ISSN 2063-5346

DNA MICROARRAY FOR CANCER CLASSIFICATION USINGDEEP LEARNING

B Shyamala Gowri[1], S Bhuvaneshwari [2], A Abirami [3], R Dilli Rani[4], K N Anirudh[5] and S Keerthi Shree[6]

Article History: Received: 01.02.2023 **Revised:** 07.03.2023 **Accepted:** 10.04.2023

Abstract

The major cause of death has always been seen as cancer. The challenge is figuring it out as soon as possible. The possibility of preserving them decreases as the stage rises. Microarray gene-based expression profiling technology is one of the most useful methods for managing cancer diagnosis, prognosis, and treatment. The expression of genetic data generally gets tens of thousands of genes for each data point (example). Examining the level of expression of genes using DNA microarray. The area of genetic studies is currently experiencing a surge in interest in technology for a specific organism. Applications for microarray studies in the medical profession include illness prediction and diagnosis, cancer research, and many more. Genetic selection is one of the best ways to deal with this problem. Deep learning is based on neural networks and is a subset of automated learning. The information has been pulled from the vast knowledge of the raw data, which is doing the discriminating and putting it into a framework that people can readily understand. Here, deep learning's primary task is to forecast illnesses. The hidden data sets and models in the medical domain must be extracted in order to obtain the medical data required for learning.

KEYWORDS: DNA, Deep Learning, Gene Sequence, Gene Expression, Microarray, Microarray Data, Cancer Classification, Image processing.

[1,2,3] Assistant Professor, Department of Computer Science And Engineering, Easwari Engineering College, Chennai, Tamil Nadu, India

[4,5,6] UG Scholar, Department of Computer Science And Engineering, Easwari Engineering College, Chennai, Tamil Nadu, India

DOI: 10.31838/ecb/2023.12.s1.168

1. Introduction

DNA microarrays offer a distinctive perspective on DNA biology and, as a result, offer a useful method for assessing living systems. Many biologists utilize the microarray technique to track the degree of genome-wide gene expression in an organism [1]. A valuable tool is provided by image processing (IP) to aid physicians, medical professionals, and radiologists in reaching more precise conclusions [6]. Microarrays are being used to simultaneously assess the interactions of hundreds of genes and build an overall picture of cellular function [5].

A key idea in molecular biology is the regulation of genetic expression, which we use to explain the majority of biological events. This control is frequently precise in terms of cell specialization and time. Development is a term we use to explain how various cell functions including mitosis, migration, differentiation, and death are coordinated through time and space [1]. Among the most frequent and significant uses of gene function genetic analysis is classification, which involves categorizing patient samples into several groups based on the characteristics of their gene's expressions [5].

In cancer research, where early cancer identification is crucial for defining the kind of treatment and the likelihood of survival, microarray data are mostly employed. Microarray Technology (MT) assists scientists in reviewing the 10,000 gene activities in a single trial and obtains crucial data on the functionality of the cell [4].

The DBN-based 104 trained to learn that are being presented exhibit excellent discrimination and prognostic accuracy for tumor and non-cancer 105 data. The recent work [2] also reveals the significance of examining the temporal connections among genes using time - series data 106 microarray data. RNA from the biological material is generated into microarray targets in microarray investigations.

Utilizing certain microarray image processing methods [2]. Pre-processing, which improves alignment of poorly expressed points in the grid and corrects picture rotation, is one category of image processing techniques. Identify each microarray spot's location. Segmentation to extract spot intensity characteristics and conduct categorization of pixels. Normalization of the data for the estimation of levels of gene expression [3].

The categorization of microarray data has been applied using a variety of ML techniques. However, because to the limited size & high complexity of Microarray data, categorization of the data continues to be difficult. Microarray gene expression studies frequently produce a big dataset with few samples because they provide a high number of characteristics for a smaller number of patients.

Given the difficulty and complexity of gene expression data, classification is typically a highly demanding task that necessitates the use of a precise and effective feature selection technique [5]. In [7] it has been discovered that proposed model is able to predict lung cancer with greater accuracy

2. Motivation

- In 2020, cancer will be the main cause of mortality with around 10 million deaths worldwide.
- It is anticipated that 1,918,030 fresh cases of cancer would be found then 609,360 people will die from the disease by 2022.
- Our goal is to make a contribution to the health sector.
- To increase accuracy, it is necessary to investigate the machine learning techniques.

