8-1-2022

Rule Learning and Swarm Intelligence Techniques for Feature Selection Optimization

Tina Gui

Rule Learning and Swarm Intelligence Techniques for Feature Selection Optimization

A Dissertation
presented in partial fulfillment of requirements
for the degree of Doctor of Philosophy
in the Department of Computer and Information Science
The University of Mississippi

by

Tina Gui

August 2022

ABSTRACT

In mathematics and computer science, solving an optimization problem is to find the best solution from all possible outcomes. In this dissertation work, two kinds of algorithms are considered to address the problems in Microarray Analysis, Numerical Optimization and Wireless Sensor Networks. In gene expression analysis and classification, feature selection is an important process of selecting the optimal subset of relevant features or useful data for further study and prediction. The main objective of feature selection is challenging due to the large search space, computational time, imbalanced samples, and quality of the selected drivers. It is necessary to construct a discriminative and stable feature selector that is robust to noises and outliers and able to select highly informative gene sets.

To address the issue of the quality of the generated features, we first propose a rule based feature selection and elimination approach, Top Discriminating Pairs (TDP), which aims to reveal features that are highly ranked according to their discrimination power. Our experiment combines the TDP methodology with various classifiers to achieve a significant feature set. To illustrate the effectiveness of this approach, we compare the proposed ap- proach with the traditional Top Scoring Pairs (TSP) method as the baseline on various artificial and real datasets. This work provides a new effective method for feature selection and dimensionality reduction in machine learning.

In order to reduce search space and improve computational capability, we next con- sider

Swarm Intelligence based methods which mimic the social behaviors of natural insects or artificial systems. These techniques have recently attracted researchers' attention and have been adopted to tackle complex feature selection problems. Biologically inspired computing has successfully been used in many areas that need simplicity in computation, optimized intelligence search, and machine learning techniques. We present a comprehensive study of the recent applications of Swarm Intelligence (SI) for optimizing feature selection processes with respect to their experimental settings and performance metrics. Then we introduce our Spider Monkey Optimization (SMO) based feature selection approach using Microarray data for human cancer classification and prediction. The results show that our SMO feature selector combining three classifiers achieved the best accuracy scores on most of the test data sets.

Furthermore, we extend the abilities of SMO by solving other problems such as finding the optimal routes in wireless communication among sensors. In this work, we aim to study the mechanism of the SMO algorithm, formulating the mathematical model of its social behavior patterns and to improve the traditional routing protocols in terms of low-energy consumption and overall system quality of the network. The experimental results show that our approach is self-organized, scalable and can be easily adapted to wireless sensor networks.

ACKNOWLEDGMENTS

I would like to express the deepest appreciation to my committee members Dr. Dawn E. Wilkins, Dr. Yixin Chen, Dr. Feng Wang, and Dr. Hailin Sang, who have been supportive of my career goals and have provided me professional guidance and taught me a great deal about both scientific research and life in general. Without their guidance and persistent help this dissertation would not have been possible.

Nobody has been more important to me in the pursuit of this dissertation than the members of my family. I would like to give sincere thanks to my parents, my husband, and my kids.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

Gene expression classification and feature selection are commonly used techniques to diagnose diseases in Microarray analysis. In fact, numerous classifiers have been pursued for correctly identifying cancerous patients based on numerical molecular information. Popular techniques for solving the diagnosis problem include Support Vector Machine (SVM) [20], Decision Tree (DT) [84], Random Forests (RF) [47, 13], Prediction Analysis of Microarray (PAM) [96], and Top Scoring Pair (TSP) [37]. However, there is no classifier that always outperforms the others. For example, Top Scoring Pairs does not extend to some difficult datasets, such as those containing a small number of samples. Thus, the objective of this study is to propose a general approach, combining dimensionality reduction and feature elimination, based on the discriminating power of gene pairs.

1.1 Rule-based Learning

1.1.1 Top Scoring Pairs

In gene expression profiles, we consider G genes whose expression levels can be assigned as $X = \{X_1, X_2, ..., X_G\}$. Each profile X has a true class label in $C = \{1, 2, ..., c\}$. In our implementation, we only consider two classes, either class 1 or class 2. Geman et al. summarized the general process of calculating expression values for each pair of genes – they detected “marker gene pairs” (i, j) under the rule when $X_i < X_j$ from class 1 to class 2 [37]. The classification is based on the distinguished pairs and the quantities of interest are,

$$p_{ij}(C) = P(X_i < X_j|C) \tag{1.1}$$

the score of each pair of genes is calculated as,

$$\Delta_{ij} = |p_{ij}(1) > p_{ij}(2)| \tag{1.2}$$

Then the paired genes are ranked based on the Δ_{ij} values in descending order and the TSP classifier only selects the top scoring pairs.

1.1.2 k-Top Scoring Pairs

The Top Scoring Pairs (TSPs) may change when the training data are perturbed by adding or deleting a few examples [37]. In Tan's work, they introduced the k-TSP classifier which increases the accuracy of the TSP classifier and generates a more stable classifier. The motivations of using k-TSP classifier are: 1) there are many top scoring pairs with the same informative ordering (same Δ score); 2) it combines the discriminating power of many 'weaker' rules; 3) it achieves better combined scores [95]. The k-TSP algorithm is similar to TSP method. In the prediction of TSP classifier (h_{TSP}), we suppose $p_{ij}(1) > p_{ij}(2)$ and X_{new} is a new sample. Then the decision rule is,

$$h_{TSP}(X_{new}) = \begin{cases} C = 1, & X_{i,new} > X_{j,new} \\ C = 2, & otherwise \end{cases} \tag{1.3}$$

The k-TSP classifier selects k-top disjoint pairs of genes in prediction according to 1.3. It simply chooses the class receiving the majority votes and consists of a list of ranked TSPs genes from the largest scores to smallest scores in Eq. 1.4 and Eq. 1.5,

$$h_{k-TSP}(X_{new}) = \underset{C=1,2}{\arg\max} \sum_{u=1}^{k} I(h_u(X_{new}) = C) \tag{1.4}$$

where

$$I(h_u(X_{new})) = \begin{cases} 1, & h_u(X_{new}) = C \\ 0, & otherwise \end{cases}, C = (1,2) \tag{1.5}$$

Ties are broken by sorting the pairs that achieve the same score Δ using the secondary ranking score Γ (Gamma) [95], which is based on the ranking differences in each sample in each class, defined to be $\Gamma_{ij} = |\gamma_{ij}(1) - \gamma_{ij}(2)|$, where

$$\gamma_{ij}(C) = \frac{\sum_{n \in N}(R_{i,n} - R_{j,n})}{|C|} \tag{1.6}$$

where $|C|$ denotes the number of samples. The k disjoint pairs of genes with the largest score values Γ are selected from those pairs with the highest value Δ_{ij} in TSP classifier (Eq. 1.2). Both original TSP and k-TSP techniques are competitive with PAM and SVM classifiers. However, the TSP-family classifiers are easier to interpret and involve many fewer genes.

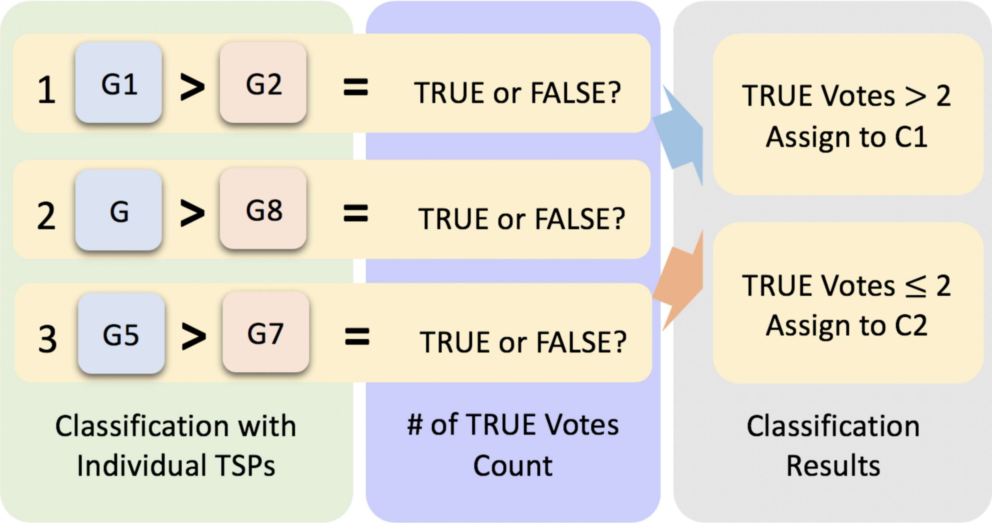


Figure 1.1. k-TSP Classifier

1.1.3 Association Rule Mining

Association rule mining (ARM) was proposed by George Piatetsky-Shapiro in 1991 and is commonly applied to market basket analysis [81]. For example, the parable of the beer and diapers, the rule *diapers* $\Rightarrow$ *beer* indicates that a customer who often buys diapers is likely to also buy beer. Such information can be used as the basis for decision making and can be applied to the discovery of frequent patterns from Microarray data as well. It is very useful in the following: 1) to discover association rules, which can only reveal biological relevant correlations between genes to identity gene regulation pathways and also help to uncover gene networks [21], 2) to discover bi-clustering of gene expression [115]. Association rule mining is defined as [2],

1. Let $I = \{i_1, i_2, ..., i_m\}$ be a set of m distinct attributes, also called items. Let $D = \{t_1, t_2, ..., t_n\}$ be a database, a set of transactions. Each transaction in D has a unique identifier and contains a set of items. An association rule is defined as an implication of the form $X \Rightarrow Y$ where $X, Y \subseteq I$ and $X \cap Y = \emptyset$. X is called antecedent (left hand side) and Y is called consequent (right hand side).

2. The support of an itemset X is the ratio of the number of occurrences of transactions in the dataset which contain the itemset to the number of total transactions.

3. The confidence of an association rule is the ratio of the support for occurrences of transactions where X and Y both appear to the number of transactions that contain just X.

1.2 Swarm Intelligence

Swarm Intelligence (SI) is a meta-heuristic and flexible optimization technique that mimics the behavior of swarms of bees, ants, monkeys, birds or fish [16]. The swarms follow very simple rules, and there is no centralized control structure. These properties make swarm intelligence a successful design paradigm to deal with increasingly complex problems, such as optimization problems. SI has been applied to and shown promising results in a variety of fields, including Social Media [16], Robotic Systems [103], Computational Biology [114], Wireless Sensor Networks [41], Power System [97], and Healthcare [50]. In this paper, we focus on recent developments of SI in the area of Bioinformatics, especially introducing the applications as optimizers for feature selection in gene expression analysis. Figure 1.2 shows how popular the topic “Swarm Intelligence” is entered by web searchers in real-time across different regions in the world.

In previous studies, Swarm Intelligence based algorithms have shown high potential to achieve optimal solutions in complex structures and can be used to solve numerical optimization problems by simulating swarm behaviors found in nature [29]. These techniques demonstrate the desirable properties of efficiency, interpretability, effectiveness, scalability, and robustness. The most popular SI frameworks include Particle Swarm Optimization (PSO) [56], Ant Colony Optimization (ACO) [27], Artificial Bee Colony (ABC) [53] and other swarm optimizations [69, 68, 22, 12]. A full list of swarm intelligence based optimization algorithms can be found in Table 1.1. Every successful swarm intelligence behavior contains two fundamental, sufficient and necessary properties [40]:

1. self-organization: an essential component that no such a swarm is a central coordinator. The interaction between the system and its local level components is not planned, nor through a central authority.

