Black Sea Journal of Engineering and Science
doi: 10.34248/bsengineering.1492652

Open Access Journal
e-ISSN: 2619 – 8991

Research Article
Volume 7 - Issue 4: 693-704 / July 2024

EFFECTIVE CANCER DIAGNOSIS THROUGH HIGH-DIMENSIONAL MICROARRAY DATA ANALYSIS BY INTEGRATING DCT AND UFS

Enes EFE[1]*

[1]*Hitit University, Department of Electrical and Electronics Engineering, 19169, Çorum, Türkiye*

Abstract: Cancer remains a global health challenge, with various types such as lung, breast, and colon cancer posing significant threats. Timely and accurate diagnosis is crucial for effective treatment and improved survival rates. Genetic research offers promising avenues in the fight against cancer, as identifying gene mutations and expression levels enables the development of targeted therapies and a deeper understanding of disease subtypes and progression. This study investigates a novel hybrid method aimed at improving the accuracy and efficiency of cancer diagnosis and classification. By combining Discrete Cosine Transformation (DCT) and Univariate Feature Selection (UFS) methods, the feature selection process is optimized for the dataset. The extracted features are then rigorously tested using established classifiers to assess their effectiveness in cancer classification. The proposed method's performance was evaluated using eight distinct datasets, and metrics such as MF1, K-score, and sensitivity were calculated and compared with various methods in the literature. Empirical evidence demonstrates that the proposed method outperforms others on 5 out of 8 datasets in terms of both accuracy and computational efficiency. The presented method represents a reliable tool for cancer diagnosis and classification.

Keywords: Cancer, Microarray data, Discrete cosine transform, Univariate feature selection, Genomic data analysis

***Corresponding author:** Hitit University, Department of Electrical and Electronics Engineering, 19169, Çorum, Türkiye
E mail: enesefe@hitit.edu.tr (E. EFE)
Enes EFE https://orcid.org/0000-0002-6136-6140

Received: May 30, 2024
Accepted: July 01, 2024
Published: July 15, 2024

Cite as: Efe E. 2024. Effective cancer diagnosis through high-dimensional microarray data analysis by integrating DCT and UFS. BSJ Eng Sci, 7(4): 693-704.

1. Introduction

Microarray data refers to high-dimensional datasets that enable the simultaneous examination of genetic changes in gene expression across thousands of cells. This extensive dimensionality facilitates comprehensive and detailed analyses in cancer research, leading to a deeper understanding of the molecular basis of cancer. However, analyzing and interpreting large datasets poses challenges that require specialized data mining and statistical techniques to achieve accurate results. As a result, cancer researchers and healthcare professionals face challenges when processing and interpreting microarray data. Nevertheless, the broad perspective provided by this technology has resulted in significant advancements in cancer diagnosis, treatment, and disease comprehension (Golub et al., 1999).

Cancer research encounters challenges due to the high dimensionality of datasets, incorporating thousands of genes, which complicates genomic data analyses (e.g., microarrays, RNA sequencing). The complexity of the analysis processes makes them challenging and hinders the achievement of accurate and clear results. Another issue is the lack of specific genes acting as classifiers for particular cancer types, diminishing the classification power of datasets (Kilicarslan et al., 2020). Researchers need to create specific algorithms and analysis techniques to identify and detect unique genes associated with different types of cancer. Furthermore, the higher interactivity among genes compared to other datasets complicates the understanding of cancer mechanisms. Some genes can activate others or trigger gene expressions, contributing to uncontrolled cell growth and metastatic processes. Thus, a comprehensive understanding and examination of gene interactions are crucial for developing targeted and effective strategies in cancer treatment.

Integrating data analysis methods is crucial for successful cancer research. Given the complexity and high dimensionality of datasets in cancer research, traditional methods may prove inadequate. Hence, using advanced statistical methods, machine learning algorithms, and network analysis techniques together is crucial for making significant advancements in cancer treatment (Orhan and Yavşan, 2023). Cancer research primarily aims to comprehend the fundamental mechanisms of cancer and define cancer types by analyzing genomic data. While microarray datasets encompass gene expression profiles, their high dimensionality and noise often render traditional feature extraction and dimensionality reduction methods insufficient. Studies reveal that existing feature extraction or dimensionality reduction methods may not be effective for all microarray datasets, necessitating the development of new approaches for feature extraction and dimensionality reduction. Hence, further work is required in this field to establish effective and personalized cancer diagnosis and treatment approaches (Li et al., 2005).

Numerous studies in the literature have aimed to classify datasets containing gene expression levels and select appropriate features. These studies utilize various methods, such as filter, wrapper, embedded, and hybrid methods for feature selection. Filter methods employ statistical criteria (e.g., Pearson Correlation, Mutual Information, Information Gain (IG)) to reduce the number of genes, and some researchers (Gao et al., 2017) have developed original algorithms within this context. Relief-F and IG are examples of filter methods that exhibit improved performance with an increasing number of genes. Wrapper methods, on the other hand, combine classification methods like Genetic Algorithm (GA), Support Vector Machine (SVM), and k-nearest Neighbors (kNN) (Gunavathi and Premalatha, 2014; Kar et al., 2015). Embedded methods involve classifiers to select features, and preferred techniques include SVM-Recursive Feature Elimination (RFE), First Order Inductive Learner (FOIL) based FRFS, and penalized DVM improved with T-test (Guyon et al., 2002; Maldonado et al., 2011). Additionally, some studies (Luo et al., 2019; Othman et al., 2020; Meenachi and Ramakrishnan, 2021; Qaraad et al., 2021) have integrated hybrid methods, combining filter, wrapper, and embedded methods. For instance, MRMR and SVM-RFE have been combined (Mundra and Rajapakse, 2009), the Relief-F filter method applied as preprocessing, and classification performed using ELM, with the SVM-RFE method enhanced with F-test (Luo et al., 2019).

After examining the literature, it is clear that the classification methods used are often complex and do not perform well. Therefore, this study proposes the integration of Discrete Cosine Transformation (DCT) and Univariate Feature Selection (UFS). DCT can represent gene expression data in a low-dimensional space, reducing noise and emphasizing relationships between features (Er et al., 2005). Furthermore, several studies in various domains demonstrate that DCT positively impacts performance and enhances model stability (Efe and Özşen, 2022; Efe and Ozsen, 2023). The Univariate Feature Selection method evaluates the performance of each feature separately through classifiers, identifying the most significant features (Efe and Yavsan, 2024). By combining these two methods, the objective is to reduce the complexity of gene expression data, leading to more meaningful and effective feature selection. Consequently, this combination can contribute to obtaining less noisy and more explanatory features for the classification algorithm, reducing the risk of overfitting and enhancing the model's generalization capability. As such, the integration of Discrete Cosine Transformation and Univariate Feature Selection aims to optimize the feature selection process in the classification of gene expression levels, resulting in more reliable results.