3. Literature Survey

[1] Genes enable physicians to forecast

the kind of data on various cancers using the DNA chip as a data for categorizing various diseases. Datasets demonstrate the significance of using the identical classifier for classifying and choosing genes in order to strengthen the model. They employed four algorithms: Decision tree, Naive bayes NLP, Deep neural network, and out of these, deep neural network delivers results with a higher accuracy , however it is still less accurate than 80.

[2] In this study, time series records were modelled using Dynamic Bayesian Networks (DBNs). Furthermore, the categorization of cancer using DBN-based methods may highlight the value of making use of information about statistically genes and important regulatory chemicals. In terms of learning algorithms, the research has not adequately addressed the use of new information first level, despite the fact that in this study microarray data have been widely used by different classification algorithms for decision-making. To study networks using biological data, particularly time- series genomic measurement, Bayesian techniques have been taken into consideration.

[3] In this study, microarray targets are created using RNA taken from biological samples. RNA or single-stranded DNAs are the targets. From image processing techniques to genomic level estimation, micro- array image analysis. (For instance, the dyes Cy3 and C Image processing methods are divided into three categories: (1) pre-processing, which corrects image rotation and enhances alignment of mildly expressed points;

(2) in the grid orientation, which pinpoints the location of each micro- network point; and (3) segmentation, which performs pixel classification. (that is, identifying the pixels that correspond to the microarray area or (from its immediate backdrop) (4) Data normalization for determining gene expression levels. Patch intensity characteristics extraction.

[4] The strategies for selecting features in cancer data sets are examined in this research. They employed two techniques: search strategy and label status. In information mining and ML strategies employed in the fields of medicine, Poor categorization and diagnosis are frequently caused by genes of thousands (characteristics) vs small amount of sample, then the existence of duplicate systems in data interpretation. Numerous factors to improve accuracy like dimension reduction, etc. were seen. This study helps in selecting the most appropriate feature and method. But plenty of of time and energy is spent in finding that function and method.

[5] This article describes a hybrid model to classifying cancer, and the hybrid model makes use of a number of ML approaches, including: A characteristic selection based on correlation and reduction, a DT classifier that is simple to understand and needs no variable, and variable selection resume (cross-validation) to maximize the maximum depth hyper - parameters are all components of the Pearson correlation coefficient. To evaluate our approach, seven common DNA-based tumor datasets are employed. Several performance metrics, such as classification results, specificity, sensitivity, F1 measure, and AUC, are employed to determine the most useful and relative features utilizing the suggested model. This methodology produced good classification results and was successful in selecting an ideal or nearly ideal subset of instructive and significant genes.

[6] Using machine learning (ML) with image analysis, doctors, medical professionals, and radiologists may use a valuable tool to assist them make more correct judgments. Breast cancer (BC) affects many people worldwide and is among the medical specialties where the application of ML and IP approaches is beginning to take off. In this study, they

classified breast cancer using a machine learning approach. Results are comparatively correct. It is exclusively applied to one kind of cancer.

[7] The photos are segmented using the K-means method. The image's components may be located using this segmentation. Then, machine learning-based classification techniques are applied. ANN, K - nearest neighbors, and number machine learning methods, including RF, were used for the categorization. Medical history data is compared using image processing methods and damaged region. The research uses technologies made feasible by machine learning to provide precise categorization and prognosis of lung cancer. Results are comparatively correct. It has been discovered that the ANN model produces more precise findings for predicting lung cancer. It is exclusively applied to one kind of cancer.

[8] The IAECP framework uses MRI radio imaging in this case to predict colorectal cancer. To investigate cancer in its early phases, both high- and low-level abilities are needed. In this case, utilizing a deep learning network, the entire colon MRI image is analyzed. in order to pinpoint the location, point of birth, and training process that are influenced by cancer. 100,000 histology colorectal images are used in this method. The dataset comprises of a number of split pictures used for experimentation.

4. Existing System

Because there is typically such a big amount of data from DNA microarrays, it is absolutely crucial to evaluate this data in the most effective way possible.

