Improving Classification Performance of Cancer Microarray Data using Hybridization of Binary Grey Wolf and Particle Swarm Optimization

By

Leena Savaliya

A thesis submitted in partial fulfillment
of the requirements for the degree of
Master of Science (MSc) in Computational Sciences

The Faculty of Graduate Studies
Laurentian University
Sudbury, Ontario, Canada

THESIS DEFENCE COMMITTEE/COMITÉ DE SOUTENANCE DE THÈSE
Laurentian Université/Université Laurentienne

Faculty of Graduate Studies/Faculté des études supérieures

Title of Thesis Titre de la thèse	Improving Classification Performance of Cancer Microarray Data using Hybridization of Binary Grey Wolf and Particle Swarm Optimization	
Name of Candidate Nom du candidat	Savaliya, Leena	
Degree Diplôme	Master of Science	
Department/Program Département/Programme	Computational Sciences	Date of Defence Date de la soutenance October 11, 2019

APPROVED/APPROUVÉ

Thesis Examiners/Examinateurs de thèse:

Dr. Kalpdrum Passi
(Supervisor/Directeur de thèse)

Dr. Ratvinder Grewal
(Committee member/Membre du comité)

Dr. Youssou Gningue
(Committee member/Membre du comité)

Approved for the Faculty of Graduate Studies
Approuvé pour la Faculté des études supérieures
Dr. David Lesbarrères
Monsieur David Lesbarrères
Dean, Faculty of Graduate Studies
Doyen, Faculté des études supérieures

Dr. Gulshan Wadhwa
(External Examiner/Examinateur externe)

ACCESSIBILITY CLAUSE AND PERMISSION TO USE

Abstract

In this study, we have proposed hybridization of binary grey wolf Optimization and particle swarm optimization (BGWOPSO) method and we compared this hybrid optimization method with Particle Swarm Optimization (PSO) and Binary Grey Wolf Optimization (BGWO). We have used five significantly different classifier such as K-nearest Neighbor (KNN), Support Vector Machine (SVM), Artificial Neural Network (ANN), Logistic Regression (LR), Random Forest (RF). Furthermore, we have used ratio comparison validation for the 10-folds cross-validation method for feature selection methods. Data sets such as Leukemia, Breast Cancer, and Liver Cancer are used to apply the combinations and measure accuracy as well as the area under ROC. Moreover, the results show that Hybrid optimization method (BWOPSO), significantly outperformed the both binary grey wolf optimization (BGWO) and particle swarm optimization (PSO) method, when using several performance measures including accuracy, selecting the best optimal features. Secondly, combinations of classifier and feature pre-processing method significantly improve the accuracy. Lastly, the AUC value is been displayed in this study.

Keywords:

Hybrid binary optimization, Grey wolf optimization, Particle swarm optimization, Feature, classification.

`

Acknowledgments

First and foremost, I would like to thank and appreciate my supervisor, Dr. Kalpdrum Passi, for his golden teaching, deep understanding and wise guidance in research. His assistance and motivation through all stages of my master study. Although there have been times where I lost my direction when he always used to point me towards the path of excellence. He taught me that baby steps are more important than long jumps, through which, the definition of a teacher, a supervisor, a guide has been redefined. Without him this work had no initiation. Furthermore, I thank my family for their continuous and unparalleled love, help, and support. I am forever indebted to my mother for always being there for me as a friend. I am grateful to my parents for giving me the opportunities that have made me who I am, and I dedicate this milestone to them.

Table of Contents

List of Figures

List of Tables

List of Acronyms

Ant Colony Optimization	(ACO)
Grey Wolf Optimization	(GWO)
Particle Swarm Optimization	(PSO)
Naïve Bayes	(NB)
Support Vector Machine	(SVM)
Genetic Algorithm	(GA)
Differential Evolution	(DE)
Deoxyribonucleic Acid	(DNA)
Random Forest	(RF)
Logistic Regression	(LR)
Area under curve	(AUC)
Convolutional Neural Networks	(CNN)
Artificial Neural Network	(ANN)
Machine Learning	(ML)

List of Built-in Functions

1. str2double

2. Readtable

3. Randperm

4. Size

5. Sort

6 Fopen

7 Textscan

8 Table

9 Kmeans

10 Mean

11 Std

12 Max

13 Fitcsvm

14 fitnet (ANN)

15 Fitcknn

16 fitglm (LR)

17 fitensemble (RF)

Chapter 1

Introduction

1.1 Introduction to Metaheuristic Algorithms

With the growth in the size of datasets, data mining has recently become an important research topic and has gained so much attention from both academia and industry. Datamining contains several preprocessing steps such as data cleaning, data integration and transformation, data filtering, data reduction, knowledge presentation, and pattern evaluation [1]. One of the main preprocessing steps of data mining is called feature selection. Feature selection is a benchmark in data mining to decrease the dataset size for analysis and processing. Which is aims to identifying irrelevant information without changing the accuracy of the classifier. Feature selection takes a subset of important features and eliminates unnecessary [2] [3] and redundant features from the raw data to build an active learning model. Feature selection is critical, not only because of the dimensionality curse but also because of data complexities as well as the quantities of the data. Such as machine learning, data mining, statistics, pattern recognition, and bioinformatics [4].

Moreover, feature selection is classified into two class wrapper methods and the filter method. Wrapper methods evaluate subsets according to the execution of classifiers like Naïve Bayes (NB) or Support Vector Machine (SVM) or Random Forest (RF) whereas, for clustering, wrapper methods calculate subsets based on the performance of a clustering algorithm like K-means. As per a researcher, Wrapper-based methods are better than filter method techniques for the classification of algorithms [5].

Finding an optimal feature is a challenging and expensive task. However, metaheuristic algorithms are developing an essential part of modern optimization [6]. A wide range of metaheuristic algorithms has been employed over the few decades. Metaheuristics are a family of stochastic algorithms for solving the problem of optimizations. Metaheuristics are iterative in behavior. The same pattern is repeated during optimizations until it meets to be stopping criterion. There are so many metaheuristic algorithms that become increasable popular such as particle swarm optimization, grey wolf optimization, Ant colony optimization, genetic algorithm. Metaheuristics show a compelling performance, as they do not have to search the entire search space. We can say that they are not a complete search algorithm. Most of the metaheuristic algorithms are nature-inspired from particle swarm optimization, grey wolf optimization algorithm, and many more. Metaheuristics have mainly been employed to solve feature selection problems including GWO (grey wolf optimization) [7] [8], Genetic Algorithm (GA), Ant Colony Optimization (ACO), PSO (particle swarm optimization) [9] [10], Differential Evolution (DE) [11].

In this study, two nature-inspired algorithms (BGWO, PSO) are used to solve optimization problems. The PSO algorithm is not merely a tool for optimization. However, it also represents the social recognition of human and artificial agents, based on principles of social psychology [12]. A PSO system merges two methods local search methods and global search methods, attempting to balance exploration and exploitation. Particles float around in a multidimensional search space [13]. During the flight, every particle changes position as per its own experience, and knowledge of neighboring particles. It is making utilization of the best situation encountered by itself and neighbor. This algorithm is easy to implement; it monitors three global variables, namely target value, global best (gBest), and stopping cost. Besides, every particle in PSO contains a position vector and a vector to solve the personal best (pBest), and a variable to store the objective value of pBest [14].

GWO is a recently proposed swarm intelligence technique that has gained great popularity in the optimization community. GWO algorithm consists the seeking behavior, searching and the social hierarchy of the grey wolves [15]. Because of less randomness and varying numbers of individuals selected in global and local searching methods, the GWO algorithm is more comfortable to use and focalizes quickly. It has been proved to be more effective than the PSO algorithm and other bionic algorithms.

In this study, we have introduced a hybrid algorithm PSOGWO, how it works, and how it helps to find the best feature in cancer datasets. The rest of the Chapter contains related work, methods, results, conclusion, future work are explained.

1.2 Introduction to Bioinformatics

“Bioinformatics is an interdisciplinary field that develops methods and software tools for understanding biological data." [16] As an interdisciplinary field of knowledge, bioinformatics combines computer science, statistics, mathematics, and engineering for analysis and understand biological data. Bioinformatics is the utilization of information technology, and that is used to the study of living things, usually at the molecular level. Bioinformatics involves the use of a computer to collect, organize and use biological information to give all possible solutions in a field of evolutionary biology. Over the past eras, the quantity and quality of vital information have risen steeply, majorly because of advances in molecular biology and gene technology. The Bioinformatics Organization examined that bioinformatics is used to develop databases, like GenBank from NCBI (National Center for Biotechnology Information), SwissProt from the Swiss

Institute of Bioinformatics, Human Genome Development, that store, organize and index biological information for analysis and for improving results. Moreover, bioinformatics is a field that allows the creation of large databases of biological and health data that can be used to test theories and generate solutions to medical problems that affect all of us.

Firstly, Bioinformatics is a part of various biology areas, which are of great importance. During molecular biology, bioinformatics events such as image processing and signal processing, permit extraction of valuable results from vast amounts of data, In the field of genetics and genomics. It helps in sequencing and explaining genomes and their recognized mutations. It plays a character in the text digging of biological literature and the advancement of biological and gene metaphysics to organize and query biological data. It also plays a role in the study of gene and protein expression and regulation. Bioinformatics tools help in the observation of hereditary and genomic data and more generally in the understanding of evolutionary aspects of molecular biology.

Secondly, bioinformatics mainly involves, management and examination of data primarily obtained from a substantive number of preliminary test runs that may at times provide extensive data set. Moreover, to this effect, there is a need to establish comprehensive mechanisms to aid in the interpretation as well as processing this data and consequently producing accurate information that is needed for research and study purposes. This led to the beginning of bioinformatics, a method that integrates both biology and computer science.

Thirdly, despite the diversity of cells, they all have a life cycle such as they are born, eat, replicate, and die. During the life cycle, a cell makes different decisions through the manifestation in pathways. Three types of basic molecules are present in a cell: deoxyribonucleic acid (DNA), ribonucleic acid (RNA), and proteins. Naturally, DNA, RNA, and proteins can be viewed as strings. DNA is a very long molecule that is composed of four types of bases: adenine (A), thymine (T), guanine (G), and cytosine (C). Similar to DNA, there are four bases in RNA as well. DNA carries the genetic information of a cell and is composed of thousands of genes. Every cell contains the genetic information so that the DNA is duplicated before a cell divides (replication). Lastly, the advances in biotechnology such as the subsequent generation sequencing technologies are occurring at breathtaking speed. Progress and discoveries give competitive advantages to those who are prepared. Moreover, the driving force following the positive competition is not only restricted to technological advancement, but also to the companion data analytic skills and computational processes which are collectively called computational biology and bioinformatics. Without this, the biotechnology-output data by itself is raw and perhaps meaningless.

1.3 Gene Expressions and Microarrays

"Gene expression is the term employed for portraying the interpretation of data contained inside the DNA, the storehouse of hereditary data, into messenger RNA (mRNA) atoms that are then construed into the proteins that complete a large portion of the discriminating capacity of cells." Gene expression is a complicated process that permits cells to react to the changing inward prerequisites and furthermore to external ecological conditions. This system commands genes to express cell and moreover build or reduce the level of expression of genes.

A microarray is a lab tool applied to identify the expression of thousands of genes at the same time. DNA microarrays are microscopic slides that are printed with thousands of little spots in defined spaces, with each place holding a DNA sequence or gene. Often, these slides are introduced to as gene chips or DNA chips. The DNA molecules are connected to each slide, which acts as probes to recognize gene expression, also known as the transcriptome or the set of messenger RNA (mRNA) transcripts expressed by a group of genes.

Microarrays can be divided into two types, the first is Dual Channel Microarrays and the second is known as Single Channel Microarrays. The one we use for our study is the Single Channel Microarrays. In these microarrays, different samples are subjected to hybridization after it is marked with fluorescent color. These microarrays include the ability of declaration irrefutably. The latter procedure is performed utilizing a methodology called photolithography. Because biological processes are naturally complex and irregular expression patterns can always exist among genes that belong even to the same functional classes [17]. Moreover, since quality articulation information is boisterous and have high dimensionality, excellence capacity expectation, regardless of whether it is figured as a grouping or order issue, is difficult and usual bunching and characterization procedures, which are not initially created to manage quality articulation information, may not generally be the most appropriate.

In Dual-Channel Microarrays, standard successions and ordinary arrangements are marked with two unique fluorescent shades. Both these DNA patterns are hybridized collectively on the DNA Microarrays, and a range of fluorescence energies emitted by the two colors is estimated to determine the differential description level.

There are unique sorts of microarrays, for example, DNA microarrays, protein microarrays, and carb microarrays. The microarray discovery was advanced out of the necessity to focus measures of distinct substances inside a blend of various materials.

It is possible to achieve gene expression data almost cheaply from micro-arrays. So, this commences believing that the genes can inform us who will grow or have a disease. Reasonably one can observe the stage of the illness to enable effective remedies. However, we are currently at the point where there are numerous challenges to estimating the likelihoods for genes to be utilized in analysis and solution. There are typically several more genes that may be associated than examples for any given illness.

1.4 Importance of Optimization

The overwhelming amount of data and its continuous accumulation through online tools has significantly changed the classical approaches to data processing and raised a need for scalable and efficient algorithms with importance on high-performance computing aimed at learning hidden patterns from such data. Knowledge extraction involves various areas such as computer science, statistics, optimization, and more, where collectively development of such tools is referred to as Machine Learning (ML).

One of the most straightforward definitions for optimization is "doing the maximum with the least," said Gomez, Johnson. Optimization is the manner of finding the most effective or favorable value or condition. Optimization determines to achieve the "best" design relative to a set of prioritized criteria or constraints. These contain maximizing factors, such as productivity, strength, reliability, longevity, efficiency, and utilization. Engineers are repeatedly forced to identify a few appropriate design solutions and then decide which one best meets the need of the client. This decision-making process is known as optimization.

In our research, we have used the main two optimization algorithm, such as Grey wolf Optimization and Particle Swarm Optimization. The present grey wolf optimization algorithm has some disadvantages, such as slow convergence speed, low precision and many more. At the same time, the knowledge of PSO is familiarized, which utilizes the best value of the individual and the best value of the wolf pack to update the position information of each grey wolf. grey wolf optimization algorithm combined with particle swarm optimization (GWOPSO) and in this new algorithm, helps to initiate the individuals' position, which can increase the diversity of the wolf pack.

1.5 Classification Techniques

In this research, we apply the classification technique to measure and compare at the distinction between various optimization techniques. Any classification plans utilize many parameters to describe each object. These features are significant to the data being examined. Here we are discussing strategies for supervised learning. In supervised learning, there are labels on the data, and the algorithms figure out how to predict the output from the input data. We furthermore have a set of items with known types. A training set is used by the order projects to figure out the arrangement of the elements into the required classes. This training set is utilized to decide how the parameters ought to be weighted or combined so we can assign different categories of items. In the creaming stage, the trained classifiers can be employed to sharpen the classifications of items using new examples called the testing set.

Five classification methods that are Support Vector Machine, Random Forest, k-Nearest-neighbor, Neural Network, and Logistic Regression, are chosen in our study. The features of these methods will be discussed in a further chapter. We register the accuracy and AUC (Area under ROC) value for analysis.

1.6 Optimization for Feature Selection Methods

As we know, some optimization techniques work a feature selection. Modern-day datasets are incredibly loaded with information composed of a massive number of data sources. The enormous dimension of data increases the computation cost and reduces the performance of a machine learning model. If the dataset contains irrelevant, redundant and unnecessary features that are not beneficial to the enhancement of a predictive model. Learning models will encounter the problem of overfitting with a vast number of features. One of the best solutions to solve this problem is to select a subset of attributes from the data that are most suitable or appropriate for the challenge. Feature selection algorithms can reduce the number of features to build a machine learning model by verifying different combinations of attributes in a dataset. The wrapper-based feature selection techniques are usually proposed to advance the efficiency of classification models by choosing an optimal subset of features from feature space.

But Discovering the right combination of features is hugely a demanding task. Different optimization algorithms are used for attribute selection, such as Genetic Algorithm (GA), and Particle Swarm Optimization (PSO) by many researchers to improve the performance of the classifiers. It will give us more effective result.

1.7 Objectives of the Study and Outline of the Thesis

In this study, we proposed optimization method such as grey wolf optimization, PSO and hybrid PSO. We have also used binary GWOPSO optimization, where we are trying to find the specific condition that produces the best performance among five classifiers.

The thesis is organized as below:

Chapter 2, the literature review is presented, which highlights some existing literature pertaining to the fields of microarray data.

Chapter 3, Details about the datasets used in the study, along with some information about the specifications of the different algorithms and tools used are elucidated.

Chapter 4, It contains some feature selection method in detailed. Such as embedded method, and nature-inspired algorithm.

Chapter 5 represented methodology and tools.