The proposed method was tested using four primary classifiers: Neural Network (NN), Support Vector Machine (SVM), k-nearest Neighbors (kNN), and Convolutional Neural Network (CNN). The results were compared with other studies in the literature. The obtained results demonstrate superior performance compared to the findings in the literature, indicating a promising approach for future studies.

The main contributions of this research are threefold:

- High-dimensional microarray datasets utilized in cancer research are transformed into a lower-dimensional space through Discrete Cosine Transform (DCT), facilitating analysis and reducing noise while accentuating relationships between features.
- Univariate Feature Selection (UFS) evaluates the impact of each feature on the classifier individually, identifying the most significant attributes to enhance the performance of the classification algorithm.
- The integration of DCT and UFS optimizes the feature selection process in cancer research, refining meaningful features obtained through DCT to identify the most salient attributes, ultimately leading to improved accuracy and efficiency of the classification algorithm.

2. Materials and Methods

2.1. Dataset and Data Preparation

In this study, we conducted experimental investigations to explore the classification success using eight distinct gene microarray datasets, which are among the most commonly used datasets in the literature. These datasets were carefully selected to represent typical scenarios encountered in cancer research and classification tasks. As presented in Table 1, each of these microarray datasets was collected from various sources within the biomedical field and utilized for the classification of patients with cancer. Given the substantial number of features and the limited number of samples in these microarray datasets, dimension reduction was considered necessary during the training stage. As such, we applied dimension-reduction techniques to address this challenge effectively. The datasets listed in Table 1 have been extensively employed in various research studies, making them highly relevant for assessing the effectiveness of the methodologies employed in our study.

The leukemia dataset (Golub et al., 1999) comprises 72 bone marrow and peripheral blood samples obtained from individuals diagnosed with leukemia, with the specific goal of distinguishing between two cancer subtypes: Acute Myeloid Leukemia (AML) and Acute Lymphoblastic Leukemia (ALL). Among these samples, 25 were identified as AML, while 47 were categorized as

ALL. By employing high-density oligonucleotide microarrays, the study investigated the gene expression patterns in these two cancer subtypes, analyzing a total of 7129 genes.

Table 1. Number of genes, instances, and classes for each experimental dataset

Datasets	Gene	Instance	Class
LEUKEMİA	7129	72	2
Colon	2000	62	2
Prostate	12600	136	2
Ovarian	15154	253	2
Lymphoma (DLBCL)	7129	77	2
Breast Cancer	47293	128	2
Breast Cancer - 2	24481	97	2
CNS	3495	1209	2

In their pioneering work (Alon et al., 1999), established the colon cancer dataset through the application of oligonucleotide microarrays, which facilitated the analysis of over 6500 genes in 40 tumor samples and 22 normal colon tissue samples. With a specific focus on colon cancer, the researchers refined the dataset to encompass a high-density set of 2000 genes.

The prostate cancer dataset (Singh et al., 2002) comprises 136 samples, with 59 being normal tissues and 77 being tumor tissues. It aims to explore gene expression patterns associated with prostate cancer for potential biomarker discovery and improved understanding of the disease.

The ovarian cancer dataset (Petricoin et al., 2002) comprises 253 samples, with 91 being normal tissues and 162 tumor tissues, featuring gene expression levels from 15,154 genes. It offers valuable data for exploring gene expression patterns in ovarian cancer, aiding in potential biomarker discovery and a better understanding of the disease.

The DLBCL lymphoma dataset (Shipp et al., 2002) contains 77 samples and was constructed specifically for differentiating between common diffuse large B-cell lymphoma (DLBCL) and follicular lymphoma. Each sample in the dataset is characterized by the expression levels of 7129 genes. Researchers can utilize this dataset to investigate gene expression patterns and develop classification models to accurately distinguish between these two lymphoma subtypes.

The breast cancer (BRC) dataset (Naderi et al., 2007) consists of 128 samples and is specifically utilized for differentiating between luminal and non-luminal breast cancer subtypes. Each sample in the dataset is characterized by the expression levels of 47293 genes. Researchers can leverage this dataset to explore gene expression patterns and develop robust models for accurate classification of luminal and non-luminal breast cancer types.

The breast cancer-2 (BRC-2) dataset (Van't Veer et al., 2002) includes two subsets used to differentiate between metastasis occurrence and non-occurrence within the first five years following breast cancer diagnosis. These subsets provide valuable information for identifying potential predictive factors related to early-stage breast cancer patients' metastatic outcomes.

The CNS dataset (Pomeroy et al., 2002) is specifically designed to explore the differentiation between survival and mortality outcomes in patients with central nervous system (CNS) cancer. With a total of 60 samples, each dataset entry includes the expression levels of 7129 genes. This dataset holds significant potential for investigating gene expression patterns associated with patient prognosis in CNS cancer, potentially leading to advancements in personalized treatment approaches.

2.2. Method

2.2.1. The discrete cosine transform

The Discrete Cosine Transform (DCT) is a mathematical transformation technique used in analyzing temporal or spatial data, such as gene expression levels. It proves to be a powerful tool for feature extraction in data analysis by converting the data into its frequency components. By applying this transformation, the data is broken down into fundamental components, thereby unveiling essential features and representing the signal's energy content in the frequency domain. Employing the DCT on gene expression levels or other temporal/spatial data allows for the exploration of valuable information, leading to advancements in data compression, pattern recognition, and data processing applications.

The DCT equation, which transforms the input data *f(x)* into its frequency domain representation *Y(u)*, is given in Equation 1:

$$Y(u) = \sqrt{\frac{2}{N}}a(u)\sum_{x=0}^{N-1} f(x).\cos(\frac{\pi.(2x+1).u}{2N}),$$
$$a(u) = \begin{cases} \frac{1}{\sqrt{2}}, u = 0 \\ 1, u > 0 \end{cases} \tag{1}$$

where N represents the total number of samples in the dataset, $Y(u)$ denotes the DCT result for the particular frequency component u, and $f(x)$ represents an element of the input data at index x. The $a(u)$ coefficient takes the value $\frac{1}{\sqrt{2}}$ when $u = 0$ and 1 for $u > 0$, reflecting the symmetry properties of the DCT. This equation allows the extraction of frequency components from the input data, enabling valuable information discovery and contributing to data compression, pattern recognition, and data processing applications.