2. division of labor: involving various kinds of division of labor. In a social group, many circumscribed tasks are performed by particular individuals.

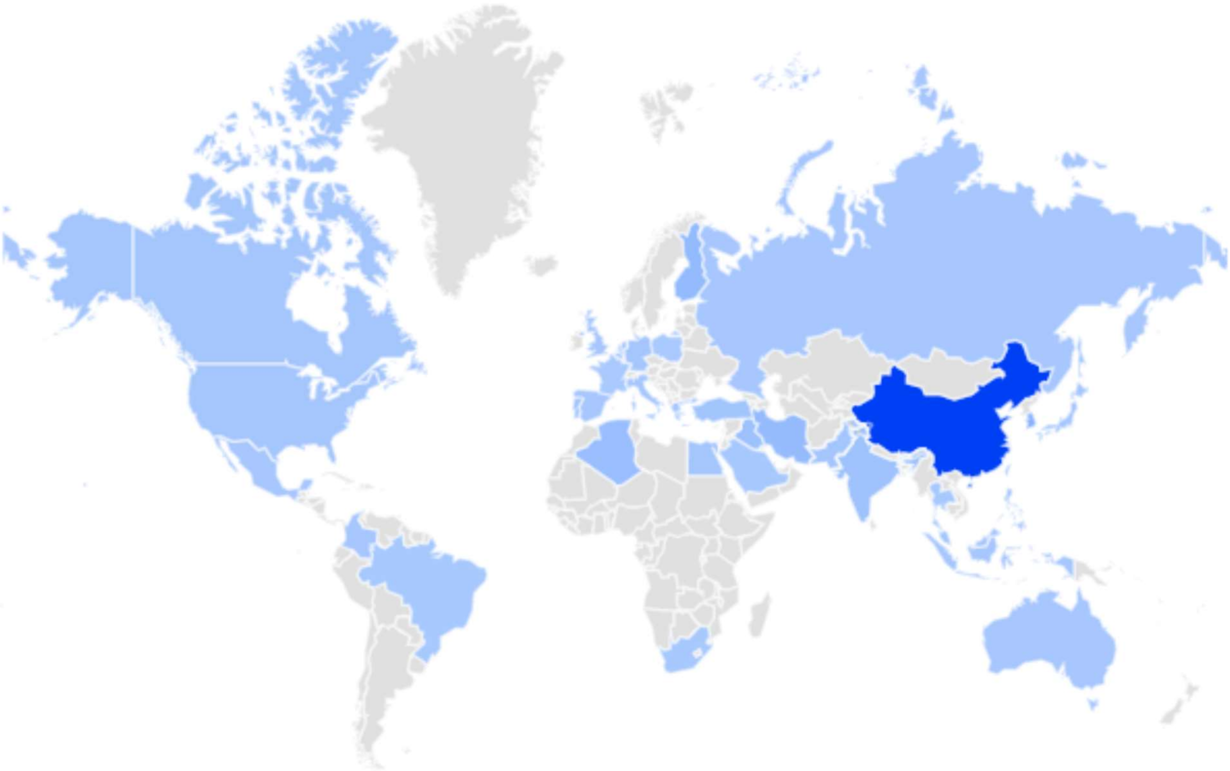


Figure 1.2. Google Trends Search Volume and Geographical Distribution on the Topic of "Swarm Intelligence" in All Languages (Grayscale color is used to colorized the search volume of a particular region, darker color means a higher proportion of all queries)

Table 1.1. Swarm Intelligence based Optimization Algorithm in Literature

SI Algorithm	**Year**	**Reference**
Ant Colony Optimization (ACO)	1991	[27]
Particle Swarm Optimization (PSO)	1995	[56]
Marriage in Honey Bees Optimization Algorithm	2001	[1]
Artificial Fish-Swarm Algorithm	2003	[62]
Termite Algorithm	2005	[85]
Artificial Bee Colony (ABC)	2006	[14]
Wasp Swarm Algorithm	2007	[82]
Wolf Pack Search Algorithm	2007	[108]
Monkey Search	2007	[73]
Bee Collecting Pollen Algorithm	2008	[66]
Cuckoo Search	2009	[111]
Dolphin Partner Optimization	2009	[91]
Firefly Algorithm	2010	[109]
Bat-inspired Algorithm	2010	[110]
Hunting Search	2010	[77]
Bird Mating Optimizer	2012	[10]
Krill Herd	2012	[35]
Fruit Fly Optimization Algorithm	2012	[78]
Dolphin Echolocation	2013	[55]
Social Spider Optimization (SSO)	2013	[22]
Grey Wolf Optimization (GWO)	2014	[69]
Spider Monkey Optimization (SMO)	2014	[12]
Monarch Butterfly Optimization	2015	[100]
Whale Optimization Algorithm (WOA)	2016	[68]
Moth Search Algorithm	2016	[99]

1.2.1 Spider Monkey Algorithm (SMO)

Spider monkey optimization is inspired by the foraging behaviors of spider monkeys. The proposed strategy follows self-organization and division of labor properties for obtaining intelligent swarming behaviors of animals [12]. In the previous studies, the scientists identified four important behaviors [60].

1. Each group selects a local leader in the 'planning' stage and then starts food foraging.
2. Search agents' positions according to the distance between itself to the food source.
3. During the food source searching phase, local leader modernizes its best location within the group.
4. In the final phase, the global leader keeps posted with the so-far best position and in case of inactivity, it splits the group into smaller, finite subgroups.

Algorithm 1 Spider Monkey Optimization

```
Initialize total Population, LocalLeaderLimit $LLL$, GlobalLeaderLimit $GLL$.
Calculate the fitness
Apply greedy selection on global leader and local leader
while termination criteria is not met do
    Generate new positions for all the monkeys
    Select the best position between existing and newly generated ones based on fitness values
    Calculate the probability $prob$ for all the monkeys
    Produce new position for all the monkeys
    Update the positions of local global leaders
    if local leader is not updating her position after a specified number of times $LLL$ then
        Update the positions of all the monkeys
    end if
    if global leader is not updating her position for a specified number of times $GLL$ then
        Divide into small groups and repeat previous steps
    end if
end while
```

1.2.2 Traditional Algorithms

1.2.2.1 Particle Swarm Optimization (PSO)

Particle Swarm Optimization (PSO) is a population-based stochastic algorithm and was first introduced by Kennedy and Eberhart in 1995 [56]. There are many advantages of the traditional PSO algorithm including but not limited to easy interpretation, fast coverage, and fewer adjustable parameters. The particle swarm concept originated as a simulation of the behavior of bird flocking – all birds are randomly distributed and have fitness values that are evaluated by the fitness function to be optimized. They also have velocities which direct the flying of the particles [29]. After reaching the best fitness values, each particle updates its position and velocity as follows,

$$x_i(t) = x_i + v_i(t) \tag{1.7}$$

$$v_i(t+1) = wv_i(t) + r_1C_1(P_i^l(t) + x_i) + r_2C_2(P^g(t) - x_i) \tag{1.8}$$

where x_i and v_i are the positions of a particle and its associated velocity. w and r are the inertia weight and number of $U(1,0)$, respectively. P_i^l indicates the particle best $pBest$ and P^g is the global best $gBest$ of the swarm. C_1 and C_2 are constants and are selected by the user in order to control the efficacy of the method. The procedure of the PSO approach is presented in Algorithm 1.

Algorithm 2 Particle Swarm Optimization

```
Initialize food source, particle best $pBest$ and global best $gBest$
for each particle do
    Initialize particle
end for
while termination criteria is not attained do
    for each particle do
        Calculate fitness value
        if fitness value $< pBest$ then
            Set $pBest =$ fitness value
        end if
    end for
    Choose the particle with $pBest$ of all the particles as the $gBest$
    for each particle do
        Calculate particle velocity according to Eq.1.8
        Update particle position according to Eq.1.7
    end for
end while
```

1.2.2.2 Ant Colony Optimization (ACO)

Have you ever wondered how ants navigate from nest to food source? As we all know, ants are incredibly capable creatures; here are several frightening facts that scientists have discovered about ants,

- Ants are blind and each ant moves at random.
- The path from home to target is discovered through pheromone trails.
- More pheromones on a path increase the probability of being followed and the shortest path is eventually finalized.

Ant Colony Optimization takes inspiration from the foraging behavior of ants for solving computational problems by finding the optimal paths in a graph. The procedure can be broken down into two sections: construct ant solutions and pheromone update [27].

1. Construct Ant Solutions: an ant moves from location i to location j with probability,

$$P(i,j) = \frac{\tau_{i,j}^{\alpha}\eta_{i,j}^{\beta}}{\sum(\tau_{i,j}^{\alpha}\eta_{i,j}^{\beta})} \tag{1.9}$$

   where $\tau_{i,j}$ and $\eta_{i,j}$ are the amount of pheromone and the desirability on a given edge (i,j). The influence of $\tau_{i,j}$ and $\eta_{i,j}$ are controlled by α and β .

2. Pheromone Update: the amount of pheromone is updated according to the equation,

$$\tau(i,j) = (1-\rho)\tau(i,j) + \Delta\tau(i,j) \tag{1.10}$$

   where ρ indicates the rate of pheromone evaporation. $\Delta\tau_(i,j)$ represents the amount of pheromone deposited and is given by,

$$\Delta\tau(i,j)^{k} = \begin{cases} L_k \\ 0 \quad otherwise \end{cases} \tag{1.11}$$

   where L_k is the cost of the trip of the k^{th} ant, if ant k travels on edge i,j. The pseudo-code of ACO method is shown in Algorithm 2.

Algorithm 3 Ant Colony Optimization

```
Initialize population, pheromone trail
while termination criteria is not satisfied do
    for each ant do
        Calculate fitness value
        Determine its best position
    end for
    Determine the global best ant
    Update the pheromone trail according to Eq.1.10 - 1.11
end while
```

1.2.2.3 Artificial Bee Colony (ABC)

In the Artificial Bee Colony (ABC) algorithm, there are three groups of bees: employed bees, onlookers and scouts [53]. An onlooker is a bee waiting at the dance area for decision making to select a food source. An employed bee is a bee that going to the target food source, and a scout is the bee randomly searching for food. Each cycle of the search consists of three major phases:

1. sending the employed bees onto the food sources and then measuring their nectar amounts;
2. selecting of the food sources by the onlookers after sharing the information of employed bees and determining the nectar amount of the foods;
3. determining the scout bees and then sending them onto possible food sources.

