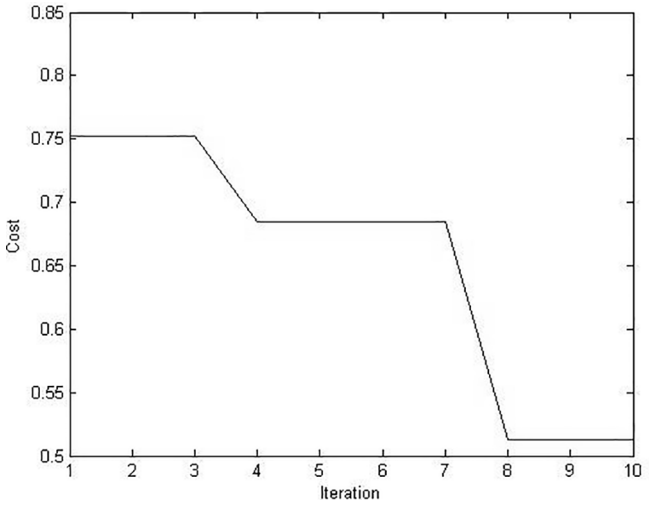

Fig. 4. The best cost function values (0.51265).

As it is clear in the above figure, scores are either 1 or very close to 1.

As the last comparison, here we have different CPU times for the methodologies in Table 7. The results in this table show that not only GA affects the accuracy of k-NN, but also decrease the CPU time significantly.

6. Conclusion and future work

While there are many machine-learning methods available in the literature whose performances depend on different aspects including the dataset they are applied on, in this paper, a machine-learning method called kNN was hybridized with a feature-selection genetic algorithm to

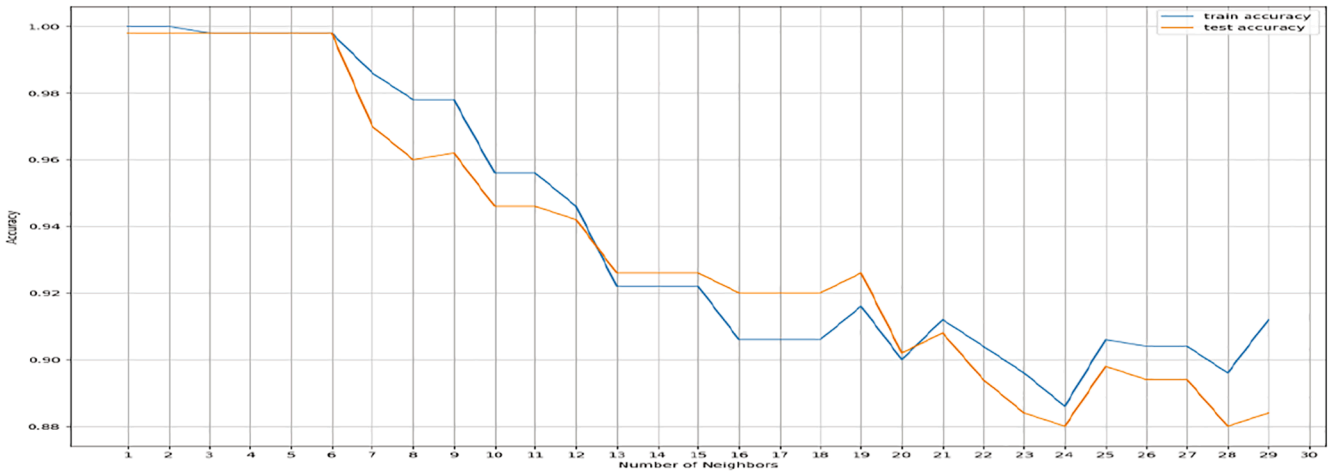

Fig. 5. kNN accuracy rate before applying GA using different "K".