There are several studies available for predicting and categorizing cancer. The DNA microarray appears to be one of the primary methods used in molecular genetics to monitor gene expression., that shows the stage for protein formation controlled by the genome.

For predicting or categorizing it, they employ a variety of deep as well as automated learning techniques. Others employ a variety of image processing methods, either directly from a picture or from an external source like a DNA microarray.

The first step is to gather images. The photos are then preprocessed using a mean filter. Image quality progressively improves as a result of this. When image processing (IP) and machine learning (ML) are combined, a powerful tool that allows medical professionals, radiologists, and other medical professionals to make decisions that are more correct.

Numerous illnesses were discovered using microarray research and machine learning approaches. The K-means algorithm is then used to section the pictures. Every system struggles to select the best algorithm and to be precise.

The process of converting raw data into something useful and consumable is known as data preparation. Data preparation modifies the data's format to enable faster and more effective processing by DM, ML, and other DS applications. Although there are numerous tools and methods for data preparation, we have chosen to concentrate on the following metrics for our dataset:

5. Proposed System

In this paper, we define machine learning for DNA microarrays as the process of selecting discriminative genes connected with classification from gene expression data, training a classifier, and then classifying fresh data using the learnt classifier. Once we have the data on gene expression that was estimated from the DNA microarray, our prediction engine will proceed through the following three stages: The data are first preprocessed by the system. After that, we split the model into a training and testing set and evaluated seven different machine learning

methods to see which one provided the most accurate results. When we have obtained the best model, we then store it away and construct a prediction function so that we can use it in the future.

6. Implementation

6.1 Module 1: Data Pre-processing

AUTHORS	METHODOLOGY	CONS
] N. Patel, K. Passi and C. K. Jain	Methylation Density, NMF, SONMF, PCA.	There are some drawbacks to the NMF approach for dimension reduction, such as the zero-locking issue.
S. Lobo and M. S. Pallavi	SVM and KNN classifier.	In this protein causing cancer were identified not the cancer type.
Bogdan belean, robert gutt1, carmen costea, and ovidiu balacescu	Preprocessing, Grid Alignment, Segmentation and Normalization.	This is a time-consuming process.
S. Wichaidit, P. Wardkean, K. Chaiwong and W. Wettayaprasit	FFT Feature, Symmetry of Methylation Density, NMF.	This algorithm does not have any conditions about the number of classes or type of input dataset.

6.1.1 Removing Unwanted Columns

Our dataset contains redundant data. We consider it a good idea to eliminate these features from the dataset because their presence has no impact on the target. To ensure that we only work with relevant data, this method of deleting unnecessary features and maintaining only the required features in the dataset is quite helpful.

Therefore, we drop a few columns from our dataset using the drop command. Python's "df.drop" function can be used to remove a single column or a number of columns from a panda’s data frame. We further define the title of the column, index, axis and labels.

6.1.2 Identifying Null Values

There are several missing pieces in the Kaggle dataset that we have available. This can be because such data was not adequately gathered or because there is no such data. These entries are represented in the dataset by none. Despite the origins, it makes our computing more difficult and distorted. So, we locate these missing data and swap them out for relevant components.

The handling of None and NaN for missing or null values is the same in Pandas. To make this practice simpler, there are numerous utilities for discovering, discarding, null values in the pandas Data Frame. To check for missing values, Pandas Data Frame employs the functions isnull () and not null (). These algorithms can be applied to pandas Series to identify null value inside a series.

6.2 Module 2: Model Making and Training

To assess how well ML algorithms work with prediction-based methods and applications, the train-test split is utilized. This is a quick and easy method for comparing our own ML model's output to that of other machines. By definition, 30% of the data in the Test set are actual, compared to 70% of the unprocessed data in the Training set.

We must separate a dataset into a train set and a test set in order to evaluate how well our ML model works. Recognizable and functional train set figurines are used in the model. The test data set, which is the second batch, only uses forecasts.

We employ the subsequent strategy to split our data into train and test. The packages for pandas and sklearn are used. The most practical and reliable machine learning library for Python is Sklearn. The scikit-learn library's model selection module includes the splitter function train test split (). The CSV file is subsequently imported using the read csv () function. The variable

df now contains the data frame.