The advantage of combining feature extraction techniques in both the time domain (original data without DCT) and the frequency domain (DCT-applied data) lies in capturing complementary information from the data. Considering the benefits of feature extraction in the frequency domain, it can be observed that:

I. Frequency Domain Information: The Discrete Cosine Transform (DCT) is primarily used for signal processing tasks and is known for its ability to represent signals in the frequency domain. By applying DCT to the microarray dataset, the model can capture frequency-related patterns and variations in the data. This enables the model to detect periodic or repetitive patterns present in the microarray data, which may be challenging to identify in the time domain alone.

II. Noise Reduction: Transforming the data into the frequency domain through DCT can help reduce the impact of noise or irrelevant features existing in the time domain. Noise components typically manifest in high-frequency regions, and DCT tends to concentrate the signal energy in a smaller number of coefficients, effectively reducing the noise-related components.

III. Dimension Reduction: DCT, when applied for feature extraction in the frequency domain, facilitates dimension reduction, resulting in a more concise representation of the dataset. Dimensionality reduction helps mitigate the curse of dimensionality, improves computational efficiency, and potentially prevents overfitting in subsequent stages of the model.

Regarding gene expression level analysis, the usage of DCT offers several advantages:

I. Identification of Relevant Genes: DCT transformation of gene expression levels reveals frequency-related patterns and variations. This transformation can emphasize genes crucial for the classification process, which might not be easily detectable in the original gene expression data.

II. Detection of Gene Interactions: Genes exhibit higher levels of interaction compared to other data sets. For instance, certain genes can activate or trigger the expression of other genes. By applying DCT to gene expression data, such interactions can be highlighted, leading to a better understanding of gene regulatory networks.

By contributing to the resolution of these issues, the use of DCT in gene expression data can enhance the performance and interpretability of classification and diagnostic procedures.

2.2.2. The univariate feature selection

Univariate Feature Selection (UFS) is a feature selection method used to improve the classification or regression performance of features within a dataset. In this technique, each feature's contribution to the independent classification performance is evaluated. The impact of each feature on classification or regression is measured, enabling the selection of the most significant features and elimination of the least relevant ones.

Univariate Feature Selection is an effective approach to reduce dataset high-dimensionality and eliminate unnecessary features. This results in decreased irrelevant information noise, reduced model complexity, and improved performance of classification/regression algorithms. Common statistical metrics such as Pearson Correlation, Anova F-test, Mutual Information, and Chi-square test are commonly used in Univariate Feature Selection. Widely applied in fields such as data analytics, machine learning, and model development, Univariate Feature Selection is a valuable tool, particularly in managing high-dimensional datasets and enhancing model performance.

2.2.3. Support vector machine

Support Vector Machine (SVM) is a powerful algorithm widely employed in machine learning for tasks such as classification, regression, and data separation. Its primary objective is to effectively segregate data points into specific classes using a hyperplane. SVM is known for its effectiveness in handling both low-dimensional and high-dimensional datasets. It achieves optimal separation between two classes by identifying support vectors from the training data and maximizing the margin between these vectors. To represent data points in higher-dimensional spaces, SVM utilizes various kernel functions. These kernel functions facilitate the transformation of data, enabling the creation of more intricate decision boundaries. Some common kernels used in SVM include:

- Linear Kernel: The basic kernel used for linearly separable datasets. It separates data points with a linear hyperplane in higher-dimensional space.
- Polynomial Kernel: This kernel handles nonlinear separations by transforming data into higher-dimensional spaces using polynomials. The degree of the polynomial controls the complexity of the kernel.
- Radial Basis Function (RBF) Kernel: A popular kernel that transforms data into infinite-dimensional spaces to address nonlinear classification problems. RBF is frequently preferred in SVM and delivers good results across various problems.
- Sigmoid Kernel: This kernel employs a hyperbolic tangent function similar to the activation function used in neural networks. It transforms data into higher-dimensional spaces.

SVM is a versatile and powerful classification algorithm that employs different kernel functions to map data into higher-dimensional spaces and achieve linear separation of classes in that space. However, selecting the appropriate kernel function and tuning the model's hyperparameters are crucial factors that significantly impact SVM's performance.

2.2.4. K-nearest neighbors

K-Nearest Neighbors (KNN) is a fundamental algorithm utilized in machine learning and statistical classification. KNN performs classification or value estimation based on the nearest neighbors surrounding a data point. The underlying principle of the KNN algorithm is straightforward. To classify or evaluate a given sample, KNN calculates the distances between the sample and all other examples in the dataset. Subsequently, it identifies the K closest neighbors and uses their labels or values to make predictions. KNN is particularly renowned for classification problems, though it can also be applied to regression problems. In classification tasks, the labels of examples are categorical (e.g., "red" or "blue"), while in regression tasks, the values of examples are continuous numbers (e.g., the price of a house). One of the strengths of KNN lies in its simplicity during the training process and its adaptability to new data. Furthermore, it does not make any specific assumptions about the structure or size of the training data, rendering it suitable for various data types. However, when dealing with large datasets, the computational load may increase, necessitating careful data preprocessing.

The primary parameter of the KNN algorithm is K, which represents the number of nearest neighbors. The selection of an appropriate K value significantly influences the model's accuracy. Smaller K values can make the model sensitive to data noise, while larger K values may result in smoother classification boundaries. In conclusion, the K-Nearest Neighbors algorithm is favored for its simplicity, interpretability, and versatility in handling different data types for classification and regression tasks. Nonetheless, careful consideration of the K value and thoughtful data preprocessing are vital factors, especially when dealing with sizable datasets to achieve optimal performance. The KNN algorithm is a fundamental technique used in machine learning and statistical classification. KNN performs either classification or value estimation by considering the closest neighbors surrounding a data point. The mathematical formulation of the KNN algorithm is as follows:

For Classification using KNN:

Let the dataset be denoted as D = {(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)}, where x_i represents the features of the examples, and y_i represents their corresponding class labels. Assume we have a new example that we want to classify, and we denote it as x'.