Algorithm 4 Artificial Bee Colony

```
Initialize food source
while termination criteria is not met do
    for each employed bee do
        Produce new solution
        Calculate the fitness value
        Apply greedy selection
        Calculate the probability value
    end for
    for each onlooker bee do
        Select a solution
        Produce new solution
        Calculate the fitness value
        Apply greedy selection
    end for
    if an abandoned solution for the scout exists then
        Replace it with a new solution at random
    end if
    Register the best solution
end while
```

1.2.3 Other Algorithms

1.2.3.1 Grey Wolf Optimization (GWO)

Grey wolf optimization (GWO) is first introduced by Mirjalili et al. [69], and it mimics the leadership hierarchy and hunting mechanism of grey wolves in nature. Group hunting is an interesting social behavior of grey wolves. The major phases of grey wolf hunting are searching and chasing the prey, encircling and harassing the prey and attacking prey.

$$\vec{D}_\alpha = |\vec{C}_1 * \vec{X}_\alpha - \vec{X}|, \vec{D}_\beta = |\vec{C}_2 * \vec{X}_\beta - \vec{X}|, \vec{D}_\delta = |\vec{C}_3 * \vec{X}_\delta - \vec{X}| \tag{1.12}$$

$$\vec{X}_1 = \vec{X}_\alpha - \vec{A}_1 * (\vec{D}_\alpha), \vec{X}_2 = \vec{X}_\beta - \vec{A}_2 * (\vec{D}_\beta), \vec{X}_3 = \vec{X}_\delta - \vec{A}_3 * (\vec{D}_\delta) \tag{1.13}$$

$$\vec{X}_{t+1} = \frac{\vec{X}_1 + \vec{X}_2 + \vec{X}_3}{3} \tag{1.14}$$

where t indicates the current iteration and $\vec{X}_{t+1}$ indicates the position vector of a grey wolf. The social hierarchy, searching, encircling, and attacking prey are mathematically modeled by following the steps below:

Algorithm 5 Grey Wolf Optimization

```
Initialize the grey wolves population $X_i (i = 1, 2, ..., n)$
Initialize a, A and C
Calculate the fitness of each search agent
$X_a$ = the best search agent
$X_\beta$ = the second best search agent
$X_\delta$ = the third best search agent
while $t <$ Max number of iterations do
    for each search agent do
        Update the position of the current search agent by Eq.1.14
    end for
    Update a, A, and C
    Calculate the fitness of all search agents
    Update $X_a, X_\beta$ and $X_\delta$
    $t = t + 1$
end while
return $X_a$
```

1.2.3.2 Whale Optimization Algorithm (WOA)

Whale optimization algorithm (WOA) is a novel meta-heuristic optimization algorithm that mimics the social behavior of humpback whales which are known as highly intelligent animals with emotion [68]. The unique hunting method of humpback whales is called bubble-net feeding method, see Figure 1.3 [104], the whales dive around 12m down and then start to create bubble in a spiral shape around the prey and swim up toward the surface. The bubble-net feeding method is mathematically modeled below in order to perform optimization. The WOA algorithm consists 3 major phases:

1. Encircling prey - this behavior is represented by the following equations:

$$\vec{D} = |\vec{C} * \vec{X}_t^* - \vec{X}_t| \tag{1.15}$$

   where t indicates the current iteration, $\vec{C}$ is the coefficient vector, $\vec{X}^*$ is the position vector of the best solution achieved so far and $\vec{X}$ is the position vector.

2. Bubble-net attacking method - as seen in Figure , this approach calculates the distance between the whale and prey as follows:

$$\vec{X}_{t+1} = \vec{D}' * e^{bl} * \cos 2\pi + \vec{X}_t^* \tag{1.16}$$

   where $\vec{D}'$ indicates the distance of the whale to the prey which is the best solution achieved so far. b and l are constant number to define the shape of the logarithmic spiral and a random number in $[-1, 1]$, respectively.

3. Search for prey

$$\vec{D} = |\vec{C} * \vec{X}_t^* - \vec{X}_t| \tag{1.17}$$

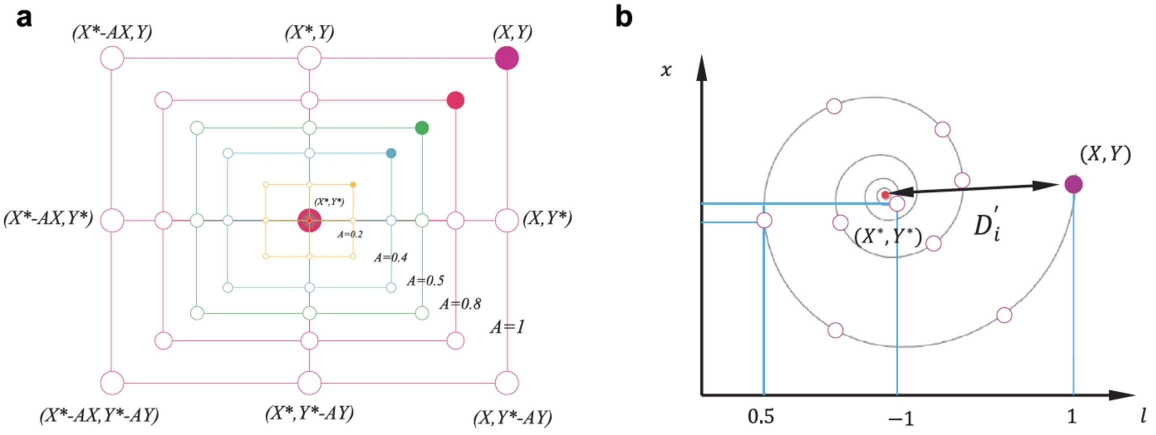


Figure 1.3. Bubble-net Search Mechanism of Humpback Whales [68].

1.2.3.3 Social Spider Optimization (SSO)

Social spider optimization algorithm is based on the simulation of cooperative behavior of social-spiders [22]. SSO considers two types of search agents: male and female spiders. Each agent, according to its gender, cooperate in different activities such as building and maintaining the communal web, prey capturing, mating and social contact [113]. Figure 1.4 shows the schematic representation of the SSO algorithm-data-flow of the female cooperative and male cooperative operators; the mating operator modifies both individual types.

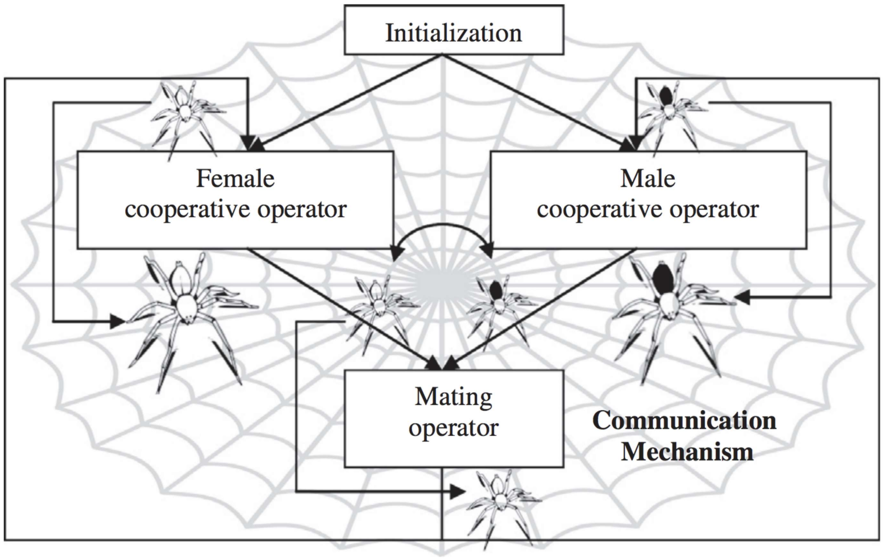


Figure 1.4. Schematic representation of the SSO algorithm-data-flow [22].

Algorithm 6 Social Spider Optimization

```
Consider total N number of male $N_m$ and female $N_f$ spiders
Initialize the female and male spiders randomly
Calculate the radius of mating
while termination criteria is not satisfied do
    Calculate the weight of every spider
    Move female spiders according to the female cooperative operator
    Move male spiders according to the male cooperative operator
    Perform the mating operation
end while
```

1.3 Motivation, Goal and Contribution

Current methodologies in prediction and optimization are good, but because the limitations we discussed, not easy to interpret or lack of robustness. Motivated by these reasons we present a pair-wise feature selection approach, Top Discriminating Pairs, to handle large dimension cancer classification/prediction problems. Moreover, we adopt the Swarm Intelligence mechanism to In this subsection we briefly resume the major contributions of this dissertation.

Specifically, in the following chapters we show,

1. Rule-based approaches are a good way of representing of knowledge or rich information. We present a new method in machine learning that improves the prediction accuracy with less number of features involved while still maintaining robustness and interpretability.

2. We prove that swarm intelligence based algorithms are able to avoid local minimums and search for global optimal solution more efficiently.

3. We construct spider monkey optimization based routing protocol that advances wireless sensor networks mechanism in the direction of optimal solution.

4. We adopt spider monkey optimization algorithm that advances feature space reduction in classification and prediction.

5. We provide evidence from real-world applications that our methods provide significant advantages in accuracy and interpretability.

1.4 Structure of the Dissertation

A chapter by chapter description of this dissertation is as follows.

Chapter 2: Introduces the rule-based feature selection approach, Top Discriminating Pairs, to handle a high number of attributes in gene expression analysis and cancer classification problems.

Chapter 3: Applies spider monkey optimization algorithm to address 11 numerical optimization problems; and also carries out the analysis of performance with comparison to some other well-established optimization algorithms.

Chapter 4: Constructs a spider monkey optimization based routing protocol in wireless sensor networks to improve network performance and reduce energy consumption.

Chapter 5: Provides a real-world example of using spider monkey optimization algorithm as a feature selector to elevate cancerous genes in Microarray analysis.

Chapter 6: Summarizes the main points and key messages of the paper and includes some closing thoughts for future work.

CHAPTER 2

TOP DISCRIMINATING PAIRS FOR FEATURE SELECTION

2.1 Design Challenges

Rule-based approaches are a good way of representing of knowledge or rich information. Moreover, rule-based feature selection techniques or classifiers are popular in various of machine learning applications due to their easy interpretability and robustness in handling high number of attributes. However, major concerns in rule-based learning approaches are expression of general knowledge, expression of specialized knowledge, naturalness, modularity, knowledge acquisition, unexpected missing inputs, inference efficiency, maintenance, updatability, provision of explanation, etc. [83]. A detailed explanation on some of these factors is provided as below:

1. Maintenance: Complex validation methods are required to to maintain redundant and conflicting rules. The maintenance process gets difficult as the size of the rule base increases.

2. Modularity: Each rule is independent and can be inserted into or removed to the knowledge base without affecting other rules.

3. Naturalness: Rules are a fundamental method for representing natural knowledge. It can be applied in many application domain by emulating human expertise in greater depth.

4. Knowledge Acquisition: acquiring rules or knowledge through experts is very time-consuming. For certain domain area, the knowledge is very complicated and may require a large number of rules.

5. Interpretation: The general nature of rules may create problems in the interpretation of their scope during reasoning. To effectively deal with a specific situation, rules may sometimes need to be specialized [71].

6. Unexpected Missing Inputs: data quality is a major factor to draw conclusions from rules. For a specific rule, a certain number of condition values must be known in order to evaluate the logical function connecting its conditions.