Once this is done, 30% of the data is for testing and 70% is for training. Also, we set random state to 0 to provide a random distribution of data between these two datasets.

6.2.1 Training the Models

We have used the following models for our analysis:

6.2.1 a) Logistic Regression

Our model's methods combine logistic regression and binary classification. The procedure of building the classifier employs logistic regression. Logistic regression is more complex than linear regression. This is because data that are widely scattered in one area cannot be classified using linear regression. When using linear regression to classify the data, the line might divide the input data into two main groups (or classes). When the data overlap, the line cannot clearly distinguish between the two classes. This limitation is overcome through logistic regression.

The LR algorithm is used to predict the binary values for the set of independent variables. When a variable outcome is categorical, the probability log is used as the dependent variable to fit a logistic function to the data and forecast the likelihood that an event will occur. LR is a use of linear regression, as you can see.

O = e^(I0 + I1*x) / (1 + e^(I0 + I1*x)) (1)

Where, O is the predicted output, I0 is intercept term, and I1 is the coefficient for the single input value (x). First, we import NumPy. Then, we'll generate a logistic regression model from the sklearn package using the Logistic Regression () function. Fit (), a function on this object, fills the regression object with information about the connection between the independent and dependent variables as parameters:

logr=linearmodel.LogisticRegression() (2)

logr.fit (X, y) (3)

6.2.1 b) Gaussian NB

The Gaussian Naive Bayes approach is a variant of the Naive Bayes strategy that use continuous data and conforms to the Gaussian normal distribution. When working with continuous data, it is conventional practise to assume that each class's continuous values are distributed according to a normal distribution. This is carried out to facilitate working with continuous data. On the possibility of the attributes, we'll proceed with the following assumption:

In the Gaussian Naive Bayes method, continuous valued features and models are thought to individually correspond to a Gaussian distribution. A fundamental model can be created by making the supposition that the data are distributed in a Gaussian fashion with no covariance between the dimensions. Here is one strategy for going about the task. Calculating the mean and SD of the points within every label is all that is necessary to create this distribution, which can then be used to fit this model.

6.2.1 c) SGD Classifier

The parameters of a function that produce the lowest cost function are found using the Stochastic Gradient Descent (SGD) Classifier optimization method. The method shares many characteristics with traditional gradient descent. But it only calculates derivative of the loss for one of the data points, not for all of them (hence the name, stochastic). The algorithm is therefore substantially faster than Gradient Descent. The SGD Classifier model is fitted with the trained data using the SGDClassifier () class from scikit-learn after the data has been separated into dependent and independent variables. By

utilizing the model to make predictions from the data, the model is ultimately validated.

6.2.1 d) Gradient Boost

Gradient Boosting is the name of one of the most popular categories of boosting algorithms. Each predictor in gradient boosting is in charge of

fixing the mistakes produced by its forerunner. As opposed to Adaboost, each predictor is trained using the labels from the preceding residual errors instead of changing the weights of the training instance. Gradient Boosted Trees is a technique, and CART is the learner that it uses most frequently. The following diagram shows how to train gradient-boosted trees to address regression problems.

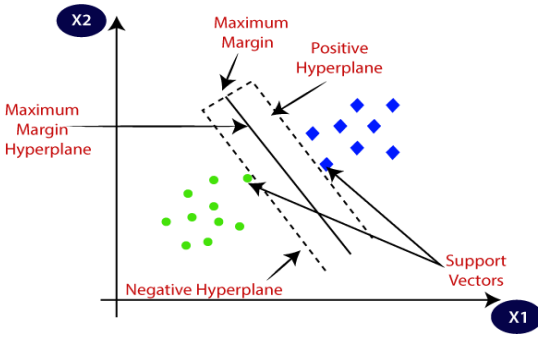
The feature matrix X and labels Y are used to train Tree1. The Residual error from the training set, designated by the letter r1, are calculated based on the predictions given by the letter y1 (hat). The labels from the feature matrix X and Tree1 residual error r1 are then used to train Tree2's neural network. The residual, denoted by r2, is then calculated using the anticipated outcomes, denoted by r1 (hat). To train each of the ensemble's N trees, this process is performed as often as necessary.