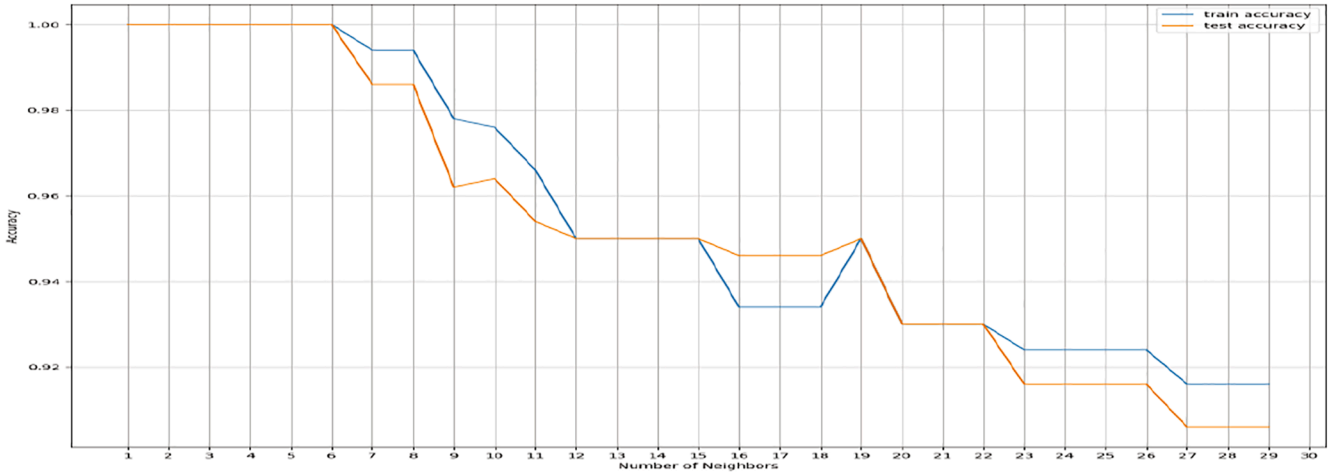

Fig. 6. kNN accuracy rate after GA using different "K".

Table 3
The confusion matrix of the decision tree method.

From\To	High	Low	Medium	Total	%Accuracy
High	167	6	0	173	96.53%
Low	0	167	6	173	96.53%
Medium	0	12	142	154	92.21%
Total	167	185	148	500	95.20%

Table 4
The confusion matrix of the kNN without GA ("K" is set to 10).

From\To	High	Low	Medium	Total	%Accuracy
High	150	0	7	157	95.54%
Low	0	151	5	156	96.79%
Medium	0	7	180	187	96.25%
Total	150	158	192	500	**96.20%**

Table 5
The confusion matrix of the kNN without GA ("K" is set to 6).

From\To	High	Low	Medium	Total	%Accuracy
High	150	1	0	151	99.35
Low	0	166	0	166	100.00%
Medium	0	0	183	183	100.00%
Total	150	167	183	500	**99.80%**

Table 6
The confusion matrix of the kNN with GA ("K" is set to 6).

From\To	High	Low	Medium	Total	%Accuracy
High	151	0	0	151	100.00%
Low	0	166	0	166	100.00%
Medium	0	0	183	183	100.00%
Total	151	166	183	500	100.00%

classify the risks of lung cancer patients in three levels of low, medium, and high. The objective of using GA was to determine the best combination of the features that minimize the overall miscalculation of the kNN method. Moreover, the best value for the number of neighbors in the kNN algorithm was determined using an algorithm coded in Python. It was shown that when the kNN method is hybridized with a feature-selection algorithm, the classification accuracy increases significantly. As it mentioned before, 6 features had been chosen via the GA algorithm, that were [2, 7, 10, 16, 17, 19], the cost function value converged at the fourth iteration and it lasted about 912.5702 s to run the program and also the best value for k was 6. All computations were performed on a laptop with 2.20 GHz CPU and 2 GB RAM. These experiments and results were analyzed carefully by a lung cancer specialist. Moreover, the specialist analyzed the outcomes using some patients' clinical data and compared their real condition with the class that the model devoted to that patient.

Future works may involve the use of other machine-learning classification algorithms or employing other population-based feature

Table 7
Comparison of the models and their results.

Algorithm	Accuracy	Time(s) (Only ML Algorithm)
Decision Tree	95.40%	0.015996
k-NN (k = 10) Euclidian Distance	96.40%	0.0312
k-NN (k = 6)	99.80%	0.0313
GA first, then k-NN	100%	0.0156

selection *meta*-heuristics and compare their performances to the one obtained by the proposed approach.

CRediT authorship contribution statement

Nagar Maleki: Methodology, Software, Validation, Formal analysis, Investigation, Data curation, Writing - original draft, Visualization. **Yasser Zeinali:** Methodology, Software, Validation, Formal analysis, Investigation, Data curation, Writing - original draft, Visualization. **Seyed Taghi Akhavan Niaki:** Conceptualization, Resources, Writing - review & editing, Supervision, Project administration.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Akben, S. B. (2018). Early stage chronic kidney disease diagnosis by applying data mining methods to urinalysis, blood analysis and disease history. *IRBM, 39*(5), 353–358.

Alharbi, A. (2018). An automated computer system based on genetic algorithm and fuzzy systems for lung cancer diagnosis. *International Journal of Nonlinear Sciences and Numerical Simulation, 19*(6), 583–594.

Alirezaei, M., Niaki, S. T. A., & Akhavan Niaki, S. A. (2019). A bi-objective hybrid optimization algorithm to reduce noise and data dimension in diabetes diagnosis using support vector machines. *Expert Systems with Applications, 127*, 47–57.

Chen, H.-L., Huang, C.-C., Yu, X.-G., Xu, X., Sun, X., Wang, G., & Wang, S.-J. (2013). An efficient diagnosis system for detection of Parkinson's disease using fuzzy k-nearest neighbor approach. *Expert systems with applications, 40*(1), 263–271.