Chapter 6 includes the discussion which takes in the broad results and generated intuitive insights based on them. Presents the conclusion and discusses upon the future work that can be performed.

Chapter 2

Literature Review

Owing to no noticeable early symptoms of cancer, most patients are diagnosed at an advanced stage, which usually results in huge costs with a more pitiable prognosis. In addition to unlimited growth, cancer cells invade the surrounding healthy tissue and even move through the body's circulatory system or indifferent system to other parts of the body. The probability of recurrence or metastasis after operation is higher than 90% after five years, as a cancer surgery is not exact. To date, the problem of completely removing the surviving cancer cells is not solved, and the recurrence rate and fatality rate of cancer are still very high. If we can make full use of possible human expression profiles and accomplish repeatable examinations, there is no doubt that it will bring great support to cancer patients. The analysis of microarray gene expression data and protein expression data can be used to seize the information of physiological activities at the molecular level, which is widely used in the field of biomedicine. However, a huge number of irrelevant and redundant values exist in expression profiles. Moreover, the high dimensionality and small sample bring great difficulties to data processing. Thus, researchers have proposed various methods to deal with these problems [18].

Breast cancer is the standard invasive cancer in females global and is a significant cause of death. Scientists claim that there exists no specific way to know and say why one person develops the disease while another does not. We are only attentive to certain risk circumstances that can influence the chances of being diagnosed with one. The ability to identify beforehand is the only way to tackle breast cancer. Appropriate tests are needed to differentiate benign tumors from malignant ones. The most common tests include Mammograms, Breast Ultrasound, Magnetic resonance imaging (MRI), Duct gram, and Biopsy. The biopsy is the best and only way to know if the breast tumor is cancerous. Where a small sample is taken from a suspected area and is the test in the lab. And positive detection of breast cancer through Biopsy can reach as small as 10%. Many statistical techniques and cognitive science approaches like Artificial Intelligence are lately being used to detect the "malignant" group in a patient.

Breast Cancer Diagnosis (BCD) problem is labeled as a classification dilemma. Researchers from different fields are attracted to analyze this problem and are using Machine Learning (ML) algorithms, Artificial Neural Networks, Support Vector Machines, and other data mining techniques. ANN, SVM, and PNN have become practical choices in modeling classification problems due to their capability to capture complex nonlinear associations among variables. Recently, functional weaves have been proposed as a very efficient scheme for statistical design recognition problems and nonlinear complex prediction. Machine learning can be utilized to get a cheap and reliable diagnosis. One of the methods of machine learning is the Feed-Forward Artificial Neural Network. The FFANN can be used to classify them into malignant and benign. The objective of this experiment is to classify and diagnose breast cancer [19].

When we are talking about Leukemia, Leukemia is a pathology that affects young people and adults, causing premature death and several other symptoms. Computer-aided systems can be used to decrease the possibility of prescribing unsuitable treatments and support specialists in the diagnosis of this disease. There is a rising usage of Convolutional Neural Networks (CNNs) in the classification and diagnosis of medical image difficulties. Though, the training of CNNs requires a large set of images. To overcome this problem, we can use transfer learning to extract images features for advance classification. We can test three state-of-the-art CNN architectures, and the features were carefully chosen according to their gain ratios and utilized as input to the Support

Vector Machine classifier. The approach aims to correctly classify images with different characteristics taken from different image databases and does not require a segmentation process. They built a new database from the union of three distinct databases presented in the literature to validate the proposed methodology. The methodology achieved hit rates overhead 99% and outperformed nine methods found in the works. An approach for leukemia diagnosis is proposed using transfer learning and SVM classifier. They extracted features using transfer learning due to the lack of a large database. We created three hybrid image databases for method validation. In our experiments, the classifier SVM outperformed the KNN, MLP, and Random Forest. The achieved results overcome the state-of-art methods [20].

Metaheuristic optimization uses metaheuristic algorithms to address optimization problems. Basically, optimization is everywhere, from business design to economics. There are so many well knows optimization algorithm. A feature selection method utilizing a combination of differential evolution (DE) optimization method and also presented a repair mechanism founded on feature distribution measures has been proposed by Rami N. Khushaba et al. [21]. The novel method, condensed as DEFS, utilizes the DE float number optimizer in the combinatorial optimization trouble of feature selection. To make the results structured by the float-optimizer suitable for feature selection, a roulette wheel structure was assembled and supplied with the probabilities of features distribution. Those probabilities were created among the iterations by distinguishing the features that contribute to the most promising solutions. Their presented DEFS was applied to search for optimal subsets of features in datasets with varying dimensionality.

A newest PSO variant, known as competitive swarm optimizer (CSO) that was dedicated to large-scale optimization, for resolving high-dimensional feature selection troubles has been developed by Shenkai Gu et al. [22]. Adding, the CSO, which was originally developed for continuous optimization, it was conceived to perform feature selection that can be examined as a combinatorial optimization problem. A documentation technique was also inducted to reduce computational cost. They trial on total six benchmark datasets and associated to the canonical PSO-built and a state-of-the-art PSO variant for feature selection, their introduced CSO-based feature selection algorithm not only chooses a much smaller number of features but also gives best result in classification performance as well.

A method HPSO-LS applies a local search strategy which was embedded in the particle swarm optimization to select the less correlated and salient feature subset has been estimated by Parham Moradi et al. [23]. The main aim of the local search technique was to guide the search process of the particle swarm optimization to choose distinct features by conceiving their correlation information. Moreover, the suggested method applies a subset size determination scheme to choose a subset of features with minimized size. The performance of the suggested method has been estimated on 13 benchmark classification problems and equated with five state-of-the-art feature selection methods. Moreover, HPSO-LS has been equated with four well-known filter-based methods admitting the information gain, term variance, fisher score and mRMR and five well-known wrapper-based methods admitting genetic algorithm, particle swarm optimization, simulated annealing and ant colony optimization. The results established that the introduced method develops the classification accuracy equated with those of the filter based and wrapper-based feature selection methods.

Prediction and classification of biological inferences of complex, high-dimensional, and voluminous data is one big task in bioinformatics. However, with the rapid increase in size and numbers of features in bioinformatics, it is critical using tools and algorithms in the process of classification automation. Moreover, several reports are evidence that Classification methods such as SVM, ANN, KNN, LR, and RF, have been involved in solving the bioinformatics problems with remarkable accuracy. A non-linear method of multivariate analysis weighted digital analysis (WDA) and estimated they could predict Breast cancer by utilizing volatile biomarkers.

In classification tasks, the performance of classifier improves with an increase in dimension until an optimal number of features are selected. After that, a further increase in the proportion without a rise in the samples causes a degrade in performance. Which is other terms is called "curse of dimensionality." The solution to this very problem, the first way is to find out which subset of the original data set is of vital importance for the classification tasks. In, a comparative study demonstrated that it is possible to use a feature selection method to solve this problem. Authors have used SVM with a different kernel and KNN method for a classification task and give a ranking method to two gene datasets. The results displayed that, the importance of informative gene ranking to classify test samples accurately present a hybrid method for feature selection. The increased number is irrelevant, and redundant features have decreased the accuracy of the classifier as well as increased the computational cost and reinforced what is called the curse of dimensionality. Thus, the feature selection method becomes a possible solution.

Chapter 3

Datasets

3.1 Leukemia Cancer Dataset

Blood cancer draws a large group of different malignancies. Which include cancer of bone marrow, lymphatic system. Leukemia raises within the bone marrow; they can affect the bone marrow's abilities to produce healthy blood cells, including white blood cells and platelets.

They are harmful neoplasms of hematopoietic stem cells. Leukemia cancer data was downloaded from UCI repository. The entire number of genes to be experienced is 7129, and several samples to be tested are 72, which are all acute Leukemia patients, either acute lymphoblastic Leukemia (ALL) or acute myelogenous Leukemia (AML).

More accurately, the count of ALL is 47, and the count of AML is 25. The dataset has been normalized at an earlier stage.

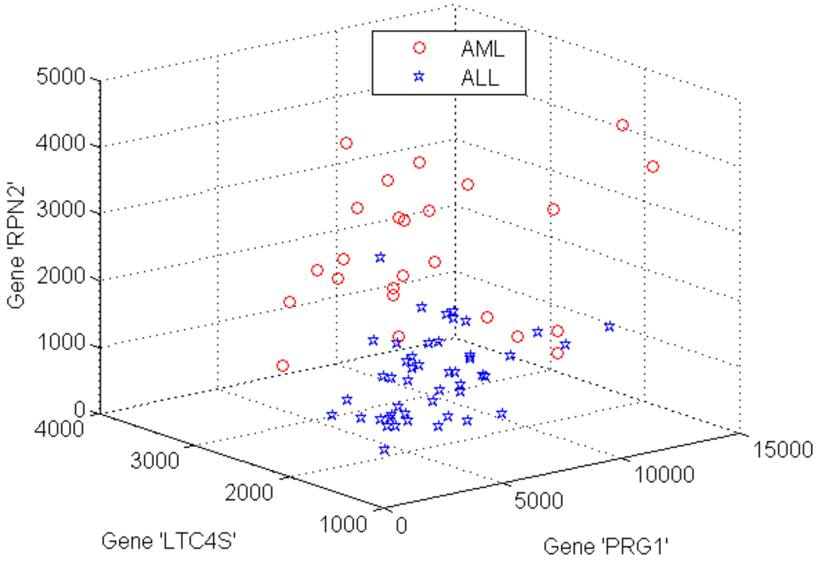


Figure 1. Projection of ReliefF's top genes in the Leukemia [24]

3.2 Breast Cancer Dataset

Breast cancer is one of the principal reasons for female fatality all over the world and is the first field of research for quite a long time with smaller improvement than expected. Features are subtracted from a digitized form of a fine needle aspirate (FNA) of a breast mass. We have used two types of breast cancer dataset such as microarray dataset and clinical dataset.

3.2.1 Microarray Breast Cancer Dataset

Microarray breast cancer dataset was downloaded from NCBI [25]. 200 samples, including 60 normal samples and 140 tumor samples, collected from patients of breast cancer. Total number of genes to be tested is 2520.

3.2.2 Clinical Breast Cancer Dataset

Breast cancer data was downloaded from UCI repository. Attribute type is real number. There are 569 instances and It contains 32 attributes.

Attribute of breast cancer dataset contains ID number and Diagnosis (M = malignant, B = benign) In breast cancer, ten real-valued features are computed for each cell nucleus, and it is listed below: Radius (mean of distances from center to points on the perimeter), texture (standard difference of gray-scale values), perimeter , area , regularity (local variation in radius lengths), compactness (perimeter$_2$ / area - 1.0), concavity (severity of curved portions of the contour), concave points (number of concave portions of the contour) , symmetry and last of all fractal dimension ("coastline approximation" - 1) [26] [27] [28].

The mean, standard error, and largest (mean of the three largest values) of these features were calculated for every image, resulting in 30 features. For example, field 3 is Mean Radius. Field 13 is Radius SE(standard error). Field 23 is Worst Radius.

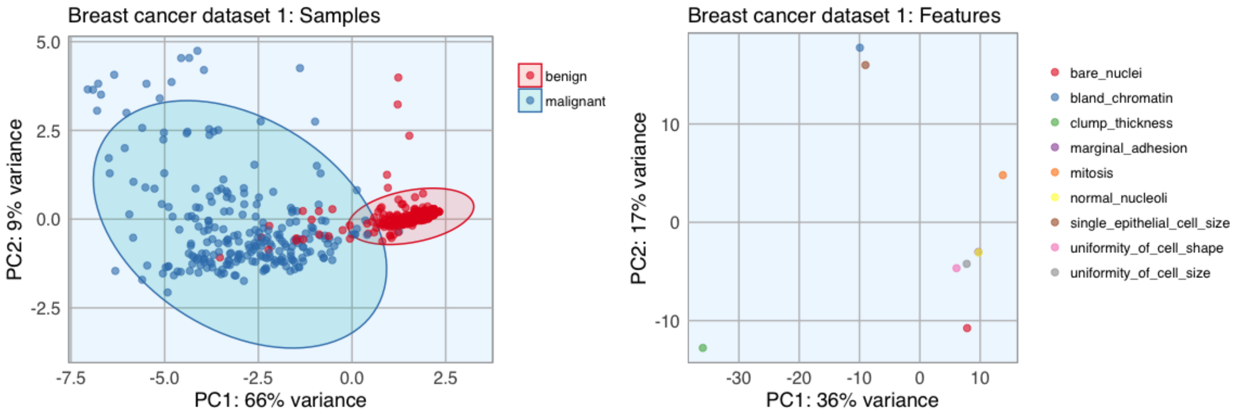


Figure 2. Breast cancer dataset [29]

3.3 Liver Cancer Dataset

Liver cancer is cancer that begins in the cells of your liver [30]. There are Several types of cancer can be developed in the liver, but the most common type of liver cancer is hepatocellular carcinoma, which starts from the main type of liver cell (hepatocyte). Other kinds of liver cancer, such as intrahepatic cholangiocarcinoma and hepatoblastoma, are much less frequent.

3.2.1 Microarray Liver Cancer Dataset

Microarray liver cancer dataset was collected from NCBI [31]. The National Center Biotechnology Information develops learning knowledge and health by offering access to biomedical and genomic information.

The dataset contains 2670 genes. More accurately 190 samples, including 150 tumor samples and 40 normal samples.

3.2.2 Clinical Liver Cancer Dataset

Liver cancer for HCC was collected from the University Hospital in Portugal and embraces several demographics, risk factors, laboratory and overall continuation features of 165 real patients diagnosed with HCC [32]. The dataset includes 49 selected feature and as per European Association for the Study of the Liver - European Organization for Research and Treatment of Cancer and Clinical Practice Guidelines, which are the current state of-the-art on the management of HCC [33].

This is a heterogeneous dataset, with 23 quantitative variables, and 26 qualitative variables. Overall, missing data describes 10.22% of the whole dataset, and only eight patients have full information in all fields (4.85%). The target variables are the survival at one year and were encoded as a binary variable: 0 (dies) and 1 (lives).

An exhaustive description of the HCC dataset that contains feature's type or scale, range, mean or mode, and missing data percentages moreover, it contains 165 instances and 49 attributes [32] [33].

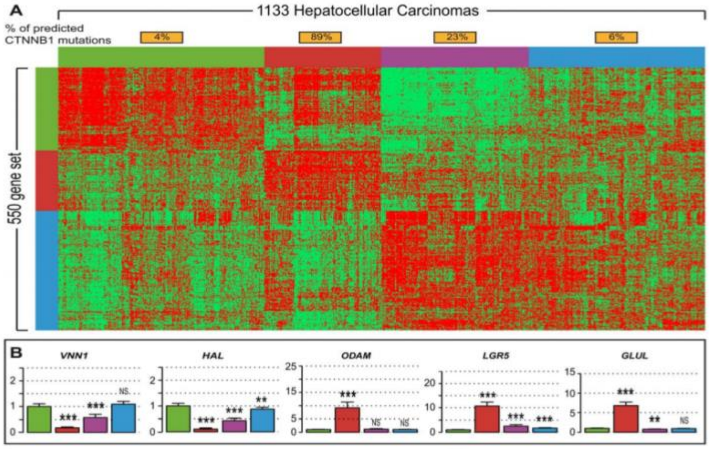


Figure 3. Liver cancer dataset [34]

Chapter 4

Feature Selection Techniques and Optimization Methods

4.1 Feature Selection Techniques

Feature selection also called as variable selection or attribute selection. It is the automated collection of attributes in your data (such as columns in tabular data) that are most appropriate to the predictive modeling problem currently working on.

Feature selection is different from dimensionality reduction. Aim of both methods is to reduce the number of attributes in the dataset, but a dimensionality reducing method does same thing by creating various combinations of attributes, whereas feature selection methods include and exclude characteristics present in the data without changing them [35].

Feature selection methods can be used to distinguish and remove unneeded, unnecessary, and excess attributes from data that do not offer the accuracy of a predictive model or may reduce the accuracy of the model. There are three over-all classes of feature selection algorithms: filter methods, wrapper methods, and embedded methods.

In machine learning, feature Selection is also known as selection of attributes or variable subsets. It is the process of choosing a subset of relevant features (factors, indicators) for use in model development. There are three different approaches to feature selection: wrapper strategy, filter method and embedded approach. The methods are briefly described below.

4.1.1 Wrapper Strategy

This strategy gives a high computing complexity. Wrapper methods consider the choice of a set of features like a search problem, where different combinations are developed, estimated, and compared to other combinations. It utilizes a learning algorithm to assess the precision of classification using the chosen characteristics. Wrapper techniques can offer specific classifiers elevated classification precision. The search process may be systematic such as a best-first search, it may be stochastic such as a random hill-climbing algorithm, or it may use heuristics. Examples of these methods are recursive feature elimination, greedy algorithms.