Step 1: If the size of the dataset is smaller than K, set K equal to the dataset size. Otherwise, utilize a distance metric (e.g., Euclidean distance) to select K nearest examples to x'.

Step 2: Obtain the class labels of these K neighbors.

Step 3: For classifying x', use the most frequently occurring class label among the K neighbors. This label will be the final classification result.

2.2.5. Neural network and convolutional neural networks

A Neural Network is a type of machine-learning model inspired by the structure and functioning of the human brain. It consists of interconnected nodes, called neurons, organized in layers. The three main types of layers in a typical neural network are:

I. Input Layer: It receives raw data or features as input and passes them to the subsequent layers for processing.

II. Hidden Layers: These layers process the input data using a combination of weights and activation functions. The number of hidden layers and neurons in each layer can vary depending on the complexity of the problem.

III. Output Layer: The final layer produces the predictions or output of the model, which can be a single value or a set of values, depending on the type of problem (classification or regression).

The information flow between layers is determined by weights (parameters) associated with the connections between neurons. The neural network learns from training data by adjusting these weights to minimize the difference between predicted outputs and actual outputs, using techniques like backpropagation and optimization algorithms. The general equation for a single neuron in a neural network can be written as given in Equation 2:

$$z = \sum(input * weight) + bias \qquad (2)$$

where z is the weighted sum of inputs, bias is a constant term added to the weighted sum, and the activation function is a non-linear function that introduces non-linearity into the model.

A Convolutional Neural Network (CNN) is a specialized type of neural network designed for image and visual data processing. CNNs use a unique layer called the convolutional layer, which applies filters (also called kernels) to input images to detect features like edges, textures, and patterns. CNNs are particularly effective for tasks such as image classification, object detection, and image segmentation. The main components of a CNN are:

I. Convolutional Layer: This layer applies convolutional filters to the input image, generating feature maps that highlight specific patterns in the image.

II. Activation Function: After the convolutional operation, an activation function (often ReLU - Rectified Linear Unit) is applied element-wise to introduce non-linearity.

III. Pooling Layer: Pooling layers reduce the spatial dimensions of feature maps, helping to make the model more computationally efficient and robust to variations in the input.

IV. Fully Connected Layers: After several convolutional and pooling layers, the extracted features are passed to fully connected layers to make the final predictions.

The equations for the convolution operation and Leaky ReLU activation function are as given in Equation 3:

$$output[i,j] = \sum\sum(input[x,y] * kernel[i,j]) \qquad (3)$$

Leaky ReLU Activation function (Equations 4):

$$Leaky\ ReLU(x) = \max(\propto x, x) \qquad (4)$$

In Leaky ReLU, ∝ represents a hyperparameter typically set to a small positive value (e.g., 0.01). When the input, x, is positive, Leaky ReLU behaves like the regular ReLU, returning the input value, x. However, if x is negative, Leaky ReLU returns $\propto x$, introducing a small positive slope in the negative range, which allows for activations even in the negative domain. This characteristic of Leaky ReLU effectively addresses the "dying ReLU" problem, where traditional ReLU neurons become inactive in the negative region, hindering learning. The adaptive nature of Leaky ReLU, especially in larger and more intricate neural architectures, offers advantages in reducing overfitting, a common challenge in deep learning. By enabling non-zero gradients in the negative range, Leaky ReLU ensures that neurons in those regions remain active and continue to learn from the data, promoting improved generalization of the model. Overall, incorporating Leaky ReLU in neural networks serves as a remedy to tackle the vanishing gradient problem associated with standard ReLU activation, leading to enhanced training procedures and facilitating convergence of learning models across various complex tasks.

2.2.6. Proposed hybrid model of DCT-UFS

This study presents the DCT-UFS hybrid model as an innovative approach for diagnosing and classifying microarray datasets. The key advantage of this hybrid model lies in its inherent capability to incorporate feature extraction from both the time domain (original data without DCT) and the frequency domain (DCT-applied data), thereby effectively capturing complementary information from the dataset. The block diagram of the DCT-UFS hybrid model is depicted in Figure 1. The model's workflow commences with the preprocessing step, wherein missing records are meticulously removed from the microarray data to ensure data integrity. To extract meaningful features, the DCT-UFS dimension reduction algorithm is judiciously employed. By leveraging the Discrete Cosine Transform (DCT) in the frequency domain, the model adeptly captures frequency-related patterns and variations inherently present in the data. Moreover, the DCT concentrates signal energy in a succinct number of coefficients, thereby facilitating noise reduction and augmenting the representation of pivotal features. A salient strength of the DCT-UFS hybrid model lies in its ability to perform dimension reduction, culminating in a more compact dataset representation. This efficacious dimensionality reduction strategy effectively mitigates the curse of dimensionality, improves computational efficiency, and holds the potential to preempt overfitting in subsequent stages of the model.

After the feature extraction phase, the reduced dataset is systematically fed into a standard artificial neural network (ANN). The ANN architecture entails multiple layers of neurons, and the Leaky ReLU activation function is proficiently utilized to introduce non-linearity and adeptly capture intricate relationships within the data. By leveraging Leaky ReLU, the model effectively circumvents the vanishing gradient problem that may impede the training of deep neural networks. The concluding layer of the ANN thoughtfully adopts the Sigmoid activation function, rendering it particularly suitable for binary classification tasks. Table 2 furnishes a comprehensive overview of the architecture and activation functions deftly employed in the artificial neural network model. By seamlessly integrating feature extraction from both the time and frequency domains, the DCT-UFS hybrid model achieves a holistic and robust representation of the microarray dataset. This comprehensive representation perceptibly contributes to the amplified classification accuracy and heightened diagnostic performance, underscoring the promise and efficacy of the proposed model as a discerning and potent approach for the meticulous analysis of microarray data.

Additionally, the study performed tests with three different classifiers, namely kNN, SVM, and CNN, to compare their performance with the artificial neural network (NN) used as the primary classifier. Figure 2 depicts a diagram showing four separate scenarios designed for each classifier.