2.2 Our Methodology

For generality, we describe the method in terms of marker pairs, which represent the most informative paired genes. Consider a training dataset of M genes whose expression levels can be assigned as $\{X_1.., X_m\}$, and total N samples $\{1, ..., N\}$. The data can be represented as a matrix of M x N dimension in which the expression value of the i^{th} gene, $i = \{1, ..., M\}$, form the n^{th} sample is denoted by X_{in}. Let $(y_1, ..., y_n)$ be the vector of class labels, where $y_n \in C$ and $C = \{C_1, ..., C_t\}$. In binary classification, we assume t = 2, where C_1 refers to normal samples (good prognosis group) and C_2 to cancerous samples (poor prognosis group). The structure of paired genes are shown as below,

$$X_{i,n} = \begin{bmatrix} X_{1,1} & X_{1,2} & \cdots & X_{1,n} \\ X_{1,1} & X_{1,1} & \cdots & X_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ X_{m,1} & X_{m,1} & \cdots & X_{m,n} \end{bmatrix} \tag{2.1}$$

For each gene expression value, we define labeling rules based on two conditions first. If the expression value of X_{in} is less than or equal to the mean value of the i^{th} gene across all samples, then we label X_{in} as Low, represented by symbol L. Otherwise, X_{in} is High, represented by symbol H, respectively. The comparison rules for the gene pair (i, j),

$$R_{ij} = \begin{cases} LL, & X_{in} \leq \bar{x_i} \quad X_{jn} \leq \bar{x_j} \\ LH, & X_{in} \leq \bar{x_i} \quad X_{jn} > \bar{x_j} \\ HL, & X_{in} > \bar{x_i} \quad X_{jn} \leq \bar{x_j} \\ HH, & X_{in} > \bar{x_i} \quad X_{jn} > \bar{x_j} \end{cases} \tag{2.2}$$

the quantities of interest in class C_t are,

$$p_{ij}(R_{ij}, C_t) = \begin{cases} Prob(LL|C_t) \\ Prob(LH|C_t) \\ Prob(HL|C_t) \\ Prob(HH|C_t) \end{cases}, t = (1, 2) \tag{2.3}$$

these probabilities are estimated by the relative frequencies of occurrences of each comparison rule within expression profiles and over samples. For selecting the highly ranked gene pairs, we first compute by calculating the difference for every single rule among all classes, C_1 and C_2, across all samples. Hence, the 'rule difference' γ_{ij} in class C_t, defined as,

$$\gamma_{ij}(R_{ij}) = |p_{ij}(R_{ij}, C_{t=1}) - p_{ij}(R_{ij}, C_{t=2})| \tag{2.4}$$

then eliminate the rules with less conditional probability. Let $\triangle_{ij}$ denotes the 'ranking score' for the gene pair (i, j),

$$\Delta_{ij} = max(\gamma_{ij}(R_{ij})) \tag{2.5}$$

based on $\triangle_{ij}$ scores, we are able to select the top discriminating pairs. These pair of gene are viewed as a subset of relevant features for further use in classification.

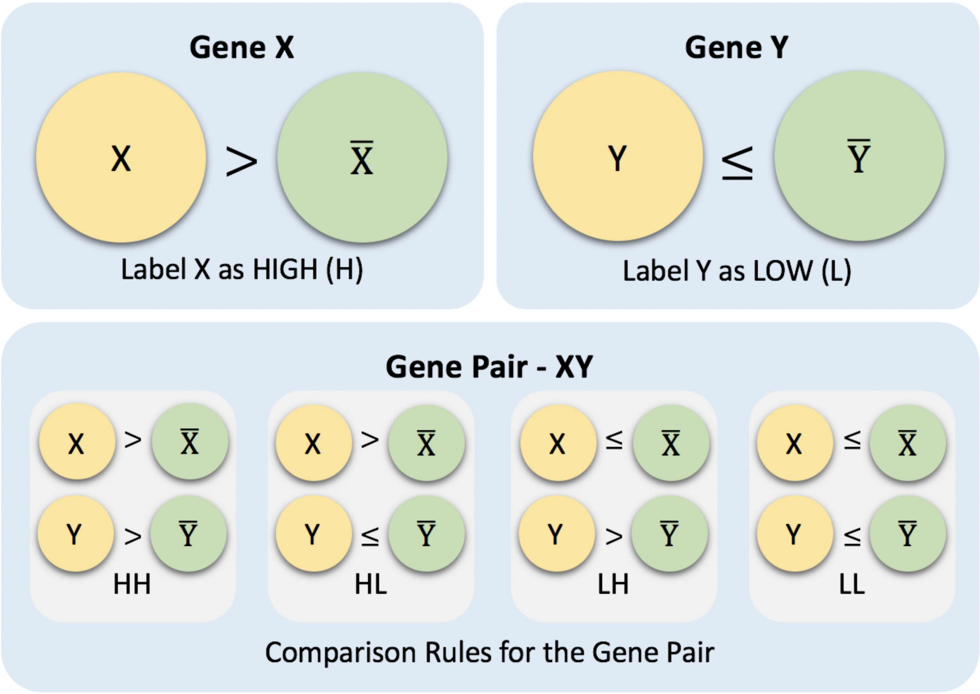


Figure 2.1. Top Discriminating Pairs

2.3 Results and Discussion

2.3.1 Simulated Data

To illustrate how feature selection methods respond to different data structure, we apply our method on a type of artificial data as follows: each sample contains 1000 genes, of which 100 are signal genes. This type of data was first generated by [90], which the signal gene follow the multivariate normal distribution $N(\mu, E)$ and $N(-\mu, E)$ for class 1 and class 2, respectively. Here, u is a vector of 10 distinct values ranging in [-0.25, 0.25] with an increment of 0.05 or 0.1. Each value is being the effect size of 10 differentially expression genes [90]. The rest of 900 genes follow standard normal distribution with mean 0 and standard deviation 1. Total 150 independent samples were generated, and each class consists of 75 samples.

First, we compare the performance of Random Forests, J48 and Naïve Bayes on simulated data, using TDP with the traditional TSP (without dimensionality reduction step) as feature ranking methods. We apply discretization methods on continuous model, transferring continuous value to discrete representations. Based on our previous experiments, three sigma edit rule approach achieved better performance to others. Hence, in the rest of our simulation experiments, we will apply three sigma rule only in discretization step. To better investigate the robustness of each method, we start with dimensionality reduction preprocessing step by removing the redundant features in each experiment; then we apply the TDP feature selection algorithm, build Random Forests, J48 and Naïve Bayes models with each level of selected genes (k) on the training set. The training set contains 100 samples (from the simulated data) and the remaining 50 samples are used as test set. The statistics of simulated data are listed in Table 2.1.

Table 2.1. Statistics of Simulated Data

Genes	Samples (+/-)	Training size	Test size	Classes
1000	150 (75/75)	100	50	2

Note: positive sample (+), negative sample (-).

Table 2.2 shows the misclassification rate in simulated data at different level of selected gene pairs using Random Forests classifiers, with correlation coefficients ρ at 0 and 0.45. The performance illustrates that our TDP method seems to perform comparably to the traditional TSP when the signal genes are independent. However, as the selection level of gene pairs is low and the signal genes become more correlated, TDP turns out to be increasingly advantageous over TSP.

Table 2.2. Misclassification Rates of TSP and TDP Methods on Simulated Data

Selection Level	$\rho = 0$		$\rho = 0.45$	
	TSP	TDP	TSP	TDP
1	0.47	0.39	0.32	0.31
10	0.43	0.41	0.38	0.36
20	0.43	0.42	0.34	0.35
30	0.38	0.35	0.20	0.30
40	0.41	0.39	0.24	0.29
50	0.35	0.32	0.27	0.33

Our TDP as a feature selection method is effective than TSP in response to the progressively increased correlation among signal genes when selecting small number of gene pairs (selection level = 10 or 20). This trend is illustrated in Table 2.2. Furthermore, we exhibit the running time of the TSP and TDP methods to demonstrate the computational efficiency, shown in Table 2.3. As can be seen, TDP outperforms the state-of-the-art TSP algorithm in all conditions, in varies number of samples and features. Hence, our TDP approach can be used to efficiently generate rules and eliminate features in a shorter execution time than TSP in most of the cases.

Table 2.3. Comparison of Various Feature Selection Methods on Running Time

Selection Level	$\rho = 0$	
	TSP	TDP
100x50	0.044	0.020
500x50	0.333	0.267
1000x50	2.069	0.414
100x150	0.051	0.031
500x150	0.622	0.337
1000x150	3.048	1.086

Note: each subset is extracted directly from our 1000 x 150 simulated data, when $\rho = 0$.

2.3.2 Microarray Data

2.3.2.1 Microarray Data Characteristics

In general, researchers are facing various types of challenges when dealing with Microarray data analysis. In our study, we mainly focus on binary microarray datasets which consist of healthy patient samples and diseased samples. The three major characteristics of the microarray data we used in our experiments are,

1. Small sample size: most of the DNA microarray data has less than 100 instances, using a very small size of samples could increases the chance of assuming as positive or negative class [32].

2. Class imbalance: this problem happens when data dominated by a major class which contain more samples than the other classe [34].

3. Data complexity: some of the data are more complicated to separate because of overlapping among classes or the linearity of the decision boundaries. It is a commonly seen issue in microarray data [48].

2.3.2.2 Experimental Results

We applied the TDP and TSP methods to two cancer datasets, and the information of these datasets is summarized in Table 2.4. The pre-processing procedure is similar to the experiments in simulated data, which includes discretization using three sigma rule and dimensionality reduction with closed BiMAX method. The first data set is breast cancer dataset [101], obtained from Shi et al. [90], which is normalized by RMA procedures using Bioconductor packages. The final expression data comprising 209 samples and 22283 genes. The dataset consists of 71 patients who developed distant metastases or dies within 5 years labeled as negative samples, and the rest 138 patients who remained healthy during the same time frame classified as positive samples. Another cancer dataset is derived from Beer et al. [15], obtained from the cancer dataset depository of the Broad Institute. This data contains 86 patients with primary lung adenocarcinoma, which 24 patients who had died and the remaining 62 of them.

Table 2.4. Statistics of Data Sets

Data	Genes	Samples (+/-)	Classes	Source
Wang Breast Cancer	22283	209 (138/71)	2	[101]
Lung Adenocarcinoma	7129	86 (62/24)	2	[15]

Note: positive sample (+), negative sample (-).

The classification performance of TDP is compared with TSP as feature selection methods in the two cancer datasets. We performed 10-fold cross-validation on breast cancer data and 5-fold cross-validation on lung adenocarcinoma data, then average the results from five experiments. The results summarized in Table 2.5 using different methods on the above datasets. The three classification algorithms we selected are Random Forests (RF), J48 and Naïve Bayes (NB). Random Forests and J48 algorithms are tree-based methods. NB is a simple yet powerful algorithm; it provides classification outcomes as well as the degree of certainty [18]. Table 2.5 shows our proposed TDP method outperforms TSP in most of the cases, especially when the feature selection level is small. However, the selection level is not the only evaluation metric in feature selection, we also provide the clssification accuracy details of each algorithm in this table. Overall, TDP approach has slightly better results on both Wang Breast Cancer and Lung Adenocarcinoma data sets when 1) the sample size is small and 2) the classes are imbalanced.

The Figure 2.2 illustrates the average accuracy of RF, J48 and NB classifiers on Wang Breast Cancer data. The horizontal axis represents the number of gene pairs (selection level) and the vertical axis represents the out-of-sample classification accuracy is obtained among all the runs. Similarly Figure 2.3 shows the results on Lung Adenocarcinoma data with various classifiers.