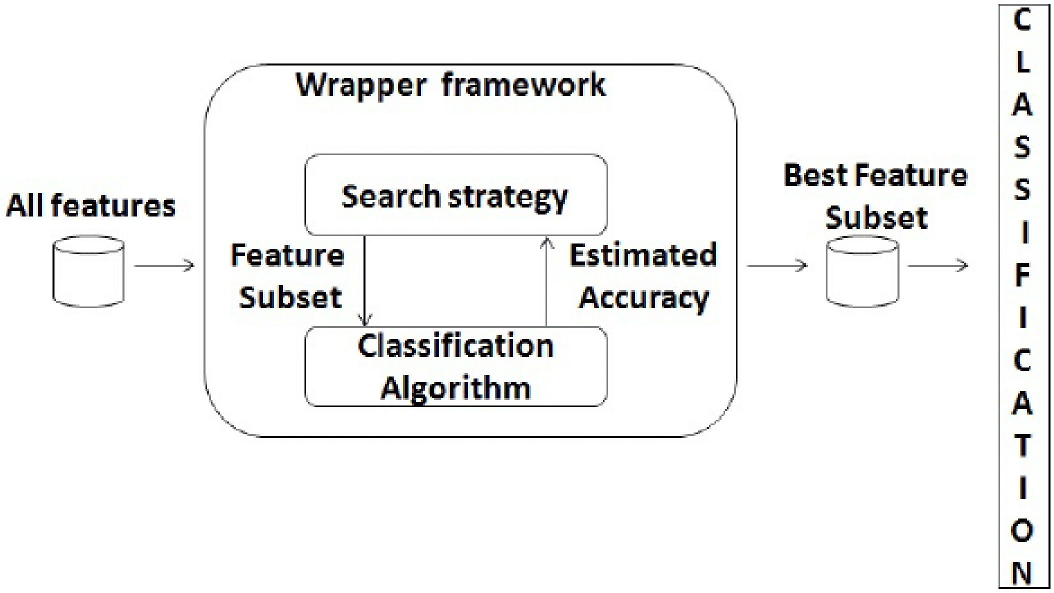


Figure 4. Wrapper Method Framework [36].

4.1.2 Filter Method

This method selects a subset of characteristics without any learning algorithm being used. Filter feature selection methods apply a mathematical measure to assign a scoring to each feature. This technique is used by higher-dimensional data sets and is comparatively quicker than methods based on the wrapper. The elements are ranked by the score and either selected to be kept or removed from the dataset. The plans are often univariate and consider the feature independently, or about the dependent variable. There are so many filter methods are available such as PCA, CFS, Gain Ratio, and relief methods. Examples of some filter methods involve the Chi-squared test, information gain, and correlation coefficient scores.

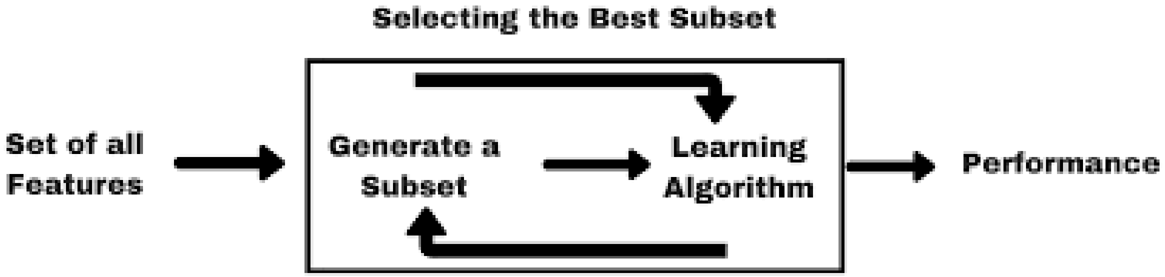


Figure 5. Flow chart of Filter Method [36]

4.1.3 Embedded Approach

The applied learning algorithms determine this approach's specificity and excellent characteristics during the information set training phase. Embedded methods learn which features finest contribute to the accuracy of the model while the model is being created. This method, we have added penalties against unpredictability to decrease the level of overfitting or fluctuation of a model by including more bias. The examples of this method are L1 (LASSO), decision tree. The most common type of embedded feature selection method is regularization methods. Regularization methods are also knowns as penalization methods that offer extra constraints into the optimization of a predictive algorithm (such as a regression algorithm) that bias the model toward more inexpensive complexity (fewer coefficients). Examples of regularization algorithms are the LASSO, Elastic Net, and Ridge Regression.

4.2 Optimization Methods

4.2.1 Particle Swarm Optimization

Microarray data are often extremely asymmetric in dimensional, highly redundant and noisy. Most of the genes are believed to be uninformative concerning studied classes. Here, we have used the feature selection approach for the classification of high dimensional cancer microarray data, which uses optimization technique as Particle swarm Optimization (PSO). The proposed system is divided into two stages. In the initial stage, the data set is clustered using k-means clustering. The SNR score is then used to rank each gene in every cluster. The highest scored genes from each group are gathered, and a new feature subset is generated. In the second stage, the new feature subset is used as input to the PSO, and an optimized feature subset is being produced. Support vector machine (SVM), k-nearest neighbor (k-NN) and Probabilistic Neural Network (PNN) are used as evaluators and leave one out cross-validation approach is used for validation. Particle swarm optimization (PSO) is a primarily population-based stochastic optimization technique established by Eberhart and Kennedy in 1995, inspired by social behavior of bird flocking or fish schooling [37]. PSO is almost same as evolutionary computation techniques such as Genetic Algorithms (GA). The system is prepared with a population of random solutions and searches for optima by updating generations. However, GA has evolution operators but PSO has no evolution operators such as crossover and mutation. In PSO there are the potential solutions and that is called particles moreover, they fly through the problem space by following the current optimum particles. Each particle keeps track of its coordinates in the problem space which is associated with the best solution (fitness) it has achieved so far. The fitness value is also stored for each particle. This value is called pbest. Additional best value that is tracked by the particle swarm optimizer is the best value, is obtained by particle so far in the neighbors of the particle. This location is called *lbest*. At the end, particle takes all the population as its topological neighbors, the best value is a global best and is called *gbest*. The particle swarm optimization theory includes, time step, changing the velocity of (accelerating) each particle on the way to its p*best* and *lbest* locations. Figure 6. shows the flow chart of the PSO process.

In PSO, the position of each partner of the mass in the global search space is updated by two mathematical equations [38] as shown in equations (1) and (2). In equation (1), c_1, c_2 are coefficients and r_1, r_2 are random numbers, each having a position xi on R^n in the search-space and a velocity vi on R^n.

$$v_i^{k+1} = v_i^k + c_1 r_1 \left(pbest_i^k - x_i^k \right) + \ c_2 r_2 \left(g_{best} - x_i^k \right) \ (1)$$

$$x_i^{k+1} = x_i^k + v_i^{k+1} \qquad (2)$$

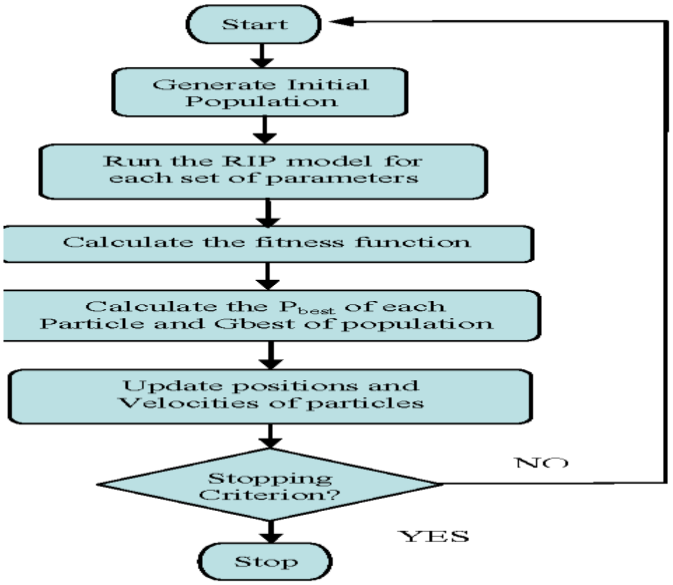

Start
Generate Initial Population
Run the RIP model for each set of parameters
Calculate the fitness function
Calculate the P_best of each Particle and Gbest of population
Update positions and Velocities of particles
Stopping Criterion?
NO
YES
Stop

Figure 6. Flow chart of PSO Controller design procedure [39]

4.2.2 Binary Grey Wolf Optimization Algorithm

The BGWO algorithm consider the searching, seeking behavior and the social hierarchy of the grey wolves. Due to more limited randomness and changing numbers of individuals assigned in global and local searching methods, the BGWO algorithm is easier to use and concentrates more rapidly. It has been determined to be more efficient than the PSO algorithm, and other bionic algorithms and added attention had been paid to its applications due to its more reliable performance. Efforts have been made in feature and band selection, automatic control, power dispatching, parameter estimation, shop scheduling, and multi-objective optimization. However, the standard BGWO algorithm was expressed with similar importance of the grey wolves' positions, which is not consistent rigidly with their social hierarchy. Recent improvements of the BGWO algorithms such as the binary BGWO algorithm, multi-objective BGWO algorithm, and mix with others, together with their applications, keep it remaining constant. If the searching and hunting areas of the grey wolves are also admitted to the social hierarchy, the BGWO algorithm will be likely developed [40].

Mirjalili [41] created the optimization algorithm for the searching and hunting process of grey wolves. In the numerical model, the fittest solution is called the alpha (α), the second-best is beta (β), and consequently, the third-best has named the delta (δ). The rest of the competitor solutions are all considered to be omegas (ω). The encircling behavior of BGWO can be calculated as follows:

$$\vec{X}(t+1) = \vec{X}_p(t) - \vec{A} \cdot \vec{D} \qquad (3)$$

where $\vec{A}$ is a coefficient vector, $\vec{X}_p$ is the prey's positions vector, X is the position of wolves in a d-dimensional space where d is the number of variables, t is the number of iterations, and $\vec{D}$ is defined as follows:

$$\vec{D} = \left|\vec{C} \cdot \vec{X}_p(t) - \vec{X}(t)\right| \qquad (4)$$

where $\vec{C}$ is a coefficient vector and $\vec{A}\,, \vec{C}$ are given as below:

$$\vec{A} = 2\vec{a} \cdot \vec{r}_1 - \vec{a} \qquad (5)$$

$$\vec{C} = 2 \cdot \vec{r}_2 \qquad (6)$$

where, $\vec{r}_1$ and $\vec{r}_2$ are random vectors in [0, 1]. $\vec{a}$ a set vector linearly decreases from 2 to 0 over iterations.

In the hunting process of grey wolves, alpha is considered the optimal applicant for the solution, beta and delta expected to be knowledgeable about the prey's possible position. Therefore, three best solutions that have been found until a certain iteration are kept and forces others (e.g. omega) to change their positions in the decision space reliable with the best place. The updating positions mechanism can be calculated as follows:

$$\vec{X}\,(t+1) = \frac{\overrightarrow{x_1 + x_2 + x_3}}{3} \qquad (7)$$

where $x_1, x_{2,} x_3$ are defined and calculated as follows:

$$\overrightarrow{x_1} = \left|\overrightarrow{X_\alpha} - A_1 \cdot \overrightarrow{D_\alpha}\right|$$

$$\overrightarrow{x_2} = \left|\overrightarrow{X_\beta} - A_2 \cdot \overrightarrow{D_\beta}\right| \qquad (8)$$

$$\overrightarrow{x_3} = \left|\overrightarrow{X_\delta} - A_3 \cdot \overrightarrow{D_\delta}\right|$$

where $\overrightarrow{x_1}$, $\overrightarrow{x_2}$ and $\overrightarrow{x_3}$ are the three best wolves (solutions) in the swarm at a given iteration t. A_1 , A_2 and A_3 are calculated as in Eq. (5). $\overrightarrow{D_\alpha}$, $\overrightarrow{D_\beta}$ and $\overrightarrow{D_\delta}$ are calculated as in Eq. 9.

$$\overrightarrow{D_\alpha} = \left|\overrightarrow{c_1}.\overrightarrow{X_\alpha} - w * \overrightarrow{X}\right|$$

$$\overrightarrow{D_\beta} = \left|\overrightarrow{c_2}.\overrightarrow{X_\beta} - w * \overrightarrow{X}\right| \qquad (9)$$

$$\overrightarrow{D_\delta} = \left|\overrightarrow{c_3}.\overrightarrow{X_\delta} - w * \overrightarrow{X}\right|$$

where $\overrightarrow{c_1}$, $\overrightarrow{c_2}$, $\overrightarrow{c_3}$ are calculated based on equation (6) and w is a constant weight.

In BGWO one of the main components to tune exploration and exploitation is the vector $\vec{a}$. In this algorithm, it is advised to decrease the vector for each of dimension linearly proportional to the number of iterations from 2 to 0. The equation to update it is as follows:

$$\vec{a} = 2 - t \cdot \frac{2}{\underset{i}{max}\ ter} \qquad (10)$$

where t is the iteration number, *ter* is the optimization total number of iterations. It should be noted that all equation used in this section are obtained from [41].

4.2.3 Hybrid Approach (GWOPSO)

Many researchers have presented several hybridization variants for heuristic variants [42]. As per researcher Talbi two variants can be hybridized at a low level or high level with a relay or coevolutionary techniques as heterogeneous or homogeneous. GWOPSO's main idea is to increase the algorithm's capability to exploit PSO with the ability to explore GWO to achieve both optimizer strength. We hybridize Particle Swarm Optimization with Grey Wolf Optimizer algorithm by using low-level coevolutionary mixed hybrid, and from that, we can say that the hybridization is low level because we merge the functionalities of both variants, we cannot use both variants one after another. In general, they execute in parallel, or we can say at the same time. It is mixed because there are two distinct variants that are involved in generating final solutions for the problems.

In hybrid GWOPSO, the first three agents' position is updated in the search space by the proposed mathematical equations (11). As an alternative of using usual mathematical equations, we control the exploration and exploitation of the grey wolf in the search space by inertia constant w.

$$\overrightarrow{D_\alpha} = \left|\overrightarrow{c_1}.\overrightarrow{X_\alpha} - w * \overrightarrow{X}\right|$$
$$\overrightarrow{D_\beta} = \left|\overrightarrow{c_2}.\overrightarrow{X_\beta} - w * \overrightarrow{X}\right| \qquad (11)$$
$$\overrightarrow{D_\delta} = \left|\overrightarrow{c_3}.\overrightarrow{X_\delta} - w * \overrightarrow{X}\right|$$

In order to combine the PSO and GWO the equation of updated velocity is as follow:

$$v_i^{k+1} = w * v_i^k + c_1 r_1\left(x_1 - x_i^k\right) + c_2 r_2\left(x_2 - x_i^k\right) + c_3 r_3\left(x_3 - x_i^k\right) \qquad (12)$$
$$x_i^{k+1} = x_i^k + v_i^{k+1} \qquad (13)$$

4.2.4 Feature Selection using Hybrid Binary Grey Wolf Optimization and Particle Swarm Optimization

A binary version of the hybrid Grey Wolf Optimization (GWO) and Particle Swarm Optimization (PSO) is used to solve feature selection problems in this research. The original PSOGWO is a new hybrid optimization algorithm that gives benefits from the GWO and PSO. Apart from the higher performance, the original hybrid approach is appropriate for problems with a continuous search space. Feature selection, however, is a binary problem. Therefore, a binary version of a hybrid PSOGWO called BGWOPSO is proposed to find the best feature subset. For the best solution, wrapper-based method K-Nearest Neighbors (KNN) classifier with Euclidean separation matric is utilized. For performance evaluation of the binary algorithm, three cancer datasets from UCI repository and two microarray datasets from NCBI and that are employed. The results indicate that BGWOPSO significantly outperformed the hybrid grey wolf optimization particle swarm optimization. Binary grey wolf (BGWO)and hybrid binary grey wolf particle swarm optimization (BGWOPSO) using several performance measures including accuracy, selecting the best optimal features.