6.2.1 e) AdaBoost

AdaBoost was the first boosting algorithm devised for the purpose of binary classification that was actually successful after its initial implementation. The term "Adaptive Boosting," which is abbreviated as "AdaBoost," is the name of a very popular strategy for "boosting," in which numerous "weak classifiers" are combined into a single "strong classifier." First, the dataset needs to be initialized, and then each data point needs to be assigned the same weight. After that, we give this information as input to the model and look for data points that have been incorrectly categorized. Finally, you should give more weight to the data points that were incorrectly categorized.

6.2.1 f) SVC

In order to categorize an n-dimensional space, the Support Vector Machine (SVM) technique attempts to create the optimum decision boundary or line. This will make it simple for us to categorize any new data points in the future. This decision's optimal border is known as a hyperplane. SVM is used to choose the extreme points and vectors that will be used to create the hyperplane. The Support Vector Machine is a piece of technology that was created as a result of these peculiar situations, or support vectors. See how two distinct groups can be divided from one another using a decision boundary or a hyperplane in the illustration below:



6.2.1 g) Ridge Classifier

A model tuning method called ridge regression is used to examine any multicollinear data. For this kind of investigation, ridge regression is necessary. This is how the L2 regularization procedure is done. The projected values really turn out to be substantially different from the actual values when the problem of multicollinearity emerges, least-squares techniques provide unbiased findings, and variances are significant. Here is a cost function for ridge regression:

$$\boldsymbol{Min}(||\boldsymbol{Y} - \boldsymbol{X}(\boldsymbol{theta})|| + \boldsymbol{\lambda}||\boldsymbol{theta}||\text{^}\boldsymbol{2} \quad (4)$$

The technical term for the punishment is lambda. The ridge function refers to the value given here as an "alpha parameter." As a result, by changing the values of alpha, we have control over the penalty term. The size of the penalty rises as alpha values rise, which in turn causes the magnitude of the coefficients to drop.

Once the dataset has been divided into training and test datasets, we apply a machine learning technique to assess how well the training dataset performed. In this study, scikit was used to fit the data. The most helpful Python machine learning library is undoubtedly scikit-learn. Clustering, classification, regression, and dimensionality reduction are just a few of the helpful tools for machine learning and statistical modelling that are included in the sklearn library.

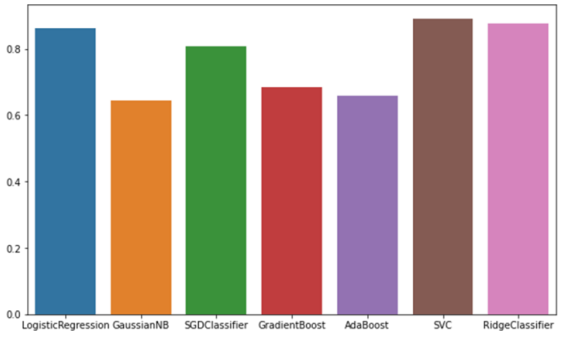
The scikit learn "fit" approach is one of these tools. The "fit" approach uses the training data to train the algorithm after the model has been initialized. Really, all it does is that. After that, we may carry on with the machine learning process using other scikit learn techniques like predict and score.

We apply the fit algorithm in this way to the training dataset. We utilize the fit technique as shown in the figure below after first defining the model instance. Lastly, we give the features and label vector for the training dataset in the parenthesis. The X train and Y train datasets are also known by those names. The fit() method's X-input, X train, is a 2-dimensional format for the model's ideal fitting. If not, our model displays a mistake.

We will train several models one at a time and evaluate the accuracy of each. Accuracy is the percentage of data that is correctly categorized on a scale from 0 to 1. We analyze the projected class to the original class quite instinctively in this measure because we want the model to categorize the data. We accomplish this by using the Sklearn package's accuracy score function. We use the command accuracy score (y test, y pred)) to produce for the same.

6.2.2 *Saving Best Model*

Let's save the model right now. The joblib.dump method is what we're using. The method's first argument has no fixed value because it depends on the model, hence there isn't one. The function's output file's name and location on storage are specified in the second argument. To reload the model, we employ the joblib.load method. It accepts the file name and directory path as parameters. To test whether it actually works, we compute some predictions after that. In every way, they should be a perfect reproduction of the rf model. We can conclude that our model was effectively retained as we got the desired result.