Table 2.5. Accuracy of TSP and TDP Methods on Various Data Sets

Selection Level	Random Forests (%)		J48 (%)		Naïve Bayes (%)	
	TSP	TDP	TSP	TDP	TSP	TDP
Wang Breast Cancer						
1	70.19	65.57	65.38	68.27	63.46	71.15
10	65.38	70.96	64.81	69.23	66.35	63.46
20	69.04	70.58	54.81	64.42	64.42	59.62
30	72.11	71.19	57.31	58.63	62.44	59.62
40	71.47	74.04	60.18	56.73	64.42	58.65
50	69.21	71.15	63.46	67.30	66.34	71.15
Average	69.57	**70.58**	60.99	**64.10**	**64.57**	63.94
Lung Adenocarcinoma						
1	60.46	70.46	71.33	69.78	62.79	69.74
10	65.38	70.96	61.84	69.23	66.35	63.46
20	69.04	70.58	54.81	64.42	64.42	61.54
30	69.76	66.86	65.11	58.13	58.14	63.13
40	70.45	67.65	69.07	65.12	55.81	62.79
50	69.90	72.09	60.46	68.13	67.44	60.46
Average	67.50	**69.77**	63.77	**65.80**	62.49	**63.52**

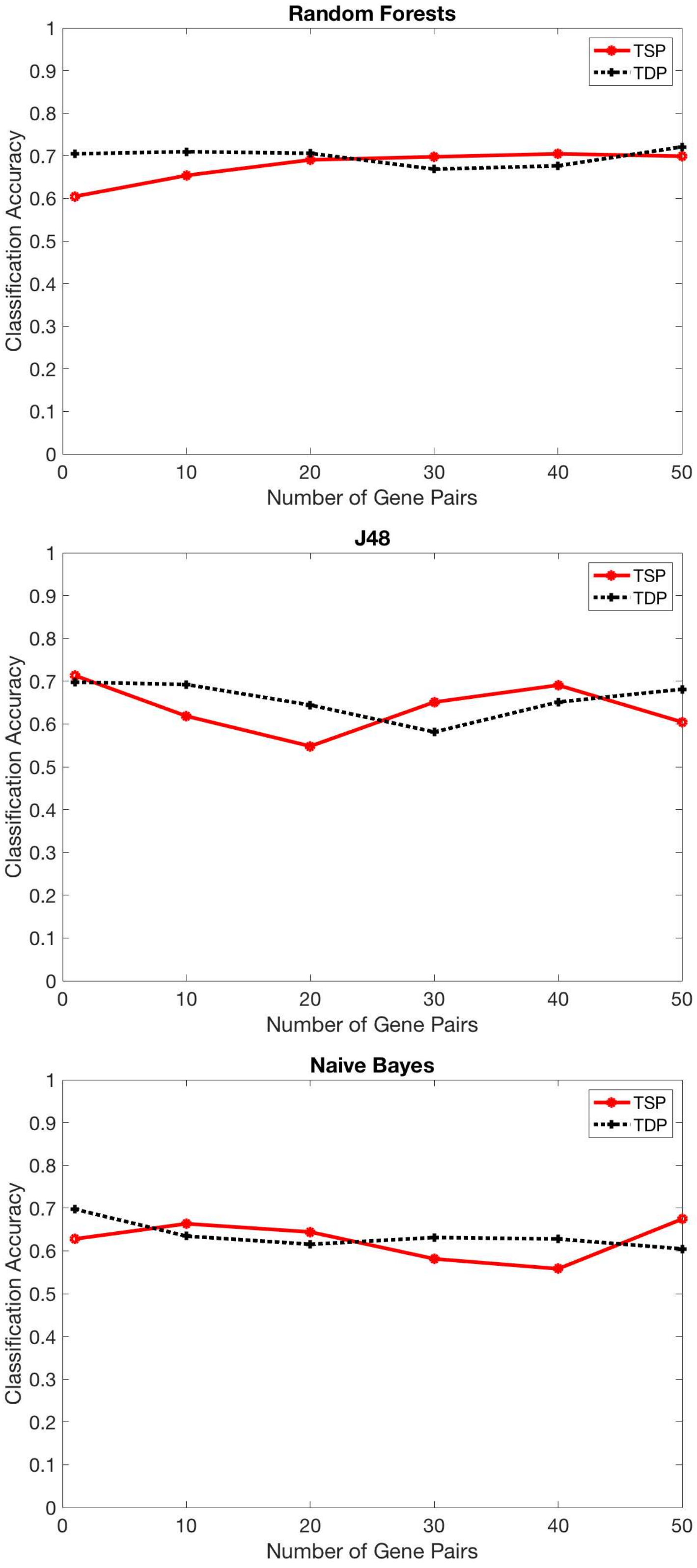


Figure 2.2. Accuracy of TDP and TSP on Wang Breast Cancer Data

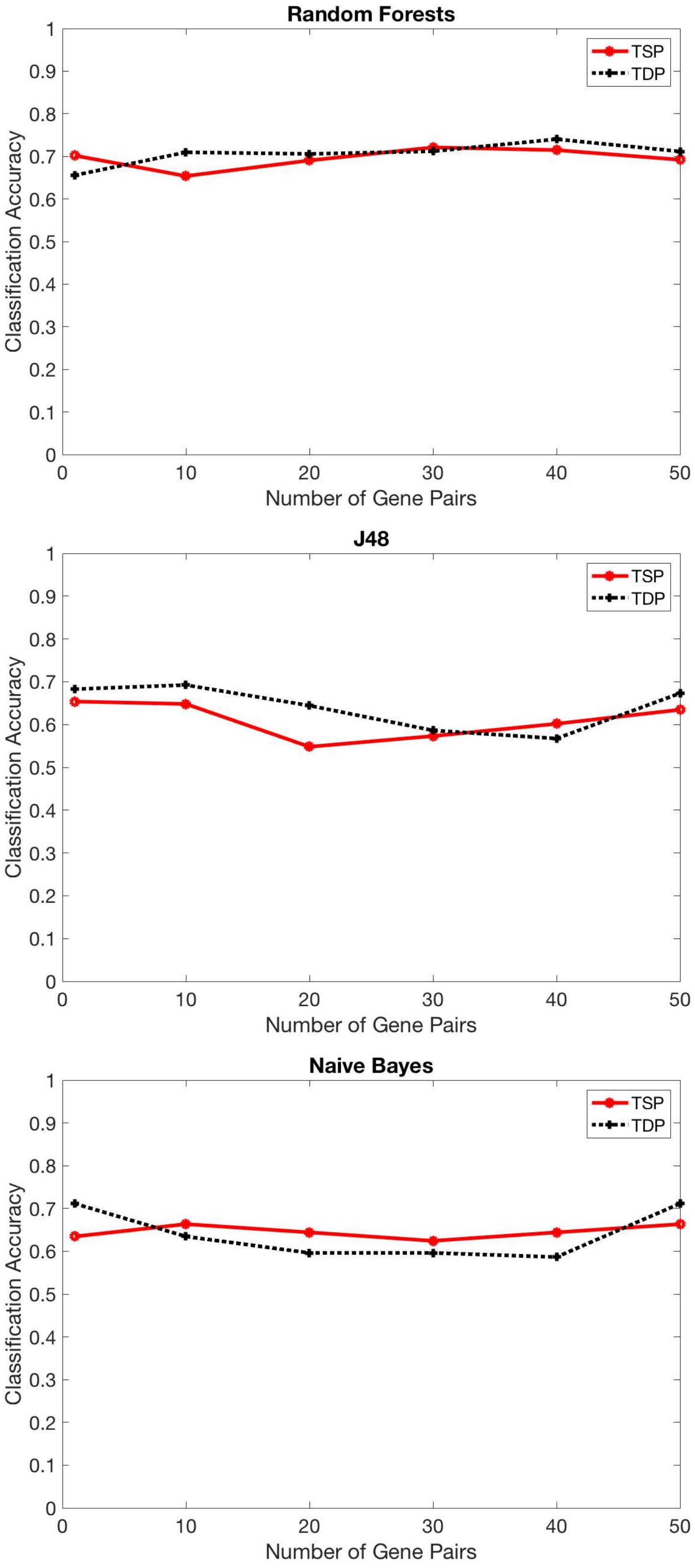


Figure 2.3. Accuracy of TDP and TSP on Lung Adenocarcinoma Data

CHAPTER 3

SPIDER MONKEY OPTIMIZATION IN NUMERICAL OPTIMIZATION

Spider monkey optimization (SMO) is a relative new addition to the family of swarm intelligence algorithms by structuring the social foraging behavior of spider monkeys. To illustrate the robustness of the introduced SI based algorithms, we tested them over well known optimization test problems as well as some popular real world optimization problems. We also carry out the Sensitivity analysis of different parameters, statistical analysis of results with comparison to some other well established optimization algorithms.

3.1 Benchmark Function

The benchmark functions are also known as the test functions or artificial landscapes. It's used to evaluate characteristics of optimization algorithms such as overall performance, robustness, precision and convergence rate [11]. In order to evaluate an algorithm, one must be tested or identify the kind of problems where it outperforms the others. In general, a wide variety of problems should be taken in to consideration, such as unimodal, multimodal, separable, non-separable, regular, irregular and multi-dimensional problems [19]. In this study, we only consider the test functions are formulated by unimodal, multimodal, separable and non-separable problems.

3.2 Comparison Results

The numerical efficiency of the PSO, ABC, ACO, GWO, WOA, SSO and SMO algorithms introduced in this study was tested by solving 11 mathematical optimization problems. Table 3.1 summarizes the test problems reporting the objective function, the number of design variables, range of variation of optimization variables, function type and the optimum

Table 3.1. Description of Benchmark Functions

Test Problem	**Objective Function**	**D**	**Range**	**Type**	**Optimum**
Ackley	$f_1(x) = 20 - 20exp(-0.2\sqrt{\frac{1}{D}\sum_{i=1}^{D} x_i^2}) + e - exp(\frac{1}{D}\sum_{i=1}^{D}\cos(2\pi x_i)) + 20 + exp$	30	[-32, 32]	MS	0
Axis Parallel Hyper-Ellipsoid	$f_2(x) = \sum_{i=1}^{D} ix_i^2$	30	[-5.12, 5.12]	US	0
Cigar	$f_3(x) = x_0^2 + 10^6\sum_{i=1}^{D} ix_i^2$	30	[-10, 10]	US	0
Dixon and Price	$f_4(x) = (x_1 - 1)^2 + \sum_{i=2}^{D} i(2x_i^2 - x_{i-1})^2$	30	[-10, 10]	US	0
Griewank	$f_5(x) = \frac{1}{4000}\sum_{i=1}^{D} x_i^2 - \prod_{i=1}^{D}(\frac{x_i}{\sqrt{i}}) + 1$	30	[-600, 600]	MN	0
Rastrigin	$f_6(x) = \sum_{i=1}^{D}(x_i^2 - 10\cos 2\pi x_i)^2 + 10)$	30	[-5.12, 5.12]	US	0
Salomon	$f_7(x) = -\cos(2\pi\sqrt{\sum_{i=1}^{D} x_i^2}) + 0.1\sqrt{\sum_{i=1}^{D} x_i^2 + 1}$	30	[-100, 100]	US	0
Schwefel	$f_8(x) = \sum_{i=1}^{D} x_i \sin\sqrt{\lvert x_i\rvert}$	30	[-500, 500]	MS	-418.9829
Sphere	$f_9(x) = \sum_{i=1}^{D} x_i^2$	30	[-5.12, 5.12]	US	0
Step	$f_{10}(x) = \sum_{i=1}^{D}(x_i + 0.5)^2$	30	[-100, 100]	US	0
Zakharov	$f_{11}(x) = \sum_{i=1}^{D} x_i^2 + (\sum_{i=1}^{D} 0.5ix_i)^2 + (\sum_{i=1}^{D} 0.5ix_i)^4$	30	[-5, 10]	US	0

Note: U - Unimodal, M - Multimodal, S - Separable, N - Non-separable value quoted in literature. The experiment compares the convergence curves of various algorithms proposed in the previous subsection. For each benchmark function, the algorithms ran 30 times starting from different populations randomly generated. The results are reported in Table 3.2 considering the following performance metrics: the averaged best solution (AB) and corresponding standard deviation (SD). In all comparisons, the swarm population has been set to 30 individuals and the maximum iteration number is 100. According to this table, GWO and SMO deliver better results than other algorithms. In particular, SMO outperforms in f_1, f_4, f_5, and f_6 which cover the multimodal-separable, unimodal-separable and multimodal-non-separable types of benchmark functions.