According to [13], the updating mechanism of wolves is a function of three vectors positions, namely $x_1, x_{2,} x_3$, which promotes every wolf to the first three best solutions. For the agents to work in a binary space, the position updating Eq. (7) can be modified into the equation (14).

$$x_d^{t+1} = \begin{cases} 1, \ if \ simoid \ \left(\frac{x_1 + x_2 x_3}{3}\right) \geq rand \\ 0, \qquad\qquad\qquad otherwise \end{cases} \quad (14)$$

where x_d^{t+1} is the binary updated position at iteration *t* in dimension *d*, *rand* is a random number drawn from distribution ∈ [0, 1] and sigmoid(x) is denoted as following [13]:

$$\text{sigmoid (x)} = \frac{1}{1+e^{-10(x-0.5)}} \quad (15)$$

$x_1, x_{2,} x_3$ in (8) are updated and calculated using the following equations:

$$x_1^d = \begin{cases} 1 \ if \ \left(x_\alpha^d + \ bstep_\alpha^d\right) \ \geq 1 \\ 0 \qquad\qquad othwewise \end{cases}$$

$$x_2^d = \begin{cases} 1 \; if \left(x_\beta^d + \; bstep_\beta^d\right) \; \geq 1 \\ 0 \qquad\qquad othwewise \end{cases} \qquad (16)$$

$$x_3^d = \begin{cases} 1 \; if \left(x_\delta^d + \; bstep_\delta^d\right) \; \geq 1 \\ 0 \qquad\qquad othwewise \end{cases}$$

where, $x_{\alpha,\beta,\delta}^d$ the position’s vector of the alpha, beta, delta wolves in *d* dimension, and $bstep_{\alpha,\beta,\delta}^d$ is a binary step in *d* dimension, which can be formulated as follow:

$$bstep\,_{\alpha,\beta,\delta}^{\;d} \;\; = \begin{cases} 1 \; if \left(cstep_{\alpha,\beta,\delta}^d\right) \; \geq rand \\ 0 \qquad\qquad othwewise \end{cases} \quad (17)$$

where *rand* a random value derived from a uniform distribution ∈[0,1], *d* indicates dimension, and $cstep_{\alpha,\beta,\delta}^d$ is *d*’s continuous value. This component is calculated using the following equation [13]:

$$cstep_{\alpha,\beta,\delta}^d = \frac{1}{1+e^{-10\left(A_1^d \, D_{\alpha,\beta,\delta}^d - 0.5\right)}} \qquad (18)$$

In BGWOPSO, and based on the best three solutions positions updated in (16), the exploration and exploitation are controlled by an inertia constant weight w mathematically modeled as follows:

$$\overrightarrow{D_\alpha} = \left|\overrightarrow{c_1}.\overrightarrow{X_\alpha} - w * \overrightarrow{X}\right|$$

$$\overrightarrow{D_\beta} = \left|\overrightarrow{c_2}.\overrightarrow{X_\beta} - w * \overrightarrow{X}\right| \qquad (19)$$

$$\overrightarrow{D_\delta} = \left|\overrightarrow{c_3}.\overrightarrow{X_\delta} - w * \overrightarrow{X}\right|$$

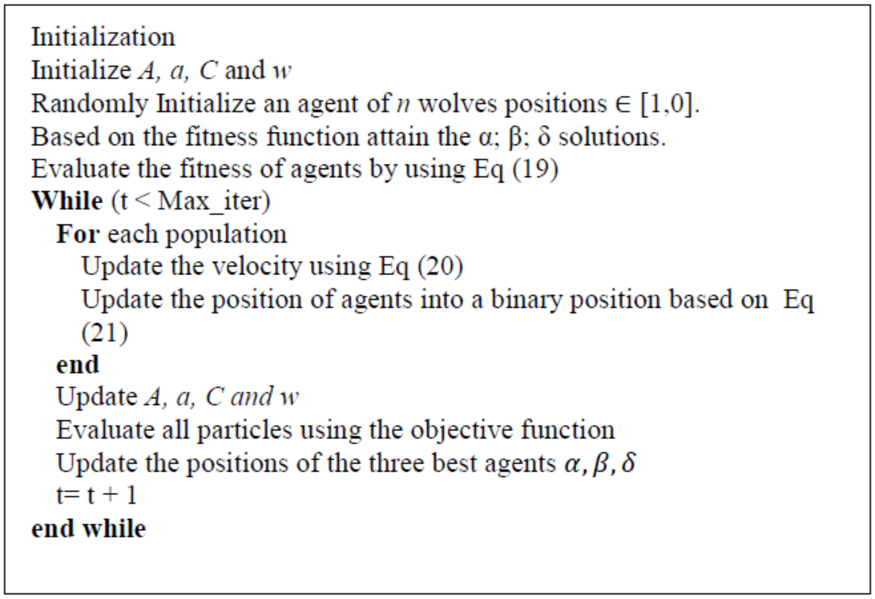
Accordingly, the velocity and positions have been updated as follows [38]:

$$v_i^{k+1} = w * v_i^k + c_1 r_1\left(x_1 - x_i^k\right) + \; c_2 r_2\left(x_2 - x_i^k\right) + c_3 r_3\left(x_3 - x_i^k\right) \qquad (20)$$

Note that in (20) the best three solutions $x_1, x_{2,} x_3$ are updated according to (16).

$$x_i^{k+1} = x_d^{t+1} + v_i^{k+1} \quad (21)$$

Where x_d^{t+1} and v_i^{k+1} are calculated based on Eq (14) and Eq (20) respectively.



```
Initialization
Initialize $A$, $a$, $C$ and $w$
Randomly Initialize an agent of $n$ wolves positions ∈ [1,0].
Based on the fitness function attain the α; β; δ solutions.
Evaluate the fitness of agents by using Eq (19)
While (t < Max_iter)
  For each population
    Update the velocity using Eq (20)
    Update the position of agents into a binary position based on  Eq
    (21)
  end
  Update $A$, $a$, $C$ and $w$
  Evaluate all particles using the objective function
  Update the positions of the three best agents $\alpha, \beta, \delta$
  t= t + 1
end while
```

Figure 7. Pseudocode of the proposed binary (BGWOPSO) [43]

The solution in this study is illustrated in a one-dimensional vector. The length of this vector is equal to the number of features. In this binary vector, 0 and 1 have the following meaning:

- 0: Feature not selected
- 1: Feature is selected

The feature selection problem is bi objective by nature. One goal is to find the minimum number of features, and the other is to maximization the classification accuracy. To consider both, the following equations used as a fitness function [13]:

$$fitness = \alpha \rho_R(D) + \beta \frac{|S|}{|T|} \quad (22)$$

Where α = [0,1], and β=$(1-\alpha)$ they are parameters adapted from [13], $\rho_R(D)$ indicates the error rate of the classifier. Moreover, $|S|$ is the selected subset of features and $|T|$ are the whole features in dataset. It should be noted that the mathematical equations used in this section are obtained from [13] [38].

Chapter 5

Machine Learning Techniques

5.1 Classification Methods

Recently machine learning algorithm is extensively available for various purpose. However, the most effective method of machine learning is a classification method. Classification methods are the heart of machine learning. As the growth of machine learning classification is getting more attention. In this study we deal with a classification problem and that majorly focuses on dividing the samples of two microarray datasets into two main categories. A classification method uses a single set of parameters to characterize each of its objects. These features are much more relevant to the data studied. Here, we are talking about techniques of supervised learning, where we know the classes into which the items are to be ordered. We likewise have a set of objects with known types.

5.1.1 Support Vector Machine

SVM has been recently established in the frame of statistical learning theory and has been magnificently applied to several applications. A support vector machine (SVM) is a machine learning (ML) algorithm that analyzes data for classification and regression analysis. The support vector machine is also known to be a supervised learning method that sorts the data into one of two categories. SVM outputs associate of the sorted data with the margins between the two as far apart as possible [44] .

"A Support Vector Machine (SVM) is a discriminative classifier formally defined by a separating hyperplane." In other terms, given labeled training data (supervised learning), the algorithm produces an optimal hyperplane which categorizes new examples". [45]

The original algorithm was first invented by Vladimir N. Vapnik and Alexey Ya. Chervonenkis in 1963. In 1992, Bernhard E. Boser, Isabelle M. Guyon, and Vladimir N. Vapnik proposed a way to build nonlinear classifiers by applying the kernel trick to the maximum-margin hyperplane. The modern standard soft margin was proposed by Corinna Cortes and Vapnik in 1993 and published in 1995 [46].

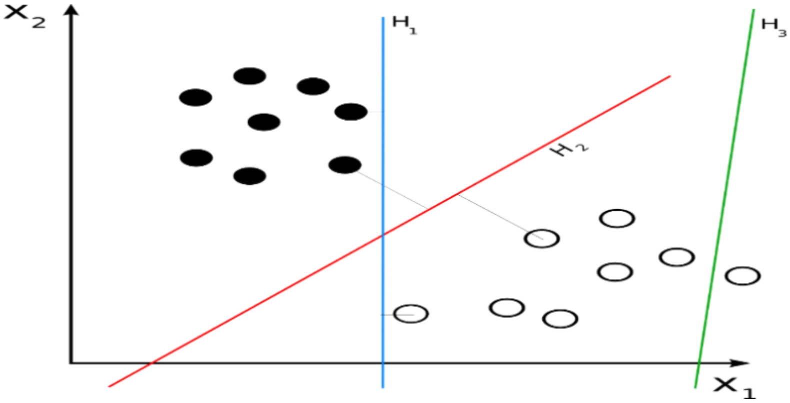


Figure 8. Support Vector Machine Classification [47]

Let's take the above example, suppose we want to find a separating straight line for a set of 2-D points. In the figure, several lines exist, and every line gives solutions to the problem. So, how we can identify the ideal solution for this problem?

A line is not good if it passes too close to the points because it will contain a certain amount of noise and not give us a precise solution. Therefore, the main aim is to find the line which is far away from all points.

Then, the process of the SVM algorithm is based on finding the hyperplane correctly. Classifies all the data while being farthest away from the data points. Therefore, the optimal separating hyperplane that maximizes the margin, defined as the distance from the hyperplane to the closest data point.

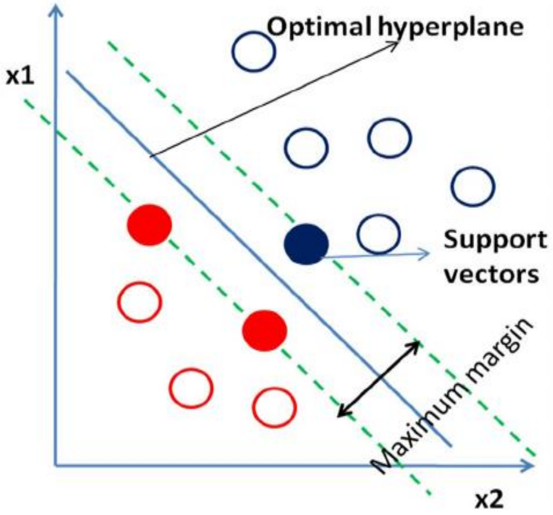


Figure 9. Optimal hyperplane [47]

Because of, 2-D points it is a straightforward task for SVM. However, for high dimensions, the same concepts apply to the job.

Formally a hyperplane can be defined as:

$$f(x) = \beta_0 + \beta^T x, \qquad (23)$$

Where β is identified as the weight vector and β_0 as the bias.

The optimal hyperplane can be represented in an uncountable number of different ways by scaling of β and β_0. SVM will choose the one that satisfied,

$$|\beta_0 + \beta^T x| = 1, \qquad (24)$$

Where x denotes the training examples closest to the hyperplane. Overall, the training examples that are closest to the hyperplane are called support vector. This representation is also recognized as the canonical hyperplane [48].

In the below equation, we have used the result of geometry that gives the distance D between a point x and a hyperplane(β, β_0):

$$D = \frac{|\beta_0 + \beta^T x|}{||\beta||} \quad (25)$$

Mainly, for canonical hyperplane, the numerator is equal to one and the distance to the support vector:

$$\mathrm{D}_{\mathrm{SupportVector}} = \mathrm{D} = \frac{|\beta_0 + \beta_T x|}{||\beta||} = \frac{1}{||\beta||} \quad (26)$$

Previously, presented margin, we denoted the margin as M, which is twice the distance to the closest examples:

$$M = \frac{2}{||\beta||} \quad (27)$$

At last, the problem of maximizing M is equivalent to the question of minimizing a function $L(\beta)$ subject to some constraints. The constraints model the requirement for the hyperplane to classify all the training examples x_i correctly [58]. Formally,

$$\min_{\beta, \beta_0} L(\beta) = \frac{1}{2} ||\beta||^2 \quad (28)$$

$$\text{subject to:} \qquad y_i(\beta^T x_i + \beta_0) \geq 1 \ \forall I, \quad (29)$$

where y_i denotes each of the labels of the training examples.

5.1.2 Artificial Neural Network

Inspired by the cultured functionality of human brains where hundreds of billions of interconnected neurons process information in parallel Artificial Neural Networks are comparatively electronic models based on the neural structure of the brain. The brain essentially learns from experience. It is natural proof that some problems are beyond the scope of current computers and those are indeed answerable by the small energy-efficient package. Within humans, there are many modifications on this primary type of neurons. Also, all-natural neurons have the same four necessary components. These components are recognized by their botanical names - dendrites, soma, axon, and synapses. Neural systems are in some cases portrayed regarding their understanding, including what number of layers they have among information and yield, or the model's supposed shrouded layers. This is the reason the term neural system is utilized synonymously with profound learning. They can likewise be portrayed by the quantity of concealed hubs the model has or as far as what number of sources of info and yields every hub has. Minor departure from the great neural system configuration permits dissimilar types of forward and in reverse proliferation of data among levels.

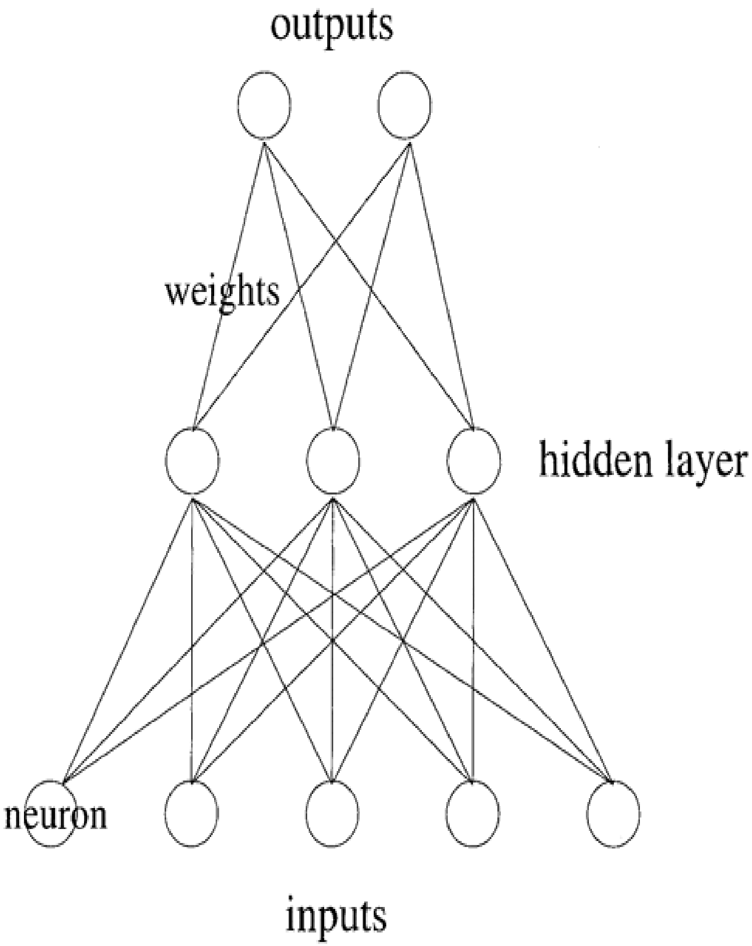


Figure 10. Artificial Neural Network Architecture [49]

An artificial neural network (ANN) consists of an input layer of neurons which is also called nodes or units. It has one or two in some cases, even three hidden layers of neurons are present, and it has a final layer of output neurons. See the above, figure for a typical architecture, where lines connecting to the neurons are also shown. Each connection is linked with a numeric number called weight. The output, h_i, of neuron i in the hidden layer is:

$$h_i = \sigma(\sum_{j=1}^{N} V_{ij} x_j + T_i^{hid}), \quad (30)$$

where $\sigma()$ is called activation function, N the number of input neurons, V_{ij} the weights, x_j inputs to the input neurons, and T_i^{hid} the threshold terms of the hidden neurons.

The frequently utilized type of ANN is the Multilayer Perceptron. ANN offers a compelling and particularly general structure for speaking to non-linear mapping from a few information variables to a few yield variables.

Perceptron is the simplest and oldest model of Neuron. Also known as a single-layered neural network. In, Feedforward neural networks all nodes are fully connected, and activation flows from input layer to output, without back loops, and there is one layer between input Moreover, In Recurrent Neural Networks involves several type of Recurrent cells and there are many variations like passing the state to input nodes, variable delays, iteration and so on. The commonly utilized type of ANN is the Multilayer Perceptron. ANN offers a compelling and exceptionally general structure for speaking to non-linear mapping from a few information variables to a few yield variables [50].

Note: In our research, ANN has 1 input layer 1 hidden layer and 1 output layer. Hidden layer has 10 neurons.

5.1.3 Logistic Regression

“Logistic regression is a classification algorithm used to allocate observations to a discrete collection of classes." Unlike linear regression, which outputs constant number values, logistic regression modifies its output using the logistic sigmoid function to deliver a probability value which can then be documented with more than two discrete classes.