6.3 *Module 3: Model Prediction*

The last stage is to develop a prediction function that will accept input as location and time and determine the crimes that can take place in the given area at the given time. Using the Python predict() function, we can determine the labels of the data values based on the training model. The test data is the only input that the predict() method typically gets. The predict() method uses the learnt label to map & predict the labels for test data on top of the training model. The dataset is initially

loaded into the environment. The dataset can be loaded from the system using the pandas.read csv() function. Further, we predict the accuracy of our model using the command accuracy_score imported from metrics metrics_accuracy_score. In our model, the created prediction function takes user input as text and convert it into array and run in-built model prediction function to predict the type of cancer the person suffering.

From the above table and graph we infer that support vector classifier produces the best model. So, we save this model for predicting the cancer

8. Conclusion And Future Work

DNA microarrays are an effective method for analyzing large amounts of data in order to categorize gene expression and forecast the likelihood of cancer illness for certain cancer types. In order to properly identify and classify distinct forms of cancerous tissue and disease, microarray data formats are required. Because just a few genes can accurately and completely explain a disease on a biological level, understanding the genes that cause disease is challenging and time-consuming. The goal of feature selection in machine learning is to attain the highest possible recognition and classification accuracy while acquiring the smallest possible subset of problem space features. The availability of high-powered computers has allowed the deep learning approach to flourish in the medical area. It has been shown that DNA microarrays used for cancer prediction and classification can be enhanced by employing deep learning techniques.

7. Experimental Results

By using the above-mentioned algorithms, we see which algorithm produces the best model based on the accuracy score we get for each method. Given below are the accuracy score produced by each algorithm.

Algorithm	Accuracy Score
Logistic Regression	86.30%
Gaussian NB	64.38%
SGD Classifier	80.82%
Gradient Boosting	68.49%
AdaBoost Classifier	65.75%
Support Vector Classifier	89.04%
Ridge Classifier	87.67%

In, future we will discover more methods to find cancer and what are the Symptoms and treatment they need to take will be prescribed. Many techniques will be improved and more technology will be developed in this field.

REFERENCES

[1] Chandrasekar, V., V. Suresh Kumar, T. Satish Kumar, and S. Shanmugapriya. "Disease prediction based on micro array classification using deep learning techniques." Microprocessors and Microsystems 77 (2020): 103189.

[2] Kourou, Konstantina, George Rigas, Costas Papaloukas, Michalis Mitsis, and Dimitrios I. Fotiadis. "Cancer classification from time series microarray data through regulatory dynamic bayesian networks." Computers in Biology and Medicine 116 (2020): 103577.

[3] Belean, Bogdan, Robert Gutt, Carmen Costea, and Ovidiu Balacescu. "Microarray image analysis: from image processing methods to gene expression levels estimation." IEEE Access 8 (2020): 159196-159205.

[4] Hambali, Moshood A., Tinuke O. Oladele, and Kayode S. Adewole. "Microarray cancer feature selection: review, challenges and research directions." International Journal of Cognitive Computing in Engineering 1(2020): 78-97.

[5] Fathi, Hanaa, Hussain AlSalman, Abdu Gumaei, Ibrahim IM Manhrawy, Abdelazim G. Hussien, and Passent El-Kafrawy. "An efficient cancer classification model using microarray and high-dimensional data." Computational Intelligence and Neuroscience 2021 (2021)

[6] Zerouaoui, Hasnae, Ali Idri, and Khalid El Asnaoui. "Machine learning and image processing for breast cancer: a systematic map." In World Conference on Information Systems and Technologies, pp. 44-53. Springer, Cham, 2020.

[7] Nageswaran, Sharmila, G. Arunkumar, Anil Kumar Bisht, Shivlal Mewada, J. N. V. R. Kumar, Malik Jawarneh, and Evans Asenso. "Lung Cancer Classification and Prediction Using Machine Learning and Image Processing." BioMed Research International 2022 (2022).