The parameter setting for each algorithm in the comparison is described as follows:

1. PSO: the personal learning coefficient is 1.5 and the global learning coefficient is 2 [46].

2. ABC: the abandonment limit parameter is around 540 and the acceleration coefficient upper bound is 1 [46].

3. ACO: the intensification factor (selection pressure) sets at 0.5 and the deviation-distance ratio is 1 [46].

4. GWO: the vector a linearly decreases from 2 to 0.

5. WOA: the vector a linearly decreases from 2 to 0 and the vector a_2 linearly decreases from -1 to -2.

6. SSO: the probability PF has been set to 0.7.

7. SMO: the parameters maximum group size, local leader limit and global leader limit are 5, 50, and 150, respectively.

Evolutionary algorithms (EA) have been widely employed for solving complex optimization problems. These methods are found to be more powerful than conventional methods based on formal logics or mathematical programming. PSO, ABC and ACO are the most

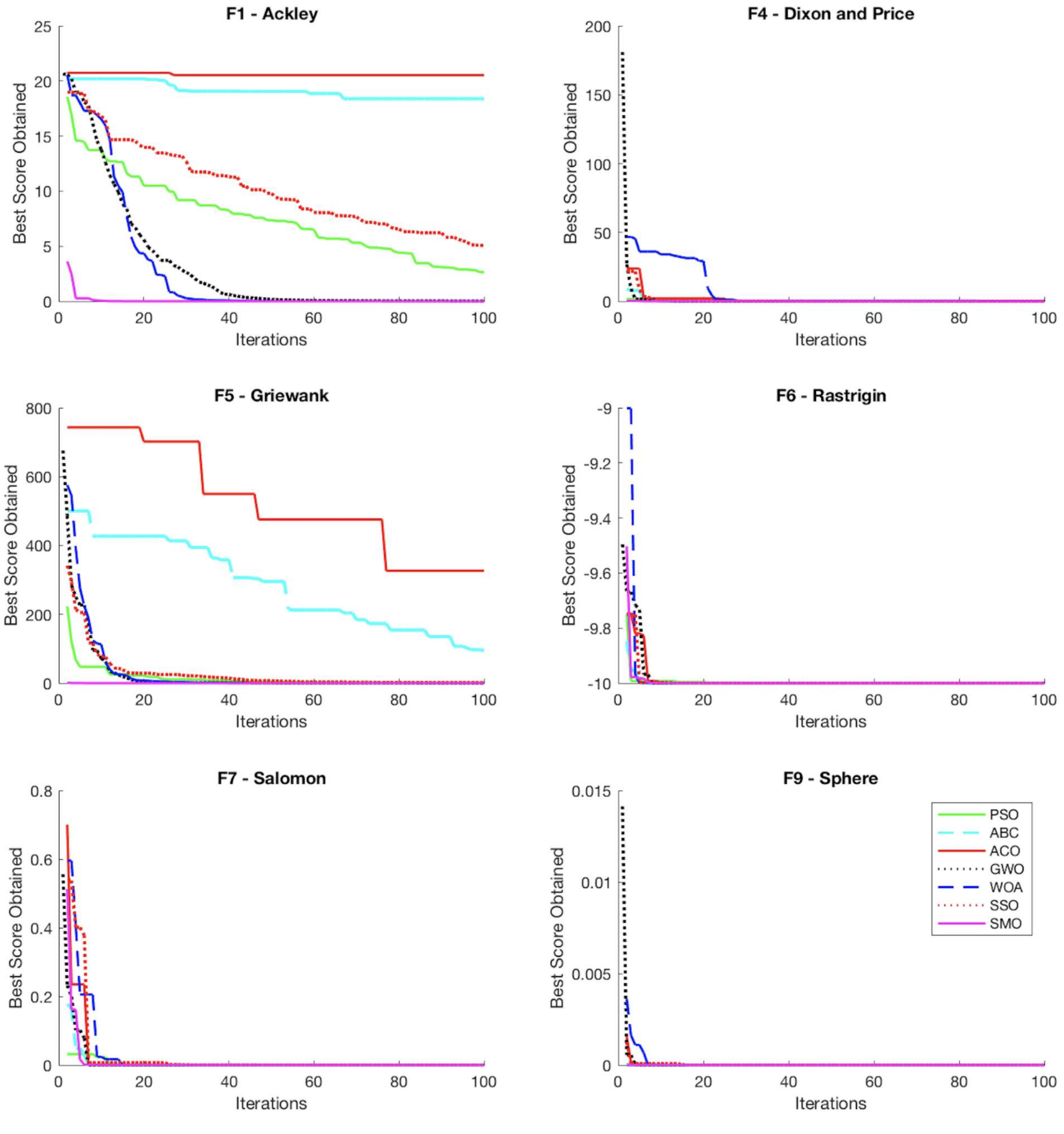


Figure 3.1. Comparison of Convergence Curves of the Proposed Algorithms Obtained in some of the Benchmark Problems

Table 3.2. Comparison of Benchmark Functions

		PSO	**ABC**	**ACO**	**GWO**	**WOA**	**SSO**	**SMO**
f_1	**AB**	10.9733	20.3252	20.5427	0.3287	4.8638e-07	6.9581	**4.8991e-09**
	SD	3.1455	0.4181	0.0987	2.5093	6.5164	3.5003	**1.0819**
f_2	**AB**	0.0047	1.7042e-04	0.0074	**1.8123e-47**	4.4848e-42	1.1112e-07	5.3671e-07
	SD	0.0184	0.0011	0.0364	**3.9399e-06**	0.0339	0.0113	0.6064
f_3	**AB**	0.0035	0.0032	8.2003e-06	**5.9393e-63**	2.7399e-37	1.8590e-06	2.8135e-04
	SD	0.0166	0.0156	2.3823e-05	**0.3991**	0.1295	0.0399	0.0012
f_4	**AB**	1.4024	0.6452	3.5482	1.2479e-04	1.6944e-04	0.0015	**2.5473e-10**
	SD	9.7965	3.8256	21.3787	2.4476	21.9963	4.3241	**0.1450**
f_5	**AB**	40.5959	397.1321	445.8961	0.0023	0.5955	16.2660	**4.8966e-07**
	SD	67.5638	104.9550	106.2475	0.3619	114.0921	97.0855	**0.2198**
f_6	**AB**	-9.9573	-9.9742	-9.9703	-10	-10	-10	**-10**
	SD	0.1565	0.0665	0.1846	8.0662e-04	0.0536	0.0847	**4.8670e-04**
f_7	**AB**	0.0501	0.0448	0.0178	**1.2808e-29**	1.7509e-21	2.5052e-05	1.3040e-05
	SD	0.1282	0.1321	0.0534	**0.0282**	0.0518	0.2138	0.0635
f_8	**AB**	1.2151e+04	-4.8405e+08	-4.9867e+08	1.2566e+04	1.2151e+04	**1.2151e+04**	1.2151e+04
	SD	1.2497	1.1125e+09	1.3627e+09	0.0194	9.9265	**0.0317**	0.8129
f_9	**AB**	3.1232e-05	4.8829e-07	2.5748e-05	**3.2140e-59**	8.9872e-43	3.7040e-09	1.7890e-08
	SD	1.0789e-04	1.9657e-06	1.7850e-04	**1.5544e-04**	0.0042	3.7591e-04	0.0202
f_{10}	**AB**	6.2662e-05	0.0223	0.0111	3.0310e-09	**2.7851e-15**	3.9362e-08	2.0101e-04
	SD	2.0383e-04	0.1453	0.0530	0.0055	**0.2198**	0.0080	0.6533
f_{11}	**AB**	0.3072	3.0058	0.4601	**9.0292e-58**	4.8576e-41	5.4788e-07	1.4601e-09
	SD	1.3654	21.2511	2.2536	**1.6224**	0.1004	4.0531	32.3139

popular swarm algorithms for solving complex optimization problems. However, they present serious flaws such as premature convergence and difficulty to overcome local minima [102]. Different to them, the newly invested nature-inspired algorithms such as SSO and SMO take gender information into consideration, allows incorporating computational mechanisms to avoid critical flaws such as premature convergence and incorrect exploration–exploitation balance commonly present in PSO, ABC and ACO algorithms [22, 12].

CHAPTER 4

SPIDER MONKEY OPTIMIZATION FOR FEATURE SELECTION

4.1 Design Challenges

Swarm Intelligence indicates a recent computational and behavioral metaphor for solving distributed problems that originally took its inspiration from the biological examples provided by social insects (ants, termites, bees, wasps) and by swarming, flocking, herding behaviors in vertebrates. Some of the major factors of design challenges in swarm intelligence are self-organization, collective intelligence, scalability, stability, data sources, etc.. We provide a detailed explanation on each factor in the following section:

1. Scalability: Large-scale gene expression data requires superior computing power. It cannot be loaded directly into the memory and limits the usage of various feature selection algorithms [61].

2. Stability: In the field of bioinformatics, the stability of a new feature selection algorithm is important that a similar set of genes should be selected each time when obtaining new samples in the small amount of perturbation [44].

3. Structured Feature: Most generic data contain the features that do not have explicit correlation. Current feature selection algorithms may propose the same subset of the features even though the features are shuffled [112].

4. Multi-Source Data: Traditional feature selections are designed to address the problems with a single data source. However, the heterogeneous data brings more extraordinary insights, leverages their associations and characteristics, and finds more correlated features [116].

5. Multi-View Data: Many problems in data mining and machine learning involve datasets with multiple views. Multi-view data represents different facets of data instances through multiple feature spaces [67].

6. Global and Local Minimum: Feature selection often leads to non-convex optimization problem and most of the existing approaches that address this issue can only guarantee the convergence to a local minimum[92].

4.2 Related Work

4.2.1 PSO for Feature Selection

Xi et al. [107] propose a binary encoded quantum-behaved Particle Swarm Optimization (BQPSO) method for gene selection combining support vector machine (SVM) for human cancer classification. It is initially inspired by the quantum theory; it considers the position of a particle as a binary string. For example, p1(1011001010) and p2(0010010110) are two particles; the distance between these two particles is calculated based on Hamming distance. This hybrid method for gene selection and cancer classification of high dimensional Microarray data, five cancerous datasets outperforms genetic algorithm (GA) with SVM and has shown great significance in robustness, efficiency and accuracy. The authors of the method believe that the proposed BQPSO/SVM approach has an obvious advantage in term of very few genes are involved in classification and it provides strong search capability.

Gao et al. [36] present a hybrid optimized classifier, PA-SVM, combing Particle Swarm Optimization (PSO) and Artificial Bee Colony (ABC) algorithms on gene expression data after filtering out redundant features using Fast Correlation-Based Feature Selection (FCBF) method. This approach improves the quality of gene selection and human cancer classification. It can be used in binary and multi-category classification; it is also able to handle high dimensional Microarray datasets. By optimizing SVM, the proposed approach achieves higher classification accuracy among other methods, PSO-SVM (employed PSO to optimize the parameters of SVM) and ABC-SVM (utilized ABC to optimize parameters of

SVM).

Cancer classification can be divided into two tasks: gene selection and classification. Jin et al. [51] propose a Binary Improving Particle Swarm Optimization (BIPSO) algorithm to select informative genes; then construct a variety of the tumor classifiers to enhance the classification performance. Generally, using a single classifier is not an ideal strategy solving complex problems. Hence, the authors introduce a boosting ensemble operation in order to achieve lower misclassification rates and thorough performance. The main characteristic of this work is that it can automatically determine the number of nominated genes. The experimental results show that BIPSO approach has better performance than the original PSO and improved PSO approaches in terms of high classification accuracy, less genes selected, and effectiveness on multi-classes datasets.