When it comes to classification, we determine the probability of an observation being part of a particular class or not. Hence, we wish to express the likelihood with a value between the Highest and Lowest, where the highest is characterized as 1 and the lowest stand for 0. A possibility close to 1 means the observation is highly likely to be of that category. To produce values between 0 and 1, we convey the probability using this equation:

$$p(X) = \frac{\exp(\beta_0 + \beta_1 X)}{1 + \exp(\beta_0 + \beta_1 X)} \tag{31}$$

The equation above is identified as the sigmoid function.

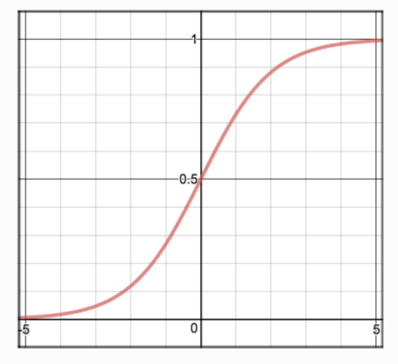


Figure 11. Sigmoid Function [51]

Plot this equation, and you will see that this equation always results in an S-shaped curve bound between 0 and 1.

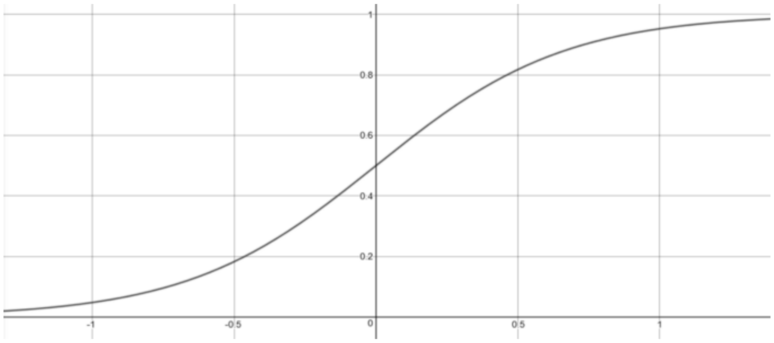


Figure 12. Logistic regression curve [51]

After iteration to the equation above, you find that:

$$\frac{p(X)}{1-p(X)} = \exp(\beta_0 + \beta_1 X) \tag{32}$$

Take the log on both sides:

$$\log\left(\frac{p(X)}{1-p(X)}\right) = \beta_0 + \beta_1 X \tag{33}$$

The equation above is recognized as the logit. It is linear in X. Here, if the coefficients are positive, then an expansion in X will result in a greater probability.

Determining the coefficient as in linear regression, we need a method to determine the ratios. For that, we maximize the likelihood function:

$$l(\beta_0, \beta_1) = \prod_{i:y_i=1} p(X_i) \prod_{i':y_i'=0} \left(1 - p(X_i{}')\right) \tag{34}$$

In here, we want factors such that the foretold probability (denoted with an apostrophe in the equation above) is as near as likely to the observed state. As in, linear regression, we use the p-value to conclude if the null hypothesis is rejected or not.

The Z-statistic is also commonly used, and A huge absolute Z-statistic means that the null hypothesis is discarded. Remember that the null hypothesis states: there is no correlation between the features and the target. Logistic regression can be enlarged to hold more than one predictor:

$$\log\left(\frac{p(X)}{1-p(X)}\right) = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p \tag{35}$$

Note, that the use of multiple logistic regression might give significantly better results as it can take into correlations among predictors, an event known as confounding. Also, infrequently will only one predictor be enough to make an absolute model for prediction.

5.1.4 K-Nearest Neighbor

K nearest neighbors is a modest algorithm that stores all available cases and classifies new cases based on a similarity measure such as distance functions. At the beginning of the 1970s as a non-parametric technique, KNN has been used in statistical estimation and pattern recognition [52].

In this method, case is classified by a majority vote of its neighbors, once the case is assigned to the class and most common amongst its K nearest neighbors measured by a distance function (distance function describes below). Moreover, if $K = 1$, then the case is attached to the type of its nearest neighbor.

There are four most popular distance functions: Euclidean Distance, Manhattan Distance, Minkowski Distance, and Hamming Distance [53].

$$\text{Euclidean: } \sqrt{\sum_{i=1}^{k}(x_i - y_i)^2} \qquad (36)$$

$$\text{Manhattan: } \sum_{i=1}^{k}|x_i - y_i| \quad (37)$$

$$\text{Minkowski: } \left(\sum_{i=1}^{k}(|x_i - y_i|)^q\right)^{\frac{1}{q}} \quad (38)$$

$$\text{Hamming: } D = \sum_{i=1}^{k}|x_i - y_i|, \text{ where } x = y \Rightarrow D = 0; x \neq y \Rightarrow D = 1 \quad (39)$$

Example: The input is a vector of some ‘k' closest training examples that exist in the hyperspace and based on the value of these k nearest neighbors, the classification for the object is done based on a majority voting about the classes of its neighbors. In the Figure, it is clear that in (a) k=1, the object is classified the same as its nearest neighbor (red square) while in (b) k=4, classification is done based on looking at the nearest four points.

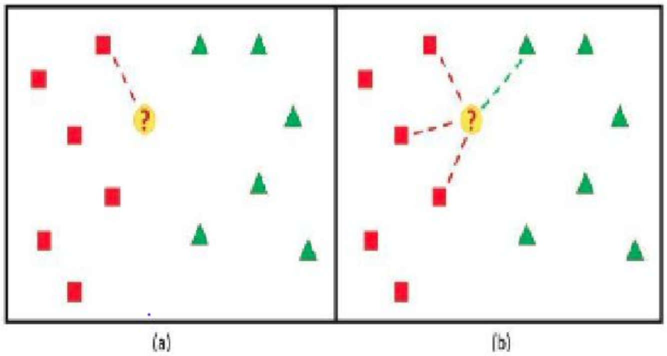


Figure 13. Classes assigned using K-NN classifier for (a) K=1 and (b) K=4 [54]

Note: We have used K = 3 - this means when a test data is input, we check the 3 closest training data points around it and take a vote of which class they belong to. The test data point is assigned to the class with the majority.

5.1.5 Random Forest

Random forest, like its name suggests, consists of a vast number of unique decision trees that work as an ensemble. Each particular tree in the random forest spits out a class forecast, and the class with the highest vote becomes our model's prediction as in figure 14.

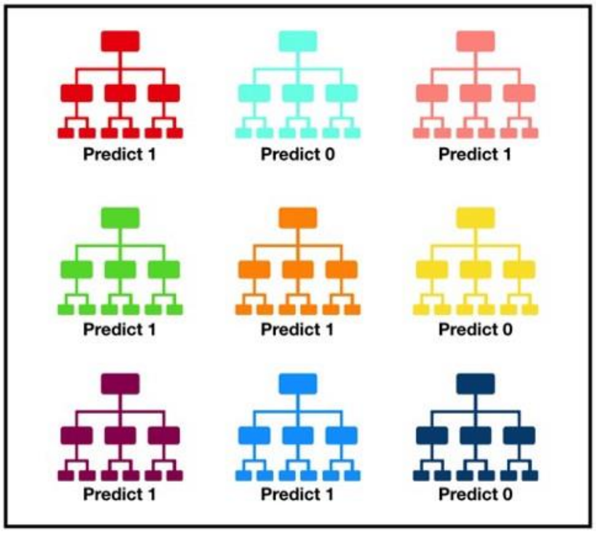


Figure 14. Visualization of a Random Forest Model Making a Prediction [55]

The first-ever algorithm for random decision forests was founded by Tin Kam Ho [56]using the random subspace method [57] which, in Ho's formulation, is a way to reach the "stochastic discrimination" approach to the classification suggested by Eugene Kleinberg. Ensembles are a divide-and-conquer strategy used to improve performance. The motto of the ensemble methods is that a group of weak learners can come together to form a powerful learner.

The random forest initiates with a standard machine learning technique called a "decision tree," which, in ensemble terms, corresponds to our weak learner. In a decision tree, an input is entered at the top of the tree and as it traverses down the tree, the data gets bucketed into smaller and smaller sets.

In this example, see Figure 15, the tree teaches us, based upon weather situations, whether to play ball. For example, it is sunny, and the humidity is less than or somewhere equal to 70, then it's probably all right to play.

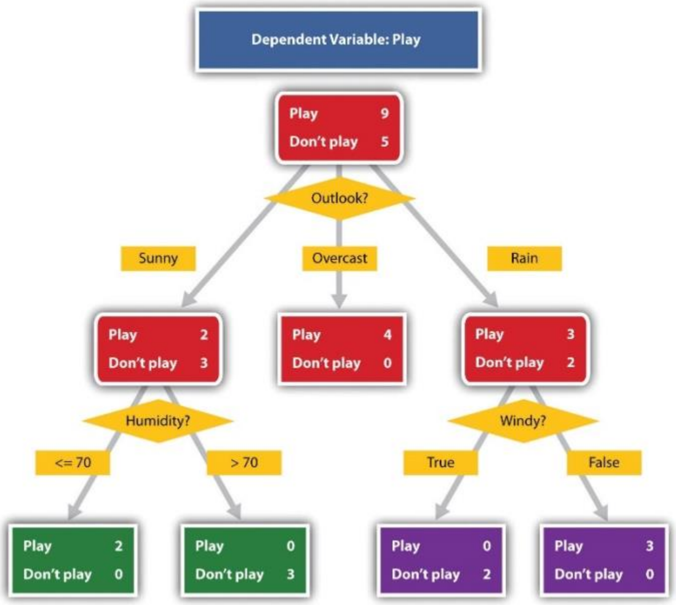
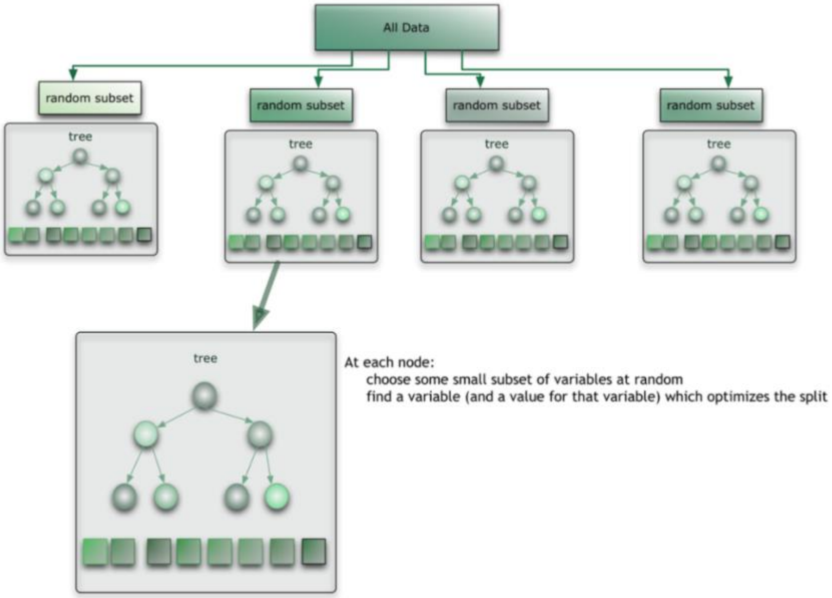


Figure 15. Example of a Decision Tree *[58]*

The random forest takes this idea to the next level by combining trees with the sense of an ensemble (See figure 15). Ergo, in ensemble terms, the trees are weak learners, and the random forest is a keen learner.



At each node:
choose some small subset of variables at random
find a variable (and a value for that variable) which optimizes the split

Figure 16. Relationship between Random Forest and Random Tree [58]

Here, system is trained for some number of trees T. Sample N cases at random with replacement to produce a subset of the data. The subset should be about 70% of the total set. At each node, m predictor variables are selected at random from all the predictor variables. The predictor variable that delivers the best split, according to some objective function, is used to do a binary split on that node. At the next node, it will choose another m variables at random from all predictor variables and repeat the same process. Based on the value of m, there are three slightly different systems and there are different method for defining values of m:

Random splitter selection: m =1

Bierman's bagger: m = total number of predictor variables

Random forest: m << number of predictor variables. Brieman suggests three possible values for m: $\frac{1}{2}\sqrt{m}, \sqrt{m}, 2\sqrt{m}$.

When a new input is entered into the system, it is run down all of the trees [59]. The result may either be an average or we can say that weighted average of all of the terminal nodes which are already reached, or in the case of categorical variables.

Random forest runtimes are generally fast, and they are capable to deal with unbalanced and missing data as well. Random Forest may overfit data sets when used for regression as it cannot predict outside the range in the training data.

5.2 Tools

Data processing, classification, as well as feature selection, is done using MATLAB.

5.2.1 MATLAB

MATLAB is a huge performance language for scientific computing. It combines calculation, visualization, and programming in an easy-to-use situation where and addresses the difficulties and resolutions in familiar mathematical representation. Typical uses include Math and computation [60] .

- Algorithm construction
- Modeling, simulation, and prototyping
- Data analysis, exploration, and visualization
- Scientific and engineering graphics
- Application advancement, including Graphical User Interface construction

MATLAB is an interactive system whose primary data portion is an array that does not need dimensioning. This functionality allows you to answer various technical computing difficulties, notably those with pattern and vector formulations, in a fraction of the moment it would take to write a memorandum in a non-interactive scalar language such as C or Fortran. Given the title, MATLAB stands for matrix laboratory. MATLAB provides easy entrance to matrix software. It is developed by the LINPACK and EISPACK, which mutually define the state of art in software for matrix computation. It has developed over a span of years with information from various users. In university environments, it is the official instructional mechanism for beginning and superior courses in arithmetic, engineering, and science. On the industrial level, MATLAB is the device of choice for high-productivity analysis, development, and analysis. MATLAB highlights a family of particular application resolutions called toolboxes. Particularly relevant to most users of MATLAB, toolboxes allow you to learn and apply technoscientific. Toolkits are extensive collections of MATLAB functions (M-files) that stretch the MATLAB surroundings to solve critical classes of enigmas. Areas in which toolboxes are open to add signal processing, control systems, neural networks, fuzzy logic, wavelets, simulation, and several others.

5.3 10-fold cross-validation

Cross-validation is a method for frequent holdout to improve. Cross-validation is a comprehensive method to do continuous holdout that effectively enhances it by decreasing the estimation variance. We take a training set, and a classifier is created. Then we're looking to assess that classifier's efficiency, and there's a certain level of variance in that assessment because it's all underneath the statistics. We want to maintain the difference as small as feasible in the evaluation. Also, cross-validation prevents overfitting.

Cross-validation is a method to reduce the variance, and a cross-validation version further reduces it. We split it only once with cross-validation, but we split it into, say, ten parts. Then we bring 9 of the elements and use them for training, and we use the last item to test. Then we bring another nine parts with the same separation and use them for practice and experimentation with the hold-out thing. We do the whole process ten occasions, each moment we use a distinct section to test. In other cases, we split the dataset into ten parts, and then we keep each piece in turn for monitoring, training on the remainder, control, and averaging the ten outcomes. That would be a "cross-validation of 10 times."

Divide the dataset into ten components, keep each portion in turn, and evaluate the outcomes. Therefore, each data point in the dataset is used for testing once and for training nine times. That's cross-validation of 10 times.