Cherif, W. (2018). Optimization of K-NN algorithm by clustering and reliability coefficients: Application to breast-cancer diagnosis. *Procedia Computer Science, 127*, 293–299.

Han, J., Pei, J., & Kamber, M. (2011). *Data mining: Concepts and techniques*. Elsevier.

Hashi, E. K., Zaman, S. U., Hasan, R. (2017). An expert clinical decision support system to predict disease using classification techniques. In The proceedings of the 2017 international conference on electrical, computer and communication engineering (ECCE), Cox's Bazar, Bangladesh.

Hayashi, Y., & Yukita, S. (2016). Rule extraction using recursive-rule extraction algorithm with J48 graft combined with sampling selection techniques for the diagnosis of type 2 diabetes mellitus in the Pima Indian dataset. *Informatics in Medicine Unlocked, 2*, 92–104.

Huang, G.-M., Huang, K.-Y., Lee, T.-Yi, Weng, J. (2015). An interpretable rule-based diagnostic classification of diabetic nephropathy among type 2 diabetes patients. In Selected articles from the thirteenth Asia pacific bioinformatics conference (APBC 2015) (Vol. 16, Supplement 1, p. S5).

Iyer, A., Jeyalatha, S., & Sumbaly, R. (2015). Diagnosis of diabetes using classification mining techniques. *International Journal of Data Mining & Knowledge Management Process (IJDKP), 5*(1), 1–14.

Joshi, A., & Mehta, A. (2018). Analysis of K-nearest neighbor technique for breast cancer disease classification. *International Journal of Recent Scientific Research, 9*(4), 26126–26130.

Khateeb, N., Usman, M. (2017). Efficient heart disease prediction system using K-nearest neighbor classification technique. In Proceedings of BDIOT2017 proceedings of the international conference on big data and internet of thing, London, UK (pp. 21–26).

Lakshmanaprabu, S. K., Mohanty, S. N., Shankar, K., Arunkumar, N., & Ramirez, G. (2019). Optimal deep learning model for classification of lung cancer on CT images. *Future Generation Computer Systems, 92*, 374–382.

Li, J., Usevich, K., & Comon, P. (2018). Globally convergent Jacobi-type algorithms for simultaneous orthogonal symmetric tensor diagonalization. *SIAM Journal on Matrix Analysis Applications, 39*(1), 1–22.

Lynch, C. M., Abdollahi, B., Fuqua, J. D., de Carlo, A. R., Bartholomai, J. A., Balgemann, R. N., … Frieboes, H. B. (2017). Prediction of lung cancer patient survival via supervised machine learning classification techniques. *International Journal of Medical Informatics, 108*, 1–8.

Melamed, I. D., Green, R., & Turian, J. (2003). Precision and recall of machine translation. *Paper presented at the companion volume of the proceedings of HLT-NAACL 2003-short papers*.

Naresh, P., & Shettar, R. (2014). Image processing and classification techniques for early detection of lung cancer for preventive health care: A survey. *International Journal on Recent Trends in Engineering Technology, 11*(1), 595–601.

Odajima, K., Pawlovsky, A. P. (2014). A detailed description of the use of the KNN method for breast cancer diagnosis. In Proceedings of the 7th international conference on biomedical engineering and informatics (BMEI), Dalian, China.

Pawlovsky, A. P., & Hiroki, M. (2017). A kNN method for breast cancer prognosis that uses a genetic algorithm for component selection. *Methods, 13*, 181–186.

Pradeep, K. R., & Naveen, N. C. (2018). Lung cancer survivability prediction based on performance using classification techniques of support vector machines, C4. 5 and naive Bayes algorithms for healthcare analytics. *Procedia Computer Science, 132*, 412–420.

Sa'di, S., Maleki, A., Hashemi, R., Panbechi, Z., Chalabi, K. (2015). Comparison of data mining algorithms in the diagnosis of type II diabetes. International Journal on Computational Science Applications 5(5), 1–12.

Septiani, N. W. P., Wulan, R., & Lestari, M. (2017). Breast cancer detection using data mining classification methods. *Journal of Mathematics, 1*(1), 185–191.

Tayeb, S., Pirouz, M., Sun, J., Hall, K., Chang, A., Li, J., … Sager, T. (2017). Toward predicting medical conditions using k-nearest neighbors. *Proceedings of 2017 IEEE International Conference on Big Data (Big Data), Boston, MA, USA*.

Thun, M. J., Hannan, L. M., Adams-Campbell, L. L., Boffetta, P., Buring, J. E., Feskanich, D., … Samet, J. M. (2008). Lung cancer occurrence in never-smokers: An analysis of 13 cohorts and 22 cancer registry studies. *PLoS Medicine, 5*(9), 1357–1371.

Wutsqa, D. U., Mandadara, H. L. R. (2017). Lung cancer classification using radial basis function neural network model with point operation. In Paper presented at the image and signal processing, BioMedical engineering and informatics (CISP-BMEI), 10th international congress on 2017.