4.2.2 ACO for Feature Selection

Tabakhi et al. [94] introduce an Ant Colony Optimization based unsupervised gene selection method, MGSACO, in order to enhance the classification accuracy. This approach aims to minimize the irrelevant of genes and maximize the relevance among genes. To better illustrate the usefulness of this method, it has been examined on 5 publicly available gene expression datasets for human cancer. Furthermore, the proposed method is compared to seven other well-known feature (gene) selection methods with respect to the classification accuracy of three classifiers including SVM, Naïve Bayes, and Decision Tree. According to the manuscripts, MGSACO obtained superior performance over different classification methods due to the population-based iterative improvement process. Within each iteration, a subset of genes is selected by a group of agents and evaluated based on a fitness function instead of using any learning model [94] .

Sharbaf et al. [89] propose a hybrid feature selection approach (CLACOFS) combining Cellular Automata (CA) and Ant Colony Optimization (ACO), which is used to model gene-gene interactions and learn the rules and structures of CA, respectively. The main contribution of the proposed method is to remove irrelevant genes, minimize search space and reduce computational complexity. CA is chosen due to its capability of parallel computing. The authors incorporate ACO with CA to boost the overall performance effectiveness in terms of greater convergence rate, less initial genes involved and higher influence on separating different categories.

El Houby et al. [30] implement an Ant Colony Optimization approach for solving the most challenging problem in feature selection, a large amount of less informative or irrelevant features in the search space. The ants are divided into 2 subgroups and each candidate feature is elevated according to different criteria [30] :

1. group 1: uses the nearest feature to the previously elected one based on its fitness value.

2. group 2: uses the furthest feature to the previously elected one in according to the fitness value.

This model uses both group 1 and 2 ants to elect different features that give the best heuristic and pheromone values. The results show that eliminating redundant and irrelevant features and the right nomination of features may minimize the classification error.

4.2.3 ABC for Feature Selection

Selecting highly informative genes is a challenging and interesting topic in Microarray data analysis. Li et al. [63], propose a multi-objective ranking binary Artificial Bee Colony algorithm (MORBABC/D) to discriminate immensely correlated genes from the original complex gene expression data. By employing extreme learning machine, the proposed method can intelligently select the most correlated features that can determine the smallest subsets, ignore redundant features and improve classification sensitivity. The MORBABC/D approach outperforms other algorithms, ELM (Linear-kernel), ELM (RBF-kernel), OS-ELM WE, LM, SVM, KNN, NB, LDA, NSGAII, MOPSO, MODE, and MOEA/D. In the classification accuracy table, TABLE 4.1, we only consider the results of the most popular classifiers such as SVM, KNN, NB and LDA.

Moosa et. al [72], presents a modified Artificial Bee Colony algorithm (mABC) to select informative biomarkers for cancer classification and prediction. This enhanced version consists of two steps: pre-selection of genes and modified ABC algorithm for gene selection,

1. Pre-selection: irrelevant genes are removed according to some basic statistical method such as Kruskal-Wallis and F-test. Then the authors determine the number of genes to be nominated in the next stage.

2. Modified ABC: after the pre-selection step, only the top ranked genes are retained. Then they are fed to the mABC model for the second filtration; and a smaller subset of relevant genes is formed.

mABC provides more accurate and promising classification results among 10 publicly available cancer datasets with a smaller subset of genes selected. In order to find the best parameter settings; mABC was tuned with full factorial combination [72], which includes the criteria of performance, run-time and method selection.

Classifying gene expression data is always challenging due to its characteristics that involve small sample size, imbalanced classes, and data complexity. Andaru et al. [6], believe fewer features/genes selected is equivalent to less computational time and space. In this work, the authors introduce a similar strategy as [72], which also involves two steps: filter and wrapper. In spite of evaluating the state-of-the-art ABC algorithm on a single classifier, Andaru has applied the proposed approach (ABC-reduced) on multiple classification methods including Decision Tree, K-NN and Rule Induction. The results show that prediction accuracy of ABC-reduced outperforms other algorithms such as GA and PSO.

4.2.4 GWO for Feature Selection

Grey wolf optimizer (GWO) is a new evolutionary computation technique to discover the optimal subset of relevant features. It mimics the leadership hierarchy and hunting mechanism of grey wolves in nature. Emary et al. [31] propose a binary grey wolf optimization (bGWO) strategy applied in the feature selection domain for maximizing the classification accuracy while reducing the number of selected features. The authors proposed two versions of the binary grey wolf optimizers, known as bGWO1 and bGWO2. To ensure the stability and statistical significance of the performance, the datasets are partitioned into 3 equal sets; and this apportioning was repeated 20 times [31]. Multiple measurements are used in order to illustrate the effectiveness and usefulness of the proposed approach such as:

1. Classification average accuracy: the correct predictions of a classifier with a given feature set.

2. Statistical best: the most optimistic solution acquired of an optimization algorithm at multiple operations.

3. Statistical worst: the pessimistic solution among the best solutions found for running an optimizer multiple times.

4. Statistical mean: the average of the solutions generated from running an optimizer for multiple runnings.

5. Standard deviation: the variation of the obtained best solutions originate for running an optimizer many times.

6. Average selection size: the average size of the feature subsets to the total number of features.

7. Average F-score: a measure of test's accuracy, calculating for individual features given the class labels [28].

8. Wilcoxon rank sum test: a nonparametric test that assigns rankings to all the scores considered as one group; then sums the ranking orders of each group [105].

The experimental results of the bGWO2 approach overcome the attained results for PSO and GA optimizers on 18 standard benchmark datasets, addressing various kinds of classification problems.

4.2.5 WOA for Feature Selection

Whale optimization algorithm (WOA) is a population-based stochastic algorithm and a new addition to the Swarm Intelligence family. It has been widely used in solving real-world problems such as economic dispatch [70], power system [3], neural network [17], image processing [98] and wireless sensor networks [4]. Mafarja et al. [88] propose two hybrid models, combining the whale optimization algorithm with a simulated annealing (SA) algorithm. SA is embedded in WOA in the first model and used to improve the best solution achieved after each iteration of WOA in the second model. The proposed approach is named WOASA and consists of two phases:

1. Exploitation phase: during hunting phase, whales first encircle the prey. This hunting strategy is known as the bubble-net attacking method.

2. Exploration phase: a random search agent is chosen to guide the search for prey.

The performance ensures the ability of the proposed hybridization models to effectively in search the feature space and select the significant features for building classifiers. The WOASAT2 (WOASA uses a tournament selection mechanism) and shows better performance than PSO and GA approaches on 16 datasets except for two, Exactly2 [33] and WaveformEW [117]. These two datasets contain a large number of samples but fewer features.

4.2.6 SSO for Feature Selection

Anter et al. [8] applied a new Social Spider Optimization algorithm (SSOA) to find global optima in the search space. It mimics the behavior of social spiders that interact with each other based on the biological law of the cooperative colony. In medical image and cancer classification areas, the feature extraction is an important stage in the pattern recognition system. SSO based algorthims designed to obtain a subset of features that is capable to present Region of Interest sufficiently [9].

4.2.7 MOA for Feature Selection

Monkey optimization algorithm (MOA) has been broadly engaged in many fields, for example, satellite imagery [49], image processing [93], wireless sensor networks [39], antenna wave [5], and power system [75]. Each evolutionary computation technique has its own strength and weakness. The main advantages of MO based algorithms are listed below:

1. Provision to handle problems such as stagnation or premature convergence in its original design [42].
2. Fewer parameters to adjust and low computational cost [76].
3. Easily handling of non-linear constraints [60].
4. The single structured group in the initiation stage that it's easier to attract newly generated food source towards the target [26].
5. Flexibility – MO can incorporate from other heuristics and optimization methods in a unique way [52].

Hafez et al. [43] introduce a hybrid optimization approach combining the Monkey Algorithm (MA) that mimics the social behavior of monkeys and the Krill Herd Algorithm (KHA), which studies the herding of the krill swarms in response to specific biological and environmental processes [35]. This new method, MAKHA, adaptively balance the exploration and exploitation to quickly reach the optimal solution. The hybrid MAKHA algorithm includes 6 major steps: watch-jump process, the somersault process, foraging motion, physical diffusion, applying foraging motion and the physical diffusion, and a genetic operator. The experimental results show that MAKHA obtains better classification accuracy in comparison with the-state-of-art PSO and GA algorithm. All results obtained from the approaches mentioned above are reported in TABLE 4.1, which includes each individual methodology, the datasets they used, classification accuracy and the development environment and settings.

Table 4.1. Comparison of Data Source and Classification Accuracies obtained from Different SI based Feature Selection Methods

Methods		**Datasets**	**Classifiers (Accuracy %)**			**Simulation**
PSO	BQPSO/BPSO	Leukemia	BPSO/SVM (100)	BQPSO/SVM (100)		MATLAB #Swarm (20) #Iteration (100)
		Prostate	BPSO/SVM (99.02)	BQPSO/SVM (99.25)		
		Colon	BPSO/SVM (91.94)	BQPSO/SVM (92.52)		
		Lung	BPSO/SVM (99.96)	BQPSO/SVM (99.96)		
		Lymphoma	BPSO/SVM (99.74)	BQPSO/SVM (99.79)		
	PA	Breast	PSO-SVM (87.63)	ABC-SVM (88.66)	PA-SVM (88.66)	MATLAB #Swarm (30) #Iteration (100)
		Lung	PSO-SVM (79.49)	ABC-SVM (74.36)	PA-SVM (79.49)	
		NervSys	PSO-SVM (90)	ABC-SVM (91.67)	PA-SVM (91.67)	
		Prostate	PSO-SVM (100)	ABC-SVM (100)	PA-SVM (100)	
		Colon	PSO-SVM (90.32)	ABC-SVM (93.55)	PA-SVM (93.55)	
		Leukemia	PSO-SVM (100)	ABC-SVM (100)	PA-SVM (100)	
		Ovarian	PSO-SVM (100)	ABC-SVM (100)	PA-SVM (100)	
		DLBCL1	PSO-SVM (98.70)	ABC-SVM (98.70)	PA-SVM (100)	
		DLBCL2	PSO-SVM (82.76)	ABC-SVM (82.76)	PA-SVM (86.21)	
	BIPSO	Leukemia	SVM (93.58)	LDA (91.95)	KNN (91.02)	Unknown #Swarm (-) #Iteration (20)
		Colon	SVM (94.95)	LDA (92.87)	KNN (93.09)	
		SRBCT	SVM (99.76)	LDA (97.71)	KNN (96.67)	
		Lymphoma	SVM (97.84)	LDA (96.53)	KNN (96.04)	
ACO	MGSACO	Colon	NB (80.00)	DT (76.37)	SVM (78.19)	MATLAB #Swarm (100) #Iteration (50)
		SRBCT	NB (84.14)	DT (77.25)	SVM (74.49)	
		Leukemia	NB (92.31)	DT (76.93)	SVM (82.06)	
		Prostate	NB (62.86)	DT (70.29)	SVM (73.15)	
		Lung	NB (80.00)	DT (80.00)	SVM (85.72)	
	CLACOFS	ALL-AM-Leukemia	NB (97.60)	KNN (95.95)	SVM (95.95)	MATLAB #Swarm (6) #Iteration
		Prostate	NB (99.40)	KNN (99.85)	SVM (99.25)	
		MLL-Leukemia	NB (99.30)	KNN (97.55)	SVM (-)	
		ALL-AML-4	NB (86.38)	KNN (80.99)	SVM (-)	
	ACO	Heart	KNN_all (82.50)	KNN_reduced (96.77)		MATLAB #Swarm (6) #Iteration (100)
		Breast	KNN_all (93.15)	KNN_reduced (97.95)		
		Thyroid	KNN_all (94.00)	KNN_reduced (98.25)		
ABC	mABC	9_Tumors	EPSO (75.00)	mABC/SVM (98.65)		MATLAB #Swarm (25) #Iteration (30)
		11_Tumors	EPSO (95.40)	mABC/SVM (99.50)		
		Brain_Tumor1	EPSO (92.11)	mABC/SVM (100)		
		Brain_Tumor2	EPSO (92.4)	mABC/SVM (100)		
		DLBCL	EPSO (100)	mABC/SVM (100)		
		Leukemia1	EPSO (100)	mABC/SVM (100)		
		Leukemia2	EPSO (100)	mABC/SVM (100)		
		Lung	EPSO (95.67)	mABC/SVM (100)		
		Prostate	EPSO (97.84)	mABC/SVM (100)		
		SRBCT	EPSO (99.64)	mABC/SVM (100)		