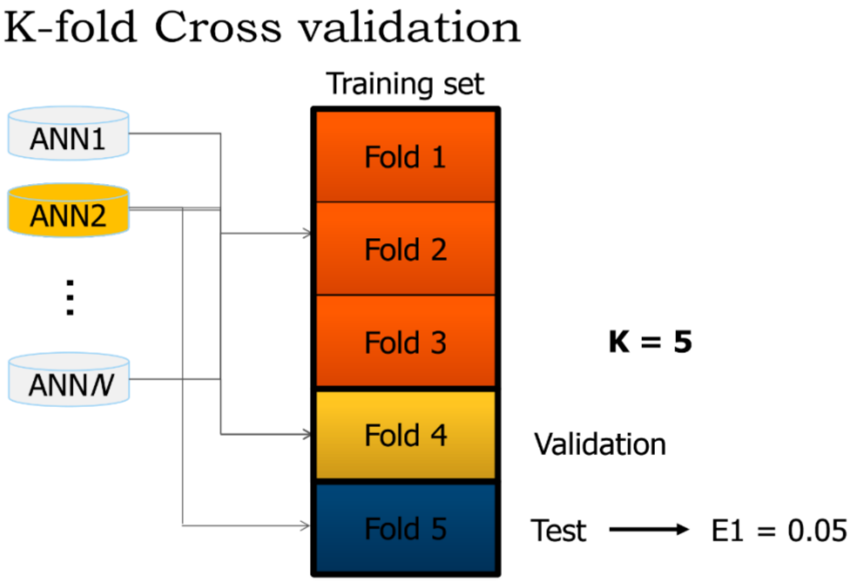


Figure 17. K-Fold Validation in Neural Network[49]

5.4 Ratio Comparison

A comparison of the ratio is defined as the ratio chosen using cross-validation. In our research, we have used ratios as follows:

1. **90:10** where 90% is a training set, and 10% is testing
2. **80:20** where 80% is a training set, and 20% is testing
3. **70:30.** where 70% is a training set, and 30% is testing
4. **60:40.** where 60% is a training set, and 40% is testing

5.5 Methodology

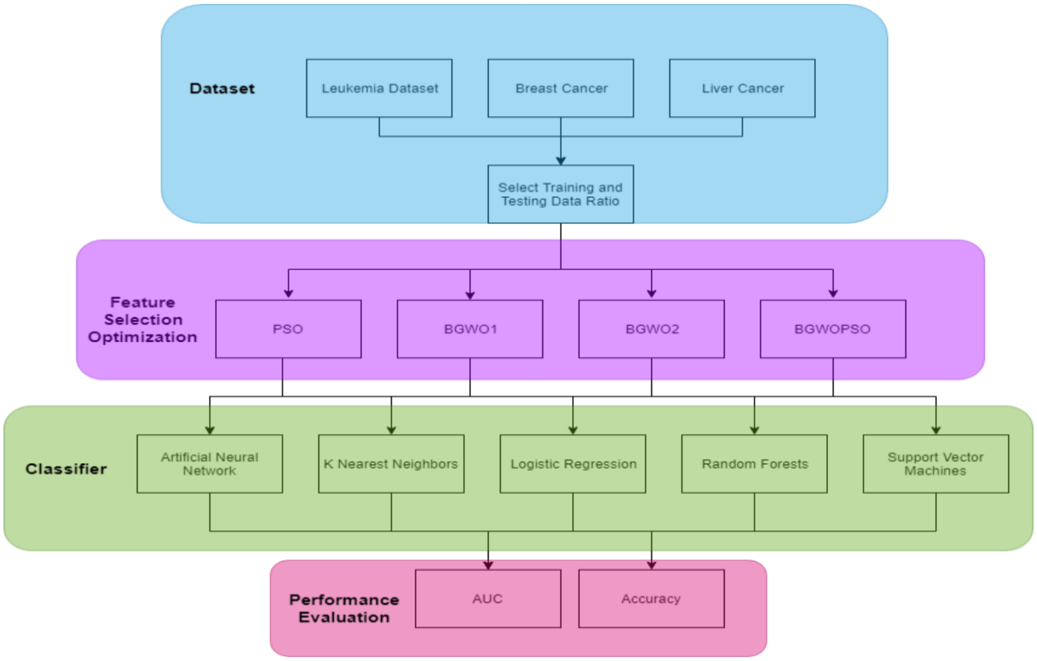


Figure 18: Workflow data analysis

Figure 18. illustrates the workflow of data analysis of cancerous data. Three different types of dataset namely Breast cancer, leukemia cancer and liver cancer dataset are used for data analysis. The next step after downloading the data is feature selection optimization, useful features are extracted from each dataset. These features are useful for better performance evaluation for all three cancer as well as to reduce the complexity of dataset. We have combined all four dataset and created model using MATLAB, which include all extracted feature with most promising number of genes. Moreover, we have used with different training- testing ratio (90:10, 80:20, 70:30, 60:40) with the combination of 5 different classifiers (SVM, KNN, Random forest, Logistic regression and Artificial Neural Network).We have also used 10-fold cross validation in the research.

Chapter 6

Results and Discussion

6.1 Leukemia Dataset Output

To optimize the cancer data, three different types of data were used, and three different kinds of optimization techniques were used to differentiate the features. Furthermore, six distinct classifiers accomplish the outcomes. For different training and testing ratios (90:10, 80:20, 70:30 and 60:40), 10-fold cross validation is performed. The results are highlighted with two different color codes. Orange color Indicates the best accuracy of all five classifier with all ratio. Blue color shows a maximum number of AUC among all datasets with all five classifiers.

Table 1. shows the results leukemia occupied over PSO, BGWO and BGWOPSO optimization algorithms. Analyzed over the ratio of 90-10, 80-20, 70-30 and 60-40, the data is further classified using various classifiers like KNN, SVM, ANN, LR, RF. Using SVM, 0.96789 was the maximum accuracy noted for 80-20 ratio and 1 was the highest AUC gained at 90-10 ratio for BGWO. Using KNN, the highest accuracy and AUC noted were 0.95239 and 0.97686 respectively for 80-20 ratio. For ANN, 0.97342 values were observed at 80-20 ratio. The highest observed accuracy and AUC for Random Forest were 0.97342 and 0.95988 respectively. For the logistic regression, highest accuracy was 0.94556 and AUC was 0.95762, found using BGWOPSO algorithm at 90-10 ratio.

Table 1. Highest Accuracy and AUC values for each classifier with best optimization technique for Leukemia dataset

Ratio	60:40		70:30		80:20		90:10	
Optimizers	**Accuracy**	**AUC**	**Accuracy**	**AUC**	**Accuracy**	**AUC**	**Accuracy**	**AUC**
Classifier: SVM								
PSO	0.86788	0.8856	0.87867	0.8687	0.89768	0.77758	0.93786	0.90345
BGWO	0.84455	0.88563	0.85244	0.96845	0.86675	0.88234	0.87298	1
BGWOPSO	0.95234	0.98567	0.91786	0.92454	0.96789	0.98857	0.90125	0.98983
Classifier: K-Nearest Neighbor								
PSO	0.90344	0.9253	0.9056	0.86254	0.85457	0.75134	0.94324	0.94788
BGWO	0.94432	0.9585	0.88213	0.89588	0.90978	0.96237	0.92453	0.96978
BGWOPSO	0.94454	0.94837	0.90059	0.96577	0.95239	0.97686	0.90245	0.95456
Classifier: Artificial Neural Network								
PSO	0.76589	0.83242	0.89367	0.86877	0.88342	0.98578	0.76211	0.7945
BGWO	0.77657	0.87234	0.89556	0.85675	0.90233	0.88778	0.91676	0.83564
GWOPSO	0.81657	0.95564	0.94582	0.95988	0.97342	0.90975	0.94633	0.88867
Classifier: Random forest								
PSO	0.73490	0.84957	0.81292	0.89089	0.90451	0.81567	0.89455	0.87378
BGWO	0.79899	0.84595	0.92677	0.89756	0.89576	0.80592	0.90485	0.85690
BGWOPSO	0.87990	0.85467	0.90324	0.90876	0.96856	0.95342	0.94889	0.90997
Classifier: Logistic Regression								
PSO	0.85657	0.84990	0.87324	0.9876	0.80324	0.87879	0.82053	0.90768
BGWO	0.82546	0.78456	0.78907	0.87437	0.81790	0.91845	0.75789	0.92878
BGWOPSO	0.86456	0.89768	0.8590	0.93345	0.83456	0.93456	0.94556	0.95762

6.2 Breast Cancer Dataset Output

Table 2. shows the Microarray Brest cancer data results. For different combination of dataset algorithms and testing-training ratios, different classifiers show different data of accuracy and AUC. For SVM classifier, the highest accuracy obtained was 0.95967 and AUC was 0.98443, both for BGWOPSO. For k-nearest neighbor, 0.98678 was the highest accuracy and 0.99677 was the highest AUC, obtained for 70-30 ratio. Using ANN, highest accuracy 0.95357 was obtained for BGWOPSO and highest 0.99638 AUC for BGWOPSO obtained for 90-10 ratio. Using Random forest, 0.94678 was the highest accuracy and 1 was the highest AUC, both found for BGWOPSO at 80-20 ratio. Similarly, Logistic regression had also, highest accuracy and AUC were noted at 90-10 ratio and it was 0.97675 and 1 respectively.

Table 2. Highest Accuracy and AUC values for each classifier with best optimization technique for Microarray Breast cancer dataset

Ratio	**60:40**		**70:30**		**80:20**		**90:10**	
Optimizers	**Accuracy**	**AUC**	**Accuracy**	**AUC**	**Accuracy**	**AUC**	**Accuracy**	**AUC**
Classifier: SVM								
PSO	0.87689	0.81675	0.85689	0.88897	0.81454	0.90679	0.82908	0.87234
BGWO	0.84897	0.85324	0.84789	0.91123	0.82798	0.83456	0.85123	0.92456
BGWOPSO	0.92780	0.96234	0.93567	0.97890	0.93234	0.97896	0.95967	0.98443
Classifier: K-Nearest Neighbor								
PSO	0.87342	0.97589	0.76789	0.87890	0.82123	0.83890	0.76677	0.81234
BGWO	0.87456	0.92334	0.92867	0.92123	0.89789	0.82890	0.90345	0.82678
BGWOPSO	0.91345	0.92676	0.98678	0.99677	0.92678	0.98678	0.90478	0.98678
Classifier: Artificial Neural Network								
PSO	0.82567	0.79657	0.79475	0.88674	0.89456	0.87345	0.83567	0.83356
BGWO	0.79568	0.88535	0.82636	0.90267	0.83656	0.89356	0.92658	0.87673
BGWOPSO	0.83530	0.83099	0.85667	0.98636	0.87367	0.88056	0.95357	0.99638
Classifier: Random forest								
PSO	0.82665	0.89560	0.89543	0.88454	0.85344	0.98734	0.88859	0.89967
BGWO	0.87785	0.99578	0.86768	0.95789	0.87665	0.85678	0.89980	0.81556
BGWOPSO	0.90532	0.90345	0.82467	0.89232	0.94678	1	0.88745	0.94467
Classifier: Logistic Regression								
PSO	0.89896	0.89789	0.88456	0.87456	0.87456	0.87456	0.84563	0.81345

BGWO	0.88552	0.87453	0.89345	0.80434	0.82469	0.89546	0.87456	0.91345
BGWOPSO	0.94555	0.95887	0.94567	0.97935	0.93657	0.95067	0.97657	1

Table 3. shows the Clinical Brest cancer dataset results. For different combination of dataset algorithms and testing-training ratios, different classifiers show different data of accuracy and AUC. For SVM classifier, the highest accuracy obtained was 0.97835 and AUC was 1 , both for BGWOPSO. For k-nearest neighbor, 0.93768 was the highest accuracy and 0.99351 was the highest AUC, obtained for 70-30 ratio. Using ANN, highest accuracy 0.93978 was obtained for BGWO and highest 0.92678 AUC for BGWOPSO. Using Random forest, 0.95678 was the highest accuracy and 1 was the highest AUC, both found for BGWOPSO at 80-20 ratio. Similarly, Logistic regression had also, highest accuracy and AUC were noted at 80-20 ratio.

Table 3. Highest Accuracy and AUC values for each classifier with best optimization technique for Clinical Breast cancer dataset

Ratio	**60:40**		**70:30**		**80:20**		**90:10**	
Optimizers	**Accuracy**	**AUC**	**Accuracy**	**AUC**	**Accuracy**	**AUC**	**Accuracy**	**AUC**
Classifier: SVM								
PSO	0.89678	0.89454	0.88678	0.83678	0.91234	0.8467	0.9298	0.85213
BGWO	0.86898	0.95564	0.88453	0.96760	0.83434	0.93768	0.83345	0.9923
BGWOPSO	0.90976	0.9634	0.95675	0.97098	0.9134	0.97438	0.97835	1
Classifier: K-Nearest Neighbor								
PSO	0.86678	0.9390	0.79445	0.87686	0.89324	0.8278	0.76980	0.83567
BGWO	0.83878	0.92343	0.90787	0.9390	0.89223	0.87456	0.91345	0.92678
BGWOPSO	0.90232	0.93447	0.93768	0.99351	0.91324	0.98456	0.90456	0.88453
Classifier: Artificial Neural Network								
PSO	0.82456	0.737876	0.70435	0.8897	0.80232	0.97891	0.82345	0.80378
BGWO	0.78578	0.78435	0.84687	0.91003	0.80678	0.85324	0.93978	0.84780
BGWOPSO	0.84590	0.89099	0.89445	0.9889	0.80456	0.80879	0.93784	0.92678
Classifier: Random forest								

PSO	0.806575	0.9490	0.81542	0.8490	0.82344	0.8734	0.83859	0.9967
BGWO	0.794785	0.98578	0.76768	0.94789	0.80665	0.80678	0.82980	0.81456
BGWOPSO	0.90532	0.99345	0.92467	0.97678	0.95678	1	0.84745	0.90467
Classifier: Logistic Regression								
PSO	0.79657	0.78789	0.80342	0.74567	0.75341	0.78678	0.81679	0.71457
BGWO	0.85642	0.87453	0.83345	0.80988	0.82789	0.82890	0.84567	0.91678
BGWOPSO	0.84565	0.90867	0.84667	0.99189	0.92677	0.99389	0.90456	0.8948

6.3 Liver Cancer Dataset Output

Table 4. shows the Microarray Liver cancer dataset results. For different combination of dataset algorithms and testing-training ratios, different classifiers show different data of accuracy and AUC. For SVM classifier, the highest accuracy obtained was 0.96865 and AUC was 1, both for BGWOPSO at 80-20 ratio. For k-nearest neighbor, 0.97848 was the highest accuracy and 0.98747 was the highest AUC, obtained for 70-30 ratio. For the ANN, highest accuracy was 0.98467 and highest AUC 0.98467 for BGWOPSO at ratio 70-30. Using Random forest, 0.96468 was the highest accuracy and 0.93876 was the highest AUC, both found for BGWOPSO at 70-30 ratio. Similarly, Logistic regression had also, highest accuracy and AUC were noted at 90-10 ratio.

Table 4. Highest Accuracy and AUC values for each classifier with best optimization technique for Microarray Liver cancer dataset

Ratio	**60:40**		**70:30**		**80:20**		**90:10**	
Optimizers	**Accuracy**	**AUC**	**Accuracy**	**AUC**	**Accuracy**	**AUC**	**Accuracy**	**AUC**
Classifier: SVM								
PSO	0.85456	0.87678	0.83454	0.82343	0.86567	0.86758	0.95633	0.93456
BGWO	0.84535	0.83424	0.89878	0.94345	0.87898	0.82342	0.87895	
BGWOPSO	0.92759	0.99567	0.95689	0.92454	0.96865	1	0.90125	0.98983

Classifier: K-Nearest Neighbor								
PSO	0.93423	0.94566	0.98958	0.85789	0.85625	0.80453	0.943567	0.935788
BGWO	0.947837	0.94768	0.84578	0.84578	0.94578	0.97895	0.937673	0.96658
BGWOPSO	0.92357	0.94646	0.97848	0.98747	0.95467	0.94749	0.904679	0.95476
Classifier: Artificial Neural Network								
PSO	0.86859	0.8674	0.84677	0.86877	0.86788	0.9467	0.86811	0.87945
BGWO	0.87657	0.8333	0.84562	0.86784	0.94564	0.86895	0.95789	0.83242
GWOPSO	0.83567	0.94567	0.98467	0.98467	0.97342	0.96775	0.92342	0.88567
Classifier: Random forest								
PSO	0.83490	0.8324	0.83455	0.84564	0.92345	0.88167	0.89567	0.8678
BGWO	0.82534	0.82343	0.96785	0.82341	0.87845	0.86784	0.92485	0.86690
BGWOPSO	0.84654	0.85675	0.96468	0.93876	0.94856	0.91589	0.91233	0.90784
Classifier: Logistic Regression								
PSO	0.87806	0.84350	0.88906	0.98076	0.87689	0.82345	0.85676	0.97899
BGWO	0.85678	0.89567	0.87806	0.84562	0.82346	0.91244	0.85789	0.82878
BGWOPSO	0.82523	0.84673	0.88590	0.93745	0.83456	0.94367	0.93633	0.99762

Table 5. shows the Clinical Liver cancer dataset results. For different combination of dataset algorithms and testing-training ratios, different classifiers show different data of accuracy and AUC. For SVM classifier, the highest accuracy obtained was 0.98804 and AUC was 1, both for BGWOPSO at 90-20 ratio. For k-nearest neighbor, 0.95678 was the highest accuracy and 0.99789 was the highest AUC, obtained for 70-30 ratio. For the ANN, highest accuracy was 0.97907 and highest AUC 0.9993 for BGWOPSO at ratio 80-20 and 70-30 respectively. Using Random forest, 0.95890 was the highest accuracy and 0.89457 was the highest AUC, both found for BGWOPSO at 80-20 and 90-10 ratio respectively. Similarly, Logistic regression had also, highest accuracy and AUC were noted at 70-30 ratio.