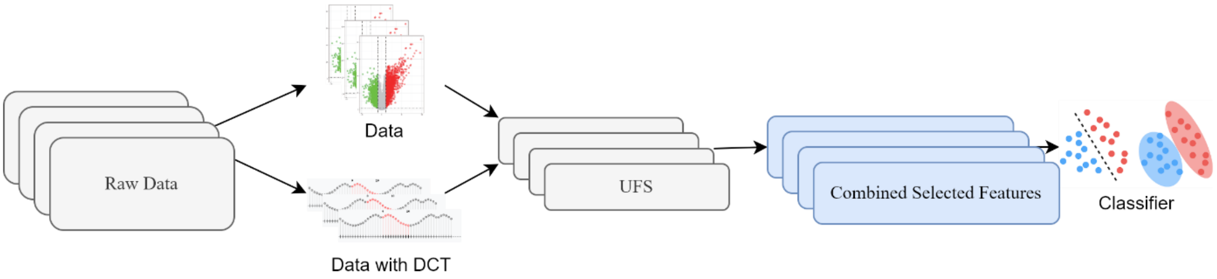

Figure 1. The architecture of the DCT-UFS hybrid model.

Table 2. Architecture of the artificial neural network (ANN)

Layer Number	Layer Type	Output Shape	Activation Function
1	Dense_1 (Dense)	(None, 64)	Leaky ReLU (alpha=0.9)
2	Dropout (Dropout rate: 0.5)	(None, 64)	-
3	Dense_2 (Dense)	(None, 16)	Leaky ReLU (alpha=0.9)
4	Dropout (Dropout rate: 0.5)	(None, 16)	-
5	Dense_3 (Dense)	(None, 1)	Sigmoid

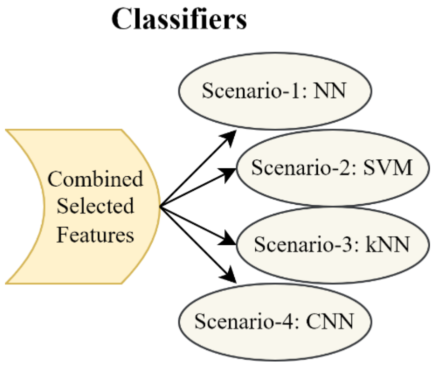

Figure 2. A block diagram illustrating the scenarios designed for different classifiers.

The CNN model depicted in Table 3 is specifically designed for the classification of time series data. Comprising 1D convolutional layers, max pooling, batch normalization, and dropout layers, the model boasts a total of 4,241 parameters. During the training process, it utilizes the 'adam' optimizer and 'binary_crossentropy' loss function, with performance evaluation conducted through the accuracy metric.

Table 3. Architecture of the Convolutional Neural Network

Layer Number	Layer Type	Output Shape	Activation Function
1	Conv1D	(None, feature_count, 64)	LeakyReLU(alpha=0.9)
2	MaxPooling1D	(None, feature_count /3, 64)	None
3	BatchNormalization	(None, feature_count /3, 64)	None
4	Dropout	(None, feature_count /3, 64)	None
5	Conv1D	(None, feature_count /3, 16)	LeakyReLU(alpha=0.9)
6	MaxPooling1D	(None, feature_count /9, 16)	None
7	BatchNormalization	(None, feature_count /9, 16)	None
8	Dropout	(None, feature_count /9, 16)	None
9	Flatten	(None, feature_count *16/9)	None
10	Dense	(None, 1)	Sigmoid

3. Results and Discussion

The study involved conducting experiments with hybrid models that incorporated dimension reduction, machine learning, and deep learning techniques to diagnose diseases using eight different microarray datasets related to LEUKEMIA, Colon, Prostate, Ovarian, DLBCL, Breast Cancer, Breast Cancer-2, and CNS diseases. Dimension reduction was achieved using Discrete Cosine Transform (DCT) and Unsupervised Feature Selection (UFS) methods, while classification utilized NN, SVM, kNN, and CNN models. The models were tested on a computer equipped with an Intel Xeon E5-2630 2.3 GHz CPU and 12 GB RAM.

3.1. Evaluation Criteria

The datasets were divided into training and test datasets using three different ratios, ranging from 60% to 80%, using the hold-out method. The division of the data was done randomly. In addition to the hold-out method, for robust evaluation, 10-fold cross-validation was applied to each proposed model. The average experimental results were then calculated for accuracy, sensitivity, specificity, precision values, the Kappa score, and the Macro F1 score, as given in Equations 5, 6, 7, 8, 9, and 10, respectively.

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} x\ 100 \qquad (5)$$

$$Sensitivity = \frac{TP}{TP + FN} x\ 100 \qquad (6)$$

$$Specificity = \frac{TN}{FP + TN} x\ 100 \qquad (7)$$

$$Precision = \frac{TP}{TP + FP} x\ 100 \qquad (8)$$

$$Kappa = \frac{P_o - P_e}{1 - P_e} \qquad (9)$$

$$Macro\ F1 = \frac{2\ x\ Macro\ Precision\ x\ Macro\ Recall}{Macro\ Precision + Macro\ Recall} \qquad (10)$$

The accuracy metric measures the proportion of correctly classified instances, encompassing true positives and true negatives, concerning the total instances. True positive (TP) refers to correctly predicted positive instances, while true negative (TN) indicates correctly predicted negative instances. False positive (FP) represents instances that were incorrectly predicted as positive, and false negative (FN) represents instances that were incorrectly predicted as negative. Sensitivity (recall) evaluates the model's ability to correctly identify actual positive instances, while specificity assesses the model's capability to correctly identify actual negative instances. The Kappa score assesses the agreement between the predicted classifications and the actual classifications, considering the agreement that could have occurred by chance. P_o represents the relative observed agreement, and P_e is the hypothetical probability of chance agreement. Precision measures the proportion of correctly predicted positive instances out of all instances predicted as positive. The Macro F1 score aims to strike a balance between precision and recall on a per-class basis by calculating the F1 score independently for each class and then taking the unweighted average (macro-average) across all classes. These equations provide a comprehensive evaluation of the proposed models, considering their ability to accurately classify instances, handle imbalanced datasets, and measure the agreement between predicted and actual classifications. The results obtained for each metric allow for a comparison of the effectiveness of the different models studied.

Furthermore, given the stochastic nature of the neural network model, it was executed ten times to account for any variations in the outcomes. The final evaluation metric was then calculated as the average of these ten runs. This approach ensures a more robust and dependable assessment of the neural network model's performance, accounting for potential variability in its predictions across multiple executions.

3.2. Results

This study utilizes sensitivity, specificity, accuracy, macro F1 (MF1), and Cohen's Kappa coefficient (K) as performance criteria. The results of the proposed model were obtained through 10-fold cross-validation and hold-out. Furthermore, experiments were conducted on eight different datasets in four different scenarios.