[8] Wang, Lina. "Predicting Colorectal Cancer Using Residual Deep Learning with Nursing Care." Contrast Media & Molecular Imaging 2022 (2022).

[9] Hasan, Mahamudul, Surajit Das Barman, Samia Islam, and Ahmed Wasif Reza. "Skin cancer detection using convolutional neural network." In Proceedings of the 2019 5th international conference on computing and artificial intelligence, pp. 254-258. 2019.

[10] Li, Yang, Andrei Păun, and Mihaela Păun. "Improvements on contours based segmentation for DNA microarray image processing." Theoretical Computer Science 701 (2017): 174-189.Type equation here.

[11] Wang, Yu, Marc Q. Ma, Kai Zhang, and Frank Y. Shih. "A hierarchical refinement algorithm for fully automatic gridding in spotted DNA microarray image processing." Information Sciences 177, no. 4 (2007):1123-1135.

[12] Osama, Sarah, Hassan Shaban, and Abdelmgeid A. Ali. "Gene reduction and machine learning algorithms for cancer classification based on microarray gene expression data: A comprehensive review." Expert Systems with Applications (2022): 118946.

[13] Nosrati, Vahid, and Mohsen Rahmani. "An ensemble framework for microarray data classification based on feature subspace partitioning."Computers in Biology and Medicine 148 (2022): 105820.

[14] Sattar, Mohsin, Abdul Majid, Nabeela Kausar, Muhammad Bilal, and Muhammad Kashif. "Lung cancer prediction using multi-gene genetic programming by selecting automatic features from amino acid sequences." Computational Biology and Chemistry 98 (2022): 107638.

[15] Yang, Yang, Li Xu, Liangdong Sun, Peng Zhang, and Suzanne S. Farid. "Machine learning application in personalised lung cancer recurrence and survivability prediction." Computational and Structural Biotechnology Journal 20 (2022): 1811-1820.

[16] Gupta, Siddharth Raj. "Prediction time of breast cancer tumour recurrence using Machine Learning." Cancer Treatment and Research Communications 32 (2022): 100602.

[17] Liang, Xueheng, Xingyan Yu, and Tianhu Gao. "Machine learning with magnetic resonance imaging for prediction of response to neoadjuvant chemotherapy in breast cancer: A systematic review and meta-analysis." European Journal of Radiology (2022): 110247.

[18] Takamatsu, Manabu, Noriko Yamamoto, Hiroshi Kawachi, Akiko Chino, Shoichi Saito, Masashi Ueno, Yuichi Ishikawa, Yutaka Takazawa, and Kengo Takeuchi. "Prediction of

early colorectal cancer metastasis by machine learning using digital slide images." Computer methods and programs in biomedicine 178 (2019): 155-161.

[19] Peterson, Leif E., and Matthew A. Coleman. "Machine learning-based receiver operating characteristic (ROC) curves for crisp and fuzzy classification of DNA microarrays in cancer research." International Journal of Approximate Reasoning 47, no. 1 (2008): 17-36.

[20] Han, Xiao Hong, Deng Ao Li, and Li Wang. "A hybrid cancer classification model based recursive binary gravitational search algorithm in microarray data." Procedia Computer Science 154 (2019): 274-282.

[21] Nazari, Elham, Mehran Aghemiri, Amir Avan, Amin Mehrabian, and Hamed Tabesh. "Machine learning approaches for classification of colorectal cancer with and without feature selection method on microarray data." Gene Reports 25 (2021): 101419.

[22] Rostami, Mehrdad, Saman Forouzandeh, Kamal Berahmand, Mina Soltani, Meisam Shahsavari, and Mourad Oussalah. "Gene selection for microarray data classification via multi-objective graph theoretic-based method." Artificial Intelligence in Medicine 123 (2022): 102228.

[23] Sayed, Sabah, Mohammad Nassef, Amr Badr, and Ibrahim Farag. "A nested genetic algorithm for feature selection in high-dimensional cancer microarray datasets." Expert Systems with Applications 121 (2019): 233-243.

[24] Garro, Beatriz A., Katya Rodríguez, and Roberto A. Vázquez. "Classification of DNA microarrays using artificial neural networks and ABC algorithm." Applied Soft Computing 38 (2016): 548-560.