Table 5. Highest Accuracy and AUC values for each classifier with best optimization technique for Clinical Liver cancer dataset

Ratio	60:40		70:30		80:20		90:10	
Optimizers	**Accuracy**	**AUC**	**Accuracy**	**AUC**	**Accuracy**	**AUC**	**Accuracy**	**AUC**
Classifier: SVM								
PSO	0.85678	0.77906	0.73056	0.94096	0.94245	0.83567	0.91223	0.79677
BGWO	0.90785	0.89436	0.92457	0.99978	0.93367	0.87256	0.94478	0.92789
BGWOPSO	0.91657	0.9867	0.92467	0.95789	0.95467	0.8966	0.98804	1
Classifier: K-Nearest Neighbor								
PSO	0.82465	0.89436	0.83346	0.8825	0.84678	0.95881	0.85679	0.7833
BGWO	0.88689	0.73483	0.856764	0.9878	0.8367	0.9987	0.84464	0.79989
BGWOPSO	0.90345	0.89022	0.95678	0.99789	0.87789	0.84567	0.76956	0.80789
Classifier: Artificial Neural Network								
PSO	0.92567	0.78234	0.94890	0.99767	0.8367	0.95678	0.93567	0.97678
BGWO	0.90678	0.94365	0.94909	0.92980	0.92456	0.96785	0.93478	0.99465
BGWOPSO	0.93890	0.93945	0.96524	0.9993	0.97907	0.78342	0.80686	0.86456
Classifier: Random forest								
PSO	0.78879	0.87678	0.94789	0.78567	0.94907	0.76456	0.93890	0.75980
BGWO	0.87906	0.83799	0.89089	0.81879	0.89678	0.76789	0.91996	0.79890
BGWOPSO	0.80678	0.80789	0.80456	0.79896	0.95890	0.86907	0.93458	0.89457
Classifier: Logistic Regression								
PSO	0.75678	0.76789	0.84789	0.78879	0.83896	0.7289	0.8267	0.74456
BGWO	0.84678	0.75789	0.85456	0.72897	0.80674	0.73789	0.79324	0.75879
BGWOPSO	0.80879	0.79678	0.89678	0.80853	0.79897	0.76567	0.85789	0.80789

6.4 Summary of Results

Table 6. displays the summary of the Leukemia cancer dataset using different optimizers and classifier. After implementing various combinations of all five classifiers with the all the three-optimization technique we got the following results for the best and highest accuracy and AUC.

For SVM the best results were obtained for BGWOPSO and BGWO respectively. However, BGWOPSO was the best optimizer for all classifier such as, k-nearest neighbors, Artificial neural network, Random forest, Logistic regression.

Table 6. Result Analysis of Leukemia Dataset

Classifier	**Optimization Technique**	**Ratio**	**Accuracy**	**AUC**
SVM	BGWOPSO	80:20	96.79%	
	BGWO	90:10		1
K-nearest neighbor	BGWOPSO	80:20	95.24%	
	BGWOPSO	80:20		0.97686
Artificial Neural Network	BGWOPSO	80:20	97.34%	
	BGWOPSO	70:30		0.95988
Random Forest	BGWOPSO	80:20	96.86%	
	BGWOPSO	80:20)		0.95342
Logistic Regression	BGWOPSO	(90:10)	94.56%	
	BGWOPSO	(90:10)		0.95762

Table 7. presents the summary of the Microarray Breast cancer dataset using different optimizers and classifier. After executing many combinations of all five classifiers with the all the three-optimization technique we found the following results for the best and highest accuracy and AUC. For SVM the best results were obtained for BGWOPSO at 90:10 ratio which is 0.95967 and 0.98443 respectively. In fact, BGWOPSO was the best optimizer for all classifier such as, k-nearest neighbors, Artificial neural network, Random forest, Logistic regression.

Table 7. Result Analysis of Microarray Breast Cancer Dataset

Classifier	**Optimization Technique**	**Ratio**	**Accuracy**	**AUC**
SVM	BGWOPSO	(90:10)	95.97%	
	BGWOPSO	(90:10)		0.98443
K-nearest neighbor	BGWOPSO	(70:30)	98.68%	
	BGWOPSO	(70:30)		0.99677
Artificial Neural Network	BGWOPSO	(90:10)	95.36%	
	BGWOPSO	(90:10)		0.99638
Random Forest	BGWOPSO	(80:20)	94.68%	
	BGWOPSO	(80:20)		1
Logistic Regression	BGWOPSO	(90:10)	97.66%	
	BGWOPSO	(90:10)		1

Table 8. indicates the summery of output for clinical Breast cancer dataset. For four different classifiers SVM, KNN, logistic regression and Random Forest, BGWOPSO was observed to be the most optimum algorithm as best AUC and accuracy were noted for the same method. For ANN, BGWO displayed the best accuracy and Again, BGWOPSO displayed the best AUC. Amongst all the combinations we have tried for the classifiers and optimization technique, we obtained the best accuracy for SVM classifier with the value of 0.97835 for the BGWOPSO at 90:10 ratio.

Table 8. Result Analysis of Clinical Breast Cancer Dataset

Classifier	**Optimization Technique**	**Ratio**	**Accuracy**	**AUC**
SVM	BGWOPSO	(90:10)	97.84%	
	BGWOPSO	(90:10)		1

K-nearest neighbor	BGWOPSO	(70:30)	93.77%	
	BGWOPSO	(70:30)		0.99351
Artificial Neural Network	BGWO	(90:10)	93.98%	
	BGWOPSO	(90:10)		0.92678
Random Forest	BGWOPSO	(80:20)	95.68%	
	BGWOPSO	(80:20)		1
Logistic Regression	BGWOPSO	(80:20)	92.68%	
	BGWOPSO	(80:20)		0.99389

Table 9. indicates the summery of output for Microarray Liver cancer dataset. For all classifiers SVM, KNN, LR ,RF, and ANN BGWOPSO was observed to be the most optimum algorithm as best AUC and accuracy were observed for the same method. Maximum accuracy was obtained for the classifier ANN that is 0.98467.

Table 9. Result Analysis of Microarray Liver Cancer Dataset

Classifier	**Optimization Technique**	**Ratio**	**Accuracy**	**AUC**
SVM	BGWOPSO	(80:20)	96.87%	
	BGWOPSO	(80:20)		1
K-nearest neighbor	BGWOPSO	(70:30)	97.85%	
	BGWOPSO	(70:30)		0.98747
Artificial Neural Network	BGWOPSO	(70:30)	98.47%	
	BGWOPSO	(70:30)		0.98467
Random Forest	BGWOPSO	(70:30)	96.47%	
	BGWOPSO	(70:30)		0.93876
Logistic Regression	BGWOPSO	(90:10)	93.63%	
	BGWOPSO	(90:10)		0.99762

In Table 10. summary of Clinical liver cancer dataset is shown. As we can understand from the table with the all classifier the BGWOPSO optimizer perform well for all three datasets. Maximum AUC were obtained for three classifier k-nearest neighbor, Random forest and SVM that is 1.

Table 10. Result Analysis of Clinical Liver Cancer Dataset

Classifier	Optimization Technique	Ratio	Accuracy	AUC
SVM	BGWOPSO	(90:10)	98.81%	
	BGWOPSO	(90:10)		1
K-nearest neighbor	BGWOPSO	(70:30)	95.68%	
	BGWOPSO	(70:30)		0.99789
Artificial Neural Network	BGWOPSO	(80:20)	97.91%	
	BGWOPSO	(70:30)		0.9993
Random Forest	BGWOPSO	(80:20)	95.89%	
	BGWOPSO	(90:10)		0.89457
Logistic Regression	BGWOPSO	(70:30)	89.68%	
	BGWOPSO	(70:30)		0.80853

In the Table 11, the highlighted Green color shows a minimum number of features selected, the chart shows the no of features being selected for each data set of the five classifiers with the three feature selection methods for leukemia, breast cancer, and Liver cancer datasets. Dataset (I) indicate Microarray dataset and dataset (II) indicates Clinical Dataset. We observe the following from the table.

1. SVM with BGWOPSO gives 228 features for Leukemia dataset, which is also the minimum no of features selected for this dataset of features selected from all classifiers with this two algorithms - BGWO and GWOPSO.
2. ANN with BGWOPSO gives 212 features for Breast cancer (dataset I), which is also the minimum no of features selected for this dataset.
3. LR with BGWOPSO provides 3 features for Breast cancer (dataset II). Which is the least no of features selected from all classifiers with all two BGWO and BGWOPSO algorithms
4. ANN with BGWOPSO gives 213 features for Liver cancer (dataset I), which is also the least no of features selected for this dataset
5. SVM with BGWOPSO provides 2 features for the Liver cancer (dataset II). Which is the least no of features selected from all classifiers with all two algorithms

Table 11. Selected Best Number of Features

Classifier	**Optimizer**	
	BGWO	**BGWOPSO**
Leukemia		
SVM	229	228
KNN	244	240
ANN	251	250
LR	252	250
RF	275	256
Breast Cancer(dataset I)		
SVM	277	235
KNN	267	282
ANN	214	212
LR	289	256
RF	279	287
Breast Cancer(dataset II)		

SVM	5	4
KNN	6	5
ANN	4	4
LR	4	3
RF	7	6
Liver Cancer(dataset I)		
SVM	289	278
KNN	235	256
ANN	234	213
LR	286	267
RF	287	234
Liver Cancer(dataset II)		
SVM	4	2
KNN	4	3
ANN	3	5
LR	5	4
RF	4	4

6.5 Discussion

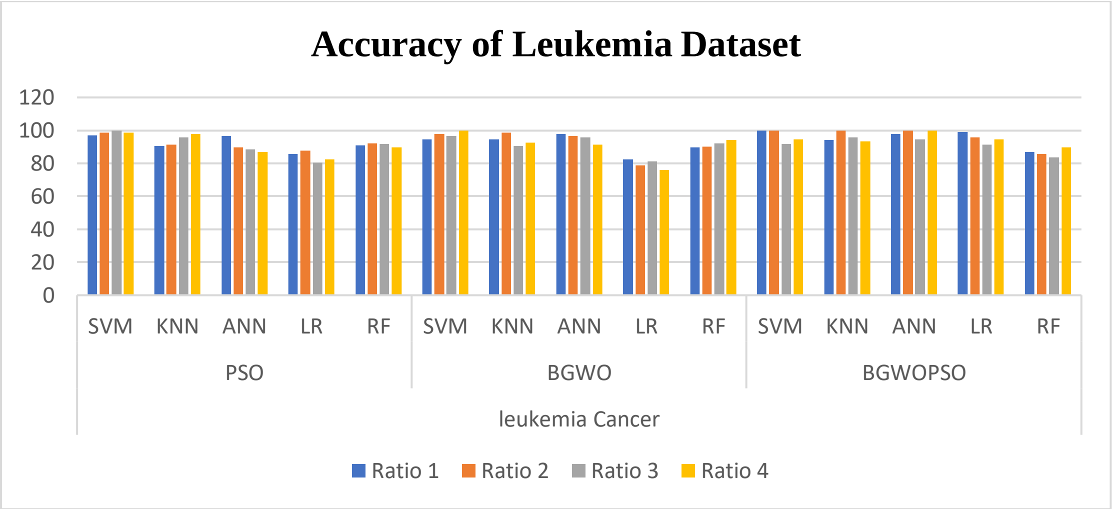


Figure 19. Graphical Analysis of Leukemia Dataset

Figure 19.describes the graphical representation of the Accuracy of the Leukemia Dataset in terms of Bar chart. The Chart shows values for four different ratio's (60:40 as Ratio 1, 70:30 as Ratio 2, 80:20 as Ratio 3, 90:10 as Ratio 4), of five different classifiers (SVM, KNN,ANN, LR, RF), where the X-axis is represented with three different algorithms and Y-axis is represented as accuracy in percentage.

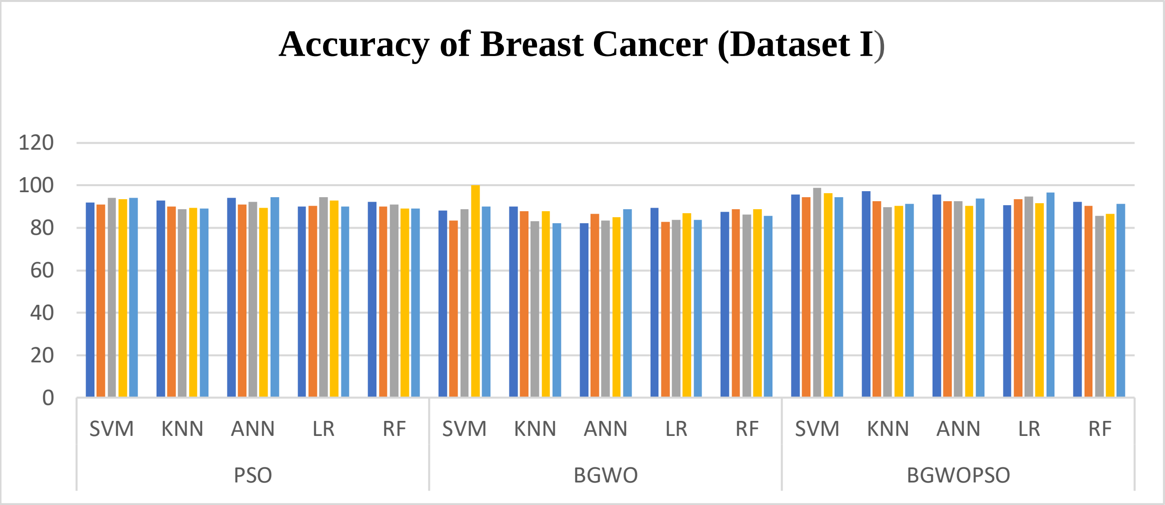


Figure 20. Graphical Analysis of Microarray Breast Cancer Dataset

Figure 20. describes the graphical representation of the Accuracy of Breast cancer (Dataset I) in terms of Bar chart. The Chart illustrates values for four different ratio's (60:40 as Ratio 1, 70:30 as Ratio 2, 80:20 as Ratio 3, 90:10 as Ratio 4), of five different classifiers (SVM, KNN,ANN, LR, RF), where the X-axis is represented with three different algorithms and Y-axis is represented as accuracy in percentage.

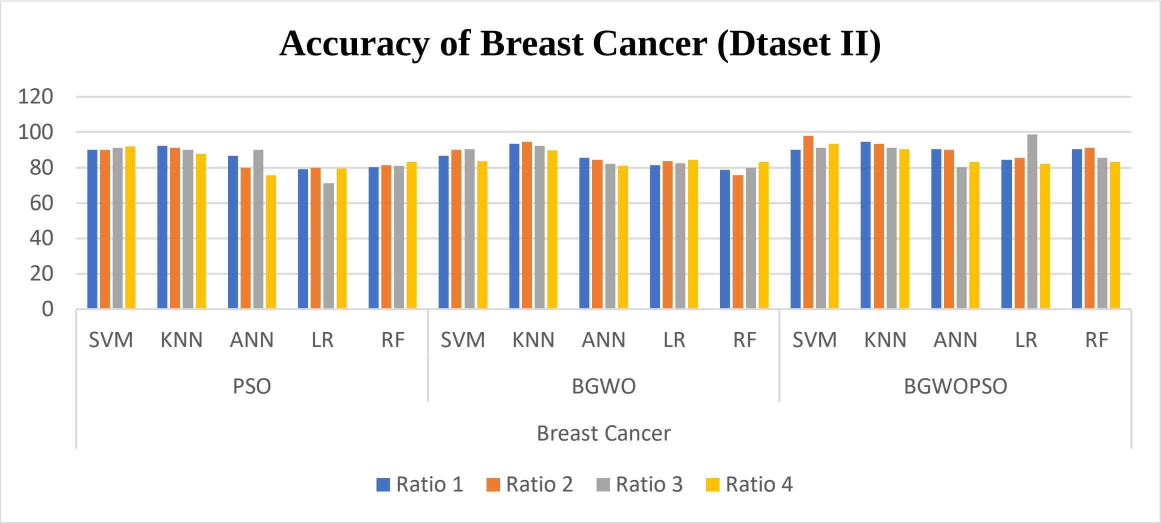


Figure 21. Graphical Analysis of Clinical Breast Cancer Dataset

Figure 21. describes the graphical representation of the Accuracy of Breast cancer (Dataset II) in terms of Bar chart. The Chart shows values for four different ratio's (60:40 as Ratio 1, 70:30 as Ratio 2, 80:20 as Ratio 3, 90:10 as Ratio 4), of five different classifiers (SVM, KNN,ANN, LR, RF), where the X-axis is represented with three different algorithms and Y-axis is represented as accuracy in percentage.