In these experiments, four different classifiers were employed, and the results obtained using the hold-out method revealed that the NN (Artificial Neural Network) based classifier exhibited the highest performance among all classifiers. The detailed outcomes of these experiments can be found in Table 4.

The detailed outcomes of these experiments, obtained using the 10-fold cross-validation method, are presented in Table 5. The highest results are highlighted in bold for easy identification. Upon examination of the table, it can be generally observed that the NN model outperforms the other models.

Table 4. Experimental results of microarray dataset using the hold-out method with NN

Datasets	Tests	Sensitivity	Specificity	Accuracy	MF1	K
Leukemia	Test1(80-20)	100.00	100.00	100.00	100.00	100.00
	Test1(70-30)	100.00	98.57	99.09	99.04	98.08
	Test1(60-40)	100.00	95.62	97.58	97.57	95.16
	Mean	100.00	98.06	98.89	98.87	97.75
Colon	Test1(80-20)	85.00	84.00	84.61	83.90	67.95
	Test1(70-30)	85.00	82.85	84.21	83.27	66.64
	Test1(60-40)	86.42	80.90	83.99	83.71	67.45
	Mean	85.47	82.58	84.27	83.63	67.35
Prostate	Test1(80-20)	76.66	93.33	88.57	85.60	71.26
	Test1(70-30)	89.28	94.11	91.93	91.82	83.65
	Test1(60-40)	90.55	96.08	93.65	93.52	87.05
	Mean	85.50	94.51	91.38	90.31	80.65
Ovarian	Test1(80-20)	100.00	100.00	100.00	100.00	100.00
	Test1(70-30)	100.00	100.00	100.00	100.00	100.00
	Test1(60-40)	99.83	100.00	99.90	99.89	99.76
	Mean	99.94	100.00	99.97	99.96	99.92
DLBCL	Test1(80-20)	100.00	100.00	100.00	100.00	100.00
	Test1(70-30)	98.88	89.33	92.91	92.66	85.40
	Test1(60-40)	99.09	90.99	93.87	93.52	87.09
	Mean	99.32	93.44	95.59	95.39	90.83
BRC	Test1(80-20)	93.75	80.99	88.84	87.99	76.01
	Test1(70-30)	94.80	68.57	85.38	83.15	66.58
	Test1(60-40)	94.06	74.00	86.34	85.02	70.20
	Mean	94.20	74.52	86.85	85.39	70.93
BRC-2	Test1(80-20)	87.77	87.27	87.50	87.39	74.83
	Test1(70-30)	80.90	89.47	86.33	85.26	70.55
	Test1(60-40)	83.84	91.92	89.23	87.89	75.80
	Mean	84.17	89.55	87.69	86.85	73.73
CNS	Test1(80-20)	100.00	100.00	100.00	100.00	100.00
	Test1(70-30)	91.66	100.00	97.22	96.65	93.35
	Test1(60-40)	80.00	97.22	92.91	89.95	80.03
	Mean	90.55	99.07	96.71	95.53	91.13

Table 5. Experimental results of microarray dataset using the 10-fold cv method with classifiers

Datasets	Classifiers	Sensitivity	Specificity	Accuracy	MF1	K
Leukemia	NN	100.00	94.00	95.89	95.61	91.41
	SVM	95.00	98.00	97.14	96.32	92.83
	KNN	85.00	100.00	95.71	93.63	87.64
	CNN	85.00	86.00	86.07	79.19	69.47
Colon	NN	90.00	81.66	87.38	85.99	72.31
	SVM	95.00	78.33	89.04	85.13	73.21
	KNN	92.50	70.00	84.28	79.89	62.68
	CNN	67.50	60.00	64.76	68.23	26.11
Prostate	NN	96.00	98.00	97.09	97.03	94.13
	SVM	88.66	96.00	92.27	92.24	84.59
	KNN	90.33	98.00	94.18	94.12	88.33
	CNN	68.33	78.00	72.72	67.05	46.33
Ovarian	NN	100.00	100.00	100.00	100.00	100.00
	SVM	100.00	96.66	98.81	98.64	97.30
	KNN	100.00	95.55	98.43	98.20	96.43
	CNN	100.00	91.11	96.87	97.77	92.61
DLBCL	NN	100.00	100.00	100.00	100.00	100.00
	SVM	100.00	98.00	98.57	98.44	96.95
	KNN	85.00	96.33	93.39	91.11	82.33
	CNN	95.00	70.00	76.60	69.42	53.56
BRC	NN	95.41	87.50	92.17	91.27	82.79
	SVM	90.83	78.00	85.89	84.01	68.59
	KNN	96.52	71.00	87.43	84.85	70.40
	CNN	88.33	69.00	81.08	85.79	56.99
BRC-2	NN	92.50	90.66	91.55	91.30	82.92
	SVM	79.00	79.00	79.11	77.88	57.82
	KNN	71.50	83.00	77.22	76.08	54.14
	CNN	65.50	71.00	67.88	61.65	36.04
CNS	NN	96.66	90.00	91.66	91.52	84.00
	SVM	91.66	89.16	90.00	89.30	79.04
	KNN	73.33	95.00	86.66	84.11	69.64
	CNN	33.33	80.83	64.99	36.66	15.23

Table 6 provides a comparative analysis of studies conducted on microarray datasets for diverse cancer types. These investigations aim to assess the distinctions between Leukemia, Colon, Prostate, Ovarian, Diffuse Large B-cell lymphoma (DLBCL), Breast Cancer (BRC), Breast Cancer Type 2 (BRC-2), and Central Nervous System (CNS) tumors. In numerous scholarly works (Alrefai and Ibrahim, 2022; Gunavathi and Premalatha, 2014; Panda, 2020; Sönmez et al., 2021), evolutionary computations have been employed. However, due to the substantial computational burden associated with numerous iterative calculations and population-based optimizations, evolutionary computations can lead to significant delays during the training and testing processes. Our proposed hybrid DCT-UFS (Discrete Cosine Transform - Univariate Feature Selection) method aims to minimize processing time by reducing data dimensions and requiring fewer computations through coefficient representations. While DCT is employed for dimensionality reduction, UFS evaluates the individual feature's relationship with the target variable during the feature selection process. This approach reduces computational time by independently examining each feature and presents a straightforward solution. Consequently, the hybrid DCT-UFS method completes both training and testing procedures in a matter of seconds. Furthermore, it has demonstrated superior performance in 5 out of 8 different datasets and achieved competitive results on the remaining 3 datasets.