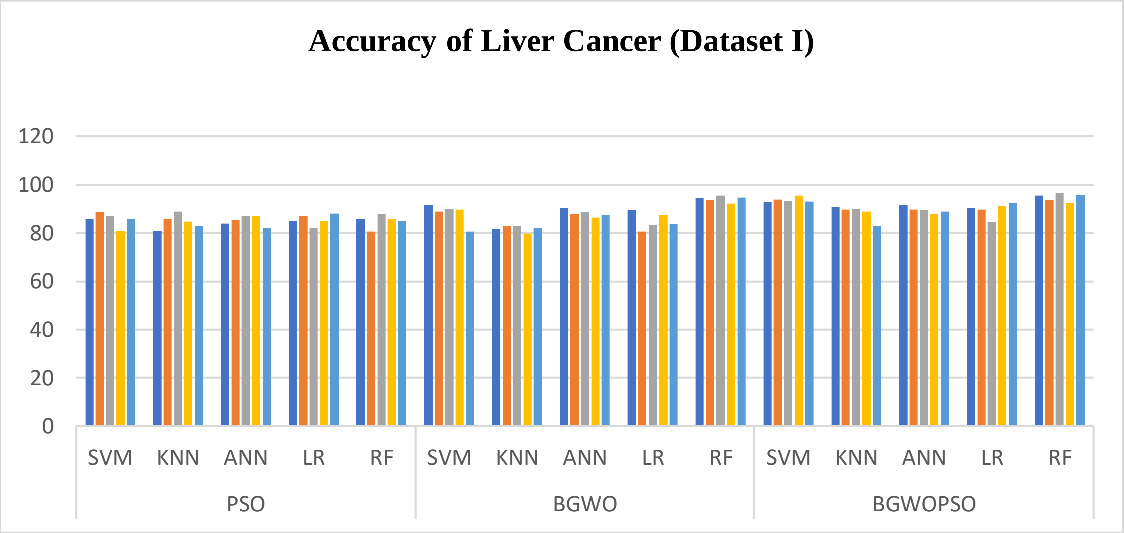


Figure 22. Graphical Analysis of Microarray Liver Cancer Dataset

Figure 22. describes the graphical representation of the Accuracy of the Liver Cancer (Dataset I) in terms of Bar chart. The Chart shows values for four different ratio's (60:40 as Ratio 1, 70:30 as Ratio 2, 80:20 as Ratio 3, 90:10 as Ratio 4), of five different classifiers (SVM, KNN,ANN, LR, RF), where the X-axis is represented with three different algorithms and Y-axis is described as accuracy in percentage.

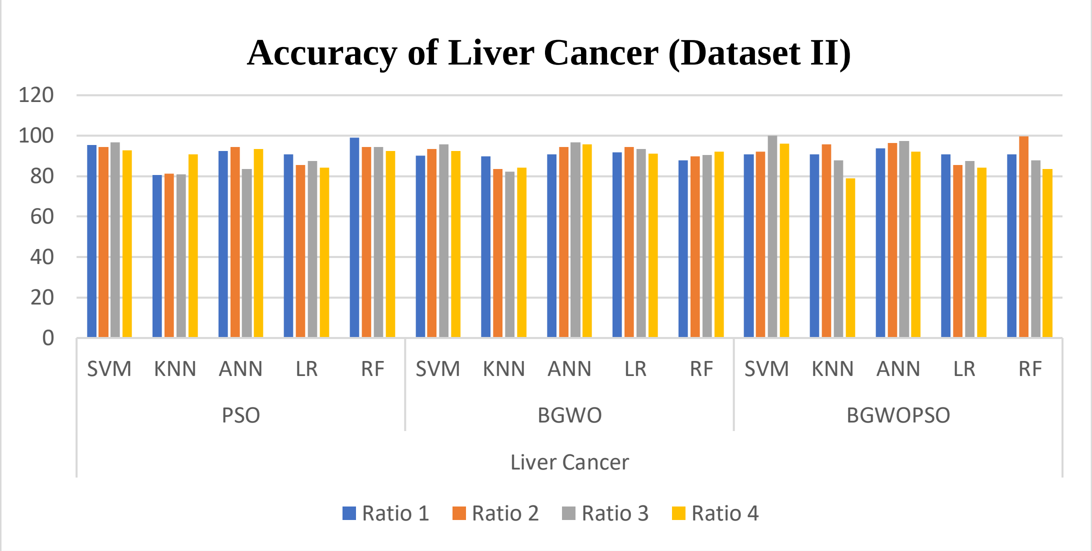


Figure 23. Graphical Analysis of Clinical Liver Cancer Dataset

Figure 23. describes the graphical representation of the Accuracy of the Liver Cancer (Dataset II) in terms of Bar chart. The Chart shows values for four different ratio's (60:40 as Ratio 1, 70:30 as Ratio 2, 80:20 as Ratio 3, 90:10 as Ratio 4), of five different classifiers (SVM, KNN,ANN, LR, RF), where the X-axis is represented with three different algorithms and Y-axis is described as accuracy in percentage.

Chapter 7

Conclusions and Future Work

The hybridization of binary version of grey wolf optimization and particle swarm optimization (BGWOPSO) was introduced for efficient feature selection. To check the efficiency and effectiveness of the algorithm, we used five cancer datasets. The BGWOPSO was compared with the original Particle Swarm Optimization (PSO) and Binary Grey Wolf Optimizations (BGWO). Accuracy and Area under the ROC curve (AUC) was used to compare the performance of the classifiers with the feature selection techniques. The results show that BGWOPSO performs better than PSO and BGWO with all the five classifiers (SVM, Neural Networks, KNN, Random Forest and Logistic Regression) on the datasets. We observe from the graphs that all classifiers perform well, but SVM gives maximum accuracy with all three optimization algorithms. Moreover, we also observe that SVM and ANN are strong competitors of KNN. Whereas, LR and RF classifier work well for breast cancer and leukemia cancer, but they give slightly low accuracy for liver cancer dataset.

As future work, we suggest to apply the hybrid algorithm in other real-world problems such as engineering optimization problems, scheduling problems, and molecular potential energy function. Moreover, this algorithm can be applied on different datasets with different classifiers and compare the results in accuracy as well as in feature selection.

References

[1] J. Han, J. Pei, and M. Kamber, Data Mining: Concepts and Techniques. Amsterdam, The Netherlands: Elsevier, 2011

[2] M. Dash and H. Liu, ‘‘Feature selection for classification,’’ Intell. Data Anal., vol. 1, nos. 1–4, pp. 131–156, 199

[3] I. Guyon and A. Elisseeff, ‘‘An introduction to variable and feature selection,’’ J. Mach. Learn. Res., vol. 3, pp. 1157–1182, Jan. 2003.

[4] H. Liu, H. Motoda, R. Setiono, and Z. Zhao,‘‘Feature selection: An ever evolvingfrontierindatamining,’’inProc.4thWorkshopF eatureSelection Data Mining, May 2010, pp. 4–13.

[5] H. Liu and Z. Zhao, ‘‘Manipulating data and dimension reduction methods: Feature selection,’’ in Encyclopedia Complexity Systems Science. New York, NY, USA: Springer, 2009, pp. 5348–5359

[6] E.-G. Talbi, Metaheuristics: From Design to Implementation, vol. 74,, Hoboken, NJ, USA: Wiley, 2009.

[7] E. Emary, H. M. Zawbaa, and A. E. Hassanien‘‘Binary grey wolf optimization approaches for feature selection,’’ Neurocomputing, vol. 172, pp. 371–381, 2016.

[8] Al-Tashi, H. Rais, and S. Jadid,, ‘Feature selection method based on grey wolf optimization for coronary artery disease classification’’ in Proc. Int. Conf. Reliable Inf. Commun. Technol pp. 257–266., 2018.

[9] R. Bello, Y. Gomez, A. Nowe, and M.M. Garcia, ‘Two step particle swarm optimization to solve the feature selection problem,’’inProc.7thInt.Conf. Intell. Syst. Des. Appl. (ISDA), pp. 691–696, oct. 2007.

[10] M. M. Kabir, M. Shahjahan, and K. Murase, ‘‘A new local search-based hybrid genetic algorithm for feature selection,’’ Neurocomputing, vol. 74, no. 17, pp. 2914–2928,, oct 2011.

[11] E. ZorarpacI and S. A. Özel, ‘‘A hybrid approach of differential evolution and artificial bee colony for feature selection,’’ Expert Syst. Appl., vol. 62, pp. 91–103, nov. 2016.

[12] D. N. Wilke. “Analysis of the particle swarm optimization algorithm”, “Analysis of the particle swarm optimization algorithm”, Master's Dissertation,, University of Pretoria, 2005.

[13] J. Kennedy and R. Eberhart, ‘‘Particle swarm optimization’’ in Proc. ICNN Int. Conf. Neural Netw., pp. 1942–1948., Dec. 1995, .

[14] J. Kennedy, ‘‘The particle swarm: Social adaptation of knowledge,’’ in Proc. IEEE Int. Conf. Evol. Comput. (ICEC), pp. 303–308., Apr. 1997.

[15] E. Emary, H. M. Zawbaa, and A. E. Hassanien, ‘‘Binary grey wolf optimization approaches for feature selection,’’ Neurocomputing, vol. 172, pp. 371–381., Jan. 2016..

[16] Z. R. Yang, "Biological applications of support vector machines," Briefings in bioinformatics, vol. 5, pp. 328-338,, 2004.

[17] N. Singh and S. B. Singh, ‘A novel hybrid GWO-SCA approach for optimization problems,’’ Eng. Sci. Technol. Int. J., vol. 20, no. 6, pp. 1586–1601,, 2017.

[18] C. L. Blake, "‘‘CJ Merz UCI repository of machine learning databases,’’ Ph.D. dissertations, Dept. Inform. Comput. Sci.,," Univ. California, Irvine, CA, USA,, 1998..

[19] S. Pal, Grey Wolf Optimization Trained Feed Foreword Neural Network for Breast Cancer Classification. International Journal of Applied Industrial Engineering, 5(2), pp.21-29, 2018.

[20] N. S. Altman, ‘‘An introduction to kernel and nearest-neighbor nonparametric regression,’’ Amer. Statist., vol. 46, no. 3, pp. 175–185, 1992.

[21] R. N. Khushaba, Al-Ani Ahmed, and Al-Jumaily Adel."Feature subset selection using differential evolution and a statistical repair mechanism." Expert Systems with Applications, Vol. 38, No. 9, pp. 11515-11526, 2011.

[22] S. Aljawarneh, A. Monther, and B. Y. Muneer "Anomaly-based intrusion detection system through feature selection analysis and building hybrid efficient model," Journal of Computational Science, Vol. 25, pp. 152-160,, 2018.

[23] P. Moradi, and G. Mozhgan."A hybrid particle swarm optimization for feature subset selection by integrating a novel local search strategy," Applied Soft Computing, Vol. 43, pp. 117-130, 2016.

[24] "_A_Novel_Aggregate_Gene_Selection_Method_for_Microarray_Data_Classification/figure https://www.researchgate.net/publication/275366089_".

[25] J. Mijnes, "“SNiPER: a Novel Hypermethylation Biomarker Panel for Liquid Biopsy Based Early Breast Cancer Detection.” Oncotarget, U.S. National Library of Medicine,," 5 Nov. 2019.

[26] M. A. Tawhid and A. F. Ali, ‘‘A Hybrid grey wolf optimizer and genetic algorithm for minimizing potential energy function,’’ Memetic Comput., vol. 9, no. 4, pp. 347–359,, 2017..

[27] P. J. Gaidhane and M. J. Nigam, ‘‘A hybrid grey wolf optimizer and artificial bee colony algorithm for enhancing the performance of complex systems,’’ J. Comput. Sci., vol. 27, pp. 284–302,, jul ,2018.

[28] Y. Wang, J. Chang, X. Li, R. Pathak, V. Sridharan, T. Jones, X.W. Mao, G. Nelson, M. Boerma, M. Hauer-Jensen, D. Zhou, L. Shao. Low Doses of Oxygen Ion Irradiation Cause Long-Term Damage to Bone Marrow Hematopoietic Progenitor and Stem Cells in Mice. PLoS One 2017 Dec 12; 12(12):e0189466.

[29] Github " availabile at : https://shiring.github.io/machine_learning/2017/01/15/rfe_ga_post".

[30] "“Liver Cancer.” Mayo Clinic, Mayo Foundation for Medical Education and Research, online available at: https://www.mayoclinic.org/diseases-conditions/liver-cancer/symptoms-causes/syc-20353659.," 4 May 2019, .

[31] M. K. Thomsen, "“The CGAS-STING Pathway Is a Therapeutic Target in a Preclinical Model of Hepatocellular Carcinoma.” Oncogene, U.S. National Library of Medicine, online available https://www.ncbi.nlm.nih.gov/pubmed/31740782," 18 Nov. 2019, .

[32] Santos, ""A new cluster-based oversampling technique for improving survival prediction of hepatocellular carcinoma patients," Journal of biomedical informatics, 58, 49-59,," 2015.

[33] Broidy, UCI Libraries Celebrate UCI. College & Research Libraries News, 61(6), pp.473-47, (2019)..

[34] K.-C. Wong, Computational Biology and Bioinformatics: Gene Regulation: CRC Press,, 2016.

[35] p. perner, "Machine Learning and Data mining in pattern recorgnition," USA, new york , 2018.

[36] S. Kaushik., “Feature Selection Methods with Example (Variable Selection Methods).” Analytics Vidhya, available at: https://www.analyticsvidhya.com/blog/2016/12/introduction-to-feature-selection-methods-with-an-example-or-how-to-select-the-right-variables.

[37] J. Kennedy and R. Eberhart, ‘‘Particle swarm optimization,’’ in Proc. ICNN Int. Conf. Neural Netw, pp. 1942–194, Dec. 1995.

[38] Singh, N. Singh and S. B. ‘‘Hybrid algorithm of particle swarm optimization and Grey Wolf optimizer for improving convergence performance,’’ J. Appl. Math., vol. 2017, 2017, Art. no. 2030489.

[39] J. Hamidi., Control-System Design Using Particle Swarm (PSo) 2012.

[40] E. R. Shi Y, A modified particle swarm optimizer. In Proceedings of the IEEE Inter-national Conference on Evolutionary Computation, 69–73., 1998.

[41] S. Mirjalili, S. M. Mirjalili, and A. Lewis, ‘‘Grey Wolf Optimizer,’’ Adv. Eng. Softw., vol. 69, pp. 46–61,, Mar. 2014..

[42] E.-G. Talbi, “A taxonomy of hybrid metaheuristics, ”Journal of Heuristic,vol.8,no.5,pp.541–564,2002.

[43] "Binary Optimization Using Hybrid Grey Wolf Optimization for Feature Selection - IEEE Journals & Magazine. online available at :https://ieeexplore.ieee.org/abstract/document/8672550.".

[44] "R. Pupale.," *online available at :https://towardsdatascience.com/https-medium-com-pupalerushikesh-svm-f4b42800e989.,* 11 Feb. 2019, .

[45] U. Alon, N. Barkai, D. A. Notterman, K. Gish, S. Ybarra, D. Mack,"Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays," Proceedings of the National Academy of Scien".

[46] N. Vapnik. Schölkopf, Bernhard, Zhiyuan Luo, and Vladimir Vovk, Schölkopf, "Empirical inference. Festschrift in honor of Vladimir," 2013.

[47] Shawe-Taylo, Nello Cristianini and John, An Introduction to Support Vector Machines and Other Kernel-based Learning Methods, 2000.

[48] J. Friedman, T. Hastie, and R. Tibshirani, "The elements of statistical learning vol. 1: Springer series in statistics Springer," berlin, 2001.

[49] "Neural Networks” Neural Networks - OpenCV 2.4.13.7 Documentation," . Available: https://docs.opencv.org/2.4/modules/ml/doc/neural_networks.html?highlight=ann_mlp..

[50] towards datascience , Available: https://towardsdatascience.com/the-mostly-complete-chart-of-neural-networks-explained-3fb6f2367464.

[51] "Logistic Regression.” Logistic Regression - ML Cheat sheet Documentation,". Available: https://ml-cheatsheet.readthedocs.io/en/latest/logistic_regression.html.

[52] T. Cover and P. Hart, "Nearest neighbor pattern classification," IEEE transactions on information theory, vol. 13, pp. 21-27, 1967.

[53] B. R. Kowalski and C. Bender, "k-Nearest Neighbor Classification Rule (pattern recognition) applied to nuclear magnetic resonance spectral interpretation," Analytical Chemistry, vol. 44, pp. 1405-1411, 1972..

[54] Introduction to k- nearest classifier K-NN classifier , spring journel , Zhang, china, 2012 .

[55] T. Yiu, " "Understanding Random Forest." Medium, Towards Data Science,," [Online]. Available: https://towardsdatascience.com/understanding-random-forest-58381e0602d2..

[56] T. K. Ho,"Random decision forests," in Document Analysis and Recognition, 1995., Proceedings of the Third International Conference on, 1995, pp. 278-282..

[57] T. K. Ho, "The random subspace method for constructing decision forests," IEEE transactions on pattern analysis and machine intelligence, vol. 20, pp. 832-844, 1998.

[58] Example of Decision tree and relationship between random fores tree:"Stand Up. Available: https://courses.lumenlearning.com/standupspeakout/chapter/types-of-presentation-aids/..

[59] " Random Forests – Sereen's blog," Available: https://15-6477.ca.uts.edu.au/data-algorithms-random-forests/.

[60] "mathwork matlab," Available: https://www.mathworks.com/discovery/numerical-analysis.html.