4. Conclusion

The increasing global prevalence of cancer has led to a significant rise in generating and analyzing microarray data from tissue samples. The accurate classification of this data is crucial for disease diagnosis and distinguishing between various tumor types. However, classifying microarray data is highly complex due to challenges like a limited number of samples, a large number of features, and the presence of data noise. In particular, genomic data analysis, including microarrays and RNA sequencing, often involves datasets with thousands of genes but only a few samples. This high data dimensionality further complicates the analysis process.

Table 6. Comparison of the studies on microarray datasets

Authors	Leukemia	Colon	Prostate	Ovarian	DLBCL	BRC	BRC-2	CNS
(Gunavathi and Premalatha, 2014) 5-Fold CV	-	85.00	92.68	-	84.00	-	-	81.25
(Gunavathi and Premalatha, 2014) 5-Fold CV	-	95.00	65.25	-	100.00	-	-	81.25
(Kumar and Rath, 2015) Hold-Out	97.22	-	-	98.42	-	-	-	-
(Gao et al., 2017) 10-Fold CV	-	89.09	96.54	-	-	-	-	-
(Gao et al., 2017) 10-Fold CV	-	90.32	96.08	-	100.00	-	-	-
(Medjahed et al., 2017) Hold-Out	95.81	-	-	98.19	-	-	-	-
(Sun et al., 2018) 10-Fold CV	-	88.00	80.00	-	-	-	-	-
(Panda, 2020) 10-Fold CV	92.11	79.03	-	99.21	-	-	73.43	53.34
(Baliarsingh et al., 2019) 10-Fold CV	-	96.74	-	-	-	-	-	-
(Pragadeesh et al., 2019) 10-Fold CV	-	-	-	-	-	-	-	92.86
(Luo et al., 2019) Hold-Out	-	-	-	-	-	-	-	75.00
(Kilicarslan et al., 2020) Hold-Out	99.86	-	-	98.60	-	-	-	83.95
(Zhang et al., 2020) 10-Fold CV	-	96.74	-	-	-	-	-	90.34
(Othman et al., 2020) 10-Fold CV	-	-	-	-	-	-	-	76.30
(Sönmez et al., 2021) 10-Fold CV	-	98.33	99.00	-	100.00	67.00	90.77	95.00
(Alrefai and Ibrahim, 2022) 10-Fold CV	100	92.86	-	100.00	-	-	86.36	85.71
This work (DCT-UFS with NN) Hold-Out	98.89	84.27	91.38	99.97	95.59	86.85	87.69	96.71
This work (DCT-UFS with NN) 10-Fold CV	95.89	87.38	97.09	100.00	100.00	92.17	91.55	91.66
This work (DCT-UFS with SVM) 10-Fold CV	97.14	89.04	92.27	98.81	98.57	85.89	79.11	90.00
This work (DCT-UFS with KNN) 10-Fold CV	95.71	84.28	94.18	98.43	93.39	87.43	77.22	86.66
This work (DCT-UFS with CNN) 10-Fold CV	86.07	64.76	72.72	96.87	76.60	81.08	67.88	64.99

Moreover, the majority of genes in these datasets may not directly contribute to the classification process or be relevant to the classes being studied. Therefore, identifying essential genes while disregarding others becomes critical in the classification process. Another significant challenge is the high level of gene interactions compared to other types of data. Some genes can activate others or trigger gene expressions, making the analysis of microarray data intricate and affecting result accuracy. To address these challenges, the DCT-UFS method can be employed in microarray data analysis. DCT-UFS is an effective technique used for dimensionality reduction and feature selection. It transforms high-dimensional data into smaller, meaningful features, which aids in identifying important genes and considering gene interactions.

The DCT (Discrete Cosine Transform) reduces processing time by reducing data dimensionality and representing it with less computationally demanding coefficients. On the other hand, UFS (Univariate Feature Selection) evaluates each feature's relationship independently with the target variable during selection. Its efficiency lies in analyzing each feature in isolation, disregarding relationships with other features, which reduces computation time and provides a straightforward approach. While literature reviews often mention the usage of evolutionary computations, such methods typically require substantial computational resources due to numerous repetitive calculations and population-based optimization requirements. This study proposed the utilization of the DCT-UFS method in their microarray data analysis. DCT-UFS serves as an effective approach for reducing data dimensionality and selecting relevant features. Transforming high-dimensional data into meaningful features aids in identifying essential genes and considering gene interactions. The insights gained from utilizing the computationally efficient DCT-UFS method in microarray data analysis may ultimately contribute to the development of improved diagnostic and therapeutic strategies for cancer. Future work will focus on validating these findings in larger and more diverse datasets, as well as exploring the potential for integrating DCT-UFS with other machine learning techniques to further enhance the accuracy and efficiency of cancer classification.

Author Contributions

The percentage of the author contributions is presented below. The author reviewed and approved the final version of the manuscript.

	E.E.
C	100
D	100
S	100
DCP	100
DAI	100
L	100
W	100
CR	100
SR	100
PM	100

C=Concept, D= design, S= supervision, DCP= data collection and/or processing, DAI= data analysis and/or interpretation, L= literature search, W= writing, CR= critical review, SR= submission and revision, PM= project management.

Conflict of Interest

The author declared that there is no conflict of interest.

Ethical Consideration

Ethics committee approval was not required for this study because of there was no study on animals or humans.

Acknowledgements

The author gratefully acknowledges the valuable comments and suggestions provided by the anonymous reviewers and editors, which significantly improved the quality of this article. Additionally, AI tools have been utilized to refine grammar and spelling, enhance stylistic elements, and elevate the overall clarity of the text.

References

Alon U, Barkai N, Notterman DA, Gish K, Ybarra S, Mack D, Levine AJ. 1999. Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. Proc National Acad Sci, 96(12): 6745-6750.

Alrefai N, Ibrahim O. 2022. Optimized feature selection method using particle swarm intelligence with ensemble learning for cancer classification based on microarray datasets. Neural Comput Appl, 34(16): 13513-13528.

Baliarsingh SK, Vipsita S, Muhammad K, Bakshi S. 2019. Analysis of high-dimensional biomedical data using an evolutionary multi-objective emperor penguin optimizer. Swarm Evol Comput, 48: 262-273.

Efe E, Özşen S. 2022. Comparison of time-frequency analyzes for a sleep staging application with CNN. J Biomimetics, Biomater Biomedic Eng, 55: 109-130.

Efe E, Ozsen S. 2023. CoSleepNet: Automated sleep staging using a hybrid CNN-LSTM network on imbalanced EEG-EOG datasets. Biomed Signal Proces Control, 80: 104299.

Efe E, Yavsan E. 2024. AttBiLFNet: A novel hybrid network for accurate and efficient arrhythmia detection in imbalanced ECG signals. Math Biosci Eng, 21(4): 5863-5880.

Er MJ, Chen W, Wu S. 2005. High-speed face recognition based on discrete cosine transform and RBF neural networks. IEEE Transact Neural Networks, 16(3): 679-691.

Gao L, Ye M, Lu X, Huang D. 2017. Hybrid method based on information gain and support vector machine for gene selection in cancer classification. Genomics Proteomics Bioinformatics, 15(6): 389-395.

Golub TR, Slonim DK, Tamayo P, Huard C, Gaasenbeek M, Mesirov JP, … Caligiuri MA. 1999. Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. Science, 286(5439): 531-537.

Gunavathi C, Premalatha K. 2014. Performance analysis of genetic algorithm with kNN and SVM for feature selection in tumor classification. Int J Comput Info Eng, 8(8): 1490-1497.

Guyon I, Weston J, Barnhill S, Vapnik V. 2002. Gene selection for cancer classification using support vector machines. Machine Learn, 46: 389-422.

Kar S, Sharma K Das, Maitra M. 2015. Gene selection from microarray gene expression data for classification of cancer subgroups employing PSO and adaptive K-nearest neighborhood technique. Expert Syst Appl, 42(1): 612-627.

Kilicarslan S, Adem K, Celik M. 2020. Diagnosis and classification of cancer using hybrid model based on ReliefF and convolutional neural network. Medic Hypot, 137: 109577.

Kumar M, Rath SK. 2015. Classification of microarray using MapReduce based proximal support vector machine classifier. Knowledge-Based Syst, 89: 584-602.

Li L, Jiang W, Li X, Moser KL, Guo Z, Du L, Rao S. 2005. A robust hybrid between genetic algorithm and support vector machine for extracting an optimal feature gene subset. Genomics, 85(1): 16-23.

Luo K, Wang G, Li Q, Tao J. 2019. An improved SVM-RFE based on $ F $-statistic and mPDC for gene selection in cancer classification. IEEE Access, 7: 147617-147628.

Maldonado S, Weber R, Basak J. 2011. Simultaneous feature selection and classification using kernel-penalized support vector machines. Info Sci, 181(1): 115-128.

Medjahed SA, Saadi TA, Benyettou A, Ouali M. 2017. Kernel-based learning and feature selection analysis for cancer diagnosis. Appl Soft Comput, 51: 39-48.

Meenachi L, Ramakrishnan S. 2021. Metaheuristic search based feature selection methods for classification of cancer. Pattern Recog, 119: 108079.

Mundra PA, Rajapakse JC. 2009. SVM-RFE with MRMR filter for gene selection. IEEE Transact Nanobiosci, 9(1): 31-37.

Naderi A, Teschendorff AE, Barbosa-Morais NL, Pinder SE, Green AR, Powe DG, Brenton JD. 2007. A gene-expression signature to predict survival in breast cancer across independent data sets. Oncogene, 26(10): 1507-1516.

Orhan H, Yavşan E. 2023. Artificial intelligence-assisted detection model for melanoma diagnosis using deep learning techniques. Math Mod Numeric Sim Appl, 3(2): 159-169.

Othman MS, Kumaran SR, Yusuf LM. 2020. Gene selection using hybrid multi-objective cuckoo search algorithm with evolutionary operators for cancer microarray data. IEEE Access, 8: 186348-186361.

Panda M. 2020. Elephant search optimization combined with deep neural network for microarray data analysis. J King Saud Univ Comput Info Sci, 32(8): 940-948.

Petricoin EF, Ardekani AM, Hitt BA, Levine PJ, Fusaro VA, Steinberg SM, Kohn EC. 2002. Use of proteomic patterns in serum to identify ovarian cancer. The Lancet, 359(9306): 572-577.

Pomeroy SL, Tamayo P, Gaasenbeek M, Sturla LM, Angelo M, McLaughlin ME, Lau C. 2002. Prediction of central nervous system embryonal tumour outcome based on gene expression. Nature, 415(6870): 436-442.

Pragadeesh C, Jeyaraj R, Siranjeevi K, Abishek R, Jeyakumar G. 2019. Hybrid feature selection using micro genetic algorithm on microarray gene expression data. J Intel Fuzzy Syst, 36(3): 2241-2246.

Qaraad M, Amjad S, Manhrawy IIM, Fathi H, Hassan BA, El Kafrawy P. 2021. A hybrid feature selection optimization model for high dimension data classification. IEEE Access, 9: 42884-42895.

Shipp MA, Ross KN, Tamayo P, Weng AP, Kutok JL, Aguiar RCT, Pinkus GS. 2002. Diffuse large B-cell lymphoma outcome prediction by gene-expression profiling and supervised machine learning. Nature Medic, 8(1): 68-74.

Singh D, Febbo PG, Ross K, Jackson DG, Manola J, Ladd C, Richie JP. 2002. Gene expression correlates of clinical prostate cancer behavior. Cancer Cell, 1(2): 203-209.

Sönmez ÖS, Dağtekin M, Ensari T. 2021. Gene expression data classification using genetic algorithm-basedfeature selection. Turkish J Elect Eng Comput Sci, 29(7): 3165-3179.

Sun L, Zhang X, Xu J, Wang W, Liu R. 2018. A gene selection approach based on the fisher linear discriminant and the neighborhood rough set. Bioengineered, 9(1): 144-151.

Van't Veer LJ, Dai H, Van De Vijver MJ, He YD, Hart AAM, Mao M, Witteveen AT. 2002. Gene expression profiling predicts clinical outcome of breast cancer. Nature, 415(6871): 530-536.

Zhang G, Hou J, Wang J, Yan C, Luo J. 2020. Feature selection for microarray data classification using hybrid information gain and a modified binary krill herd algorithm. Interdisciplinary Sci: Comput Life Sci, 12: 288-301.