Methods		Datasets	Classifiers (Accuracy %)				Simulation
ABC	MORBABC/D	Colon	MORBABC/D (98.54)	SVM (85.65)	NB (63.55)	KNN (74.51)	MATLAB #Swarm (50) #Iteration (100)
		ALL-AML	MORBABC/D (100)	SVM (98.61)	NB (85.27)	KNN (85.41)	
		Breast	MORBABC/D (92.16)	SVM (70.10)	NB (69.27)	KNN (59.89)	
		Lung	MORBABC/D (100)	SVM (99.28)	NB (99.00)	KNN (95.30)	
		Ovarian	MORBABC/D (100)	SVM (99.29)	NB (88.49)	KNN (94.98)	
		Prostate	MORBABC/D (98.43)	SVM (92.74)	NB (61.96)	KNN (83.23)	
		DLBCL	MORBABC/D (100)	SVM (97.40)	NB (78.31)	KNN (83.76)	
	ABC-reduced	CNS	DT (63.30)	RI (63.30)	KNN (63.30)		MATLAB #Swarm (30) #Iteration (40)
		Leukemia	DT (84.70)	RI (87.50)	KNN (91.70)		
		Lung	DT (89.60)	RI (89.20)	KNN (90.10)		
		Lymphoma	DT (87.90)	RI (89.40)	KNN (98.50)		
		MLL	DT (76.40)	RI (69.40)	KNN (81.90)		
		Ovarian	DT (96.40)	RI (96.40)	KNN (96.40)		
		SRBCT	DT (81.90)	RI (81.90)	KNN (96.40)		
GWO	bGWO	Breast	bGWO1 (97.60)	bGWO2 (97.50)	GA (96.80)	PSO (96.70)	MATLAB #Swarm (8) #Iteration (70)
		Exactly	bGWO1 (70.80)	bGWO2 (77.60)	GA (67.40)	PSO (68.80)	
		Exactly2	bGWO1 (74.50)	bGWO2 (75.00)	GA (74.60)	PSO (73.00)	
		Lymphography	bGWO1 (74.40)	bGWO2 (70.00)	GA (69.60)	PSO (74.40)	
		M-of-N	bGWO1 (90.80)	bGWO2 (96.30)	GA (86.10)	PSO (92.10)	
		SpectEW	bGWO1 (82.00)	bGWO2 (82.20)	GA (79.30)	PSO (82.20)	
		SonarEW	bGWO1 (73.10)	bGWO2 (72.90)	GA (75.40)	PSO (73.70)	
		PenglungEW	bGWO1 (60.00)	bGWO2 (58.40)	GA (58.40)	PSO (58.40)	
		IonosphereEW	bGWO1 (80.70)	bGWO2 (83.40)	GA (81.40)	PSO (81.90)	
		HeartEW	bGWO1 (77.60)	bGWO2 (77.60)	GA (78.00)	PSO (78.70)	
		BreastEW	bGWO1 (92.40)	bGWO2 (93.50)	GA (93.20)	PSO (96.70)	
WOA	WOASAT	Breast	WOASAT2 (97.00)	PSO (95.00)	ALO (96.00)	GA (96.00)	MATLAB #Swarm (10) #Iteration (100)
		BreastEW	WOASAT2 (98.00)	PSO (94.00)	ALO (93.00)	GA (94.00)	
		Exactly	WOASAT2 (100)	PSO (68.00)	ALO (66.00)	GA (67.00)	
		Exactly2	WOASAT2 (75.00)	PSO (75.00)	ALO (75.00)	GA (76.00)	
		HeartEW	WOASAT2 (85.00)	PSO (78.00)	ALO (83.00)	GA (82.00)	
		IonosphereEW	WOASAT2 (96.00)	PSO (84.00)	ALO (87.00)	GA (83.00)	
		Lymphography	WOASAT2 (89.00)	PSO (69.00)	ALO (79.00)	GA (71.00)	
		M-of-N	WOASAT2 (100)	PSO (86.00)	ALO (86.00)	GA (93.00)	
		PenglungEW	WOASAT2 (94.00)	PSO (72.00)	ALO (63.00)	GA (70.00)	
		SpectEW	WOASAT2 (88.00)	PSO (77.00)	ALO (80.00)	GA (78.00)	
MOA	MAKHA	Breast	MAKHA (95.97)		GA (98.20)	PSO (98.28)	MATLAB #Swarm (10) #Iteration (70)
		BreastEW	MAKHA (95.16)		GA (71.60)	PSO (96.53)	
		Exactly	MAKHA (81.38)		GA (78.26)	PSO (71.59)	
		Exactly2	MAKHA (74.05)		GA (76.58)	PSO (76.28)	
		HeartEW	MAKHA (78.22)		GA (86.22)	PSO (83.78)	
		IonosphereEW	MAKHA (84.96)		GA (88.38)	PSO (87.35)	
		Lymphography	MAKHA (74.00)		GA (87.20)	PSO (82.00)	
		M-of-N	MAKHA (97.00)		GA (96.16)	PSO (91.59)	
		SpectEW	MAKHA (80.00)		GA (87.19)	PSO (84.94)	

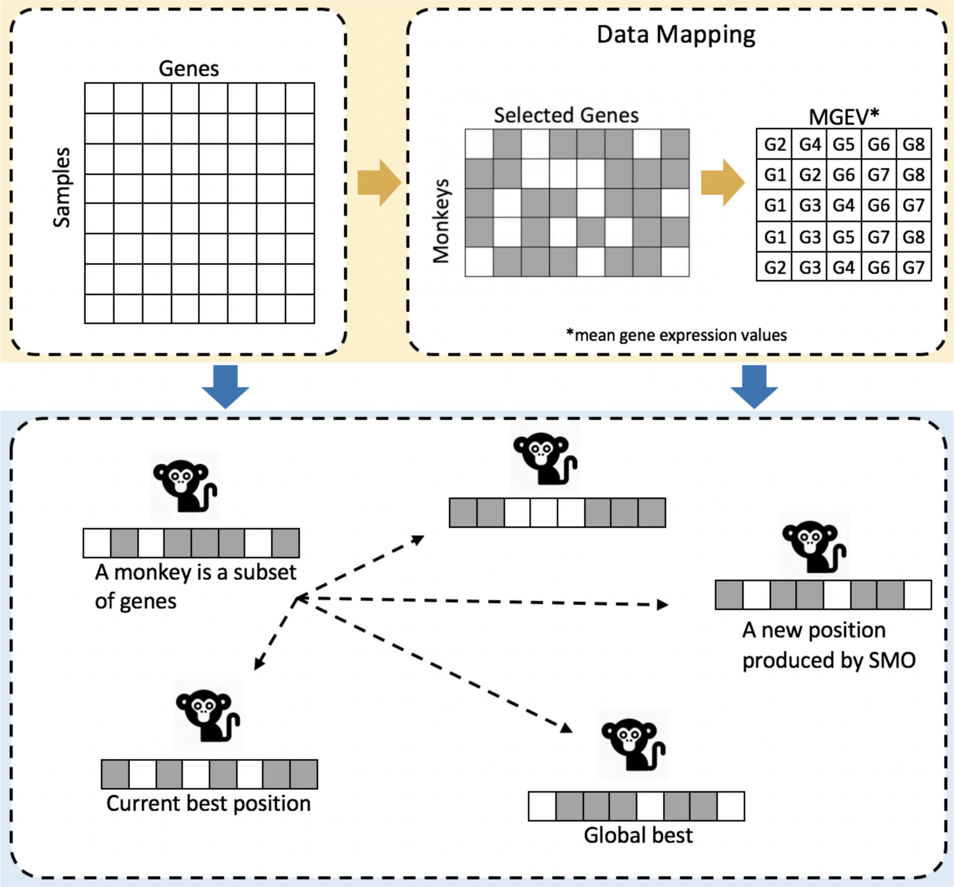
4.3 Our Methodology

Balancing exploration of the search space and exploitation of the optimal solutions are the keys in designing Spider Monkey Optimization (SMO) algorithms. In this subsection, we first initialize the food source positions. Secondly, we introduce the six main phases of SMO algorithm and their functions. The algorithm is iterated for a maximum iteration times which is a constant number set by user. Each iteration outputs a local best solution. Finally, the solution with maximum fitness is our global best which contains the optimal subset of features. In the literature, the most ideal case is that the algorithm could discover that a subset contains only one feature with 100% accuracy [72].

In this work we focus on employing the Spider Monkey Optimization (SMO) technique to minimize the number of features been used for classifying the cancer disease. The major steps are,

1. process and insert the microarray gene expression data
2. each monkey is represented as a subset of features
3. calculate the fitness scores and select the optimal feature subsets using spider monkey optimization
4. the model stops when the termination criterion is reached
5. the best subset of features evaluated using the predefined classifiers

The pseudo-code of the proposed SMO algorithm for feature selection is given in Figure 4.1 and the model flowchart is described in Figure 4.2 .


Genes
Samples
Data Mapping
Selected Genes
Monkeys
MGEV*
*mean gene expression values
A monkey is a subset of genes
A new position produced by SMO
Current best position
Global best

G2	G4	G5	G6	G8
G1	G2	G6	G7	G8
G1	G3	G4	G6	G7
G1	G3	G5	G7	G8
G2	G3	G4	G6	G7

Figure 4.1. SMOFS Process for Optimal Gene Selection

Population Initialization Phase (PIP) In this phase, we set up the food source positions for total N spider monkeys. The position of the i_{th} spider monkey SM_i is represented as $SM_i = \{SM_i^1, SM_i^2, ..., SM_i^P\}$, where P is the feature size or dimension of the input data.

Local Leader Phase (LLP) This phase is based on the local leader and individual group members (SM_i) experience to adjust the new location. Greedy selection is applied by comparing between new position and current position by a well-defined fitness function [39].

$$SM_i^P(new) = SM_i^P + \Theta_1 * (LL_k^P - SM_i^P) + \Theta_2 * (SM_j^P - SM_i^P) \qquad (4.1)$$
$$\text{where } \Theta_1 \in [0,1] \text{ and } \Theta_2 \in [-1,1]$$

Global Leader Phase (GLP) This phase starts based on global leader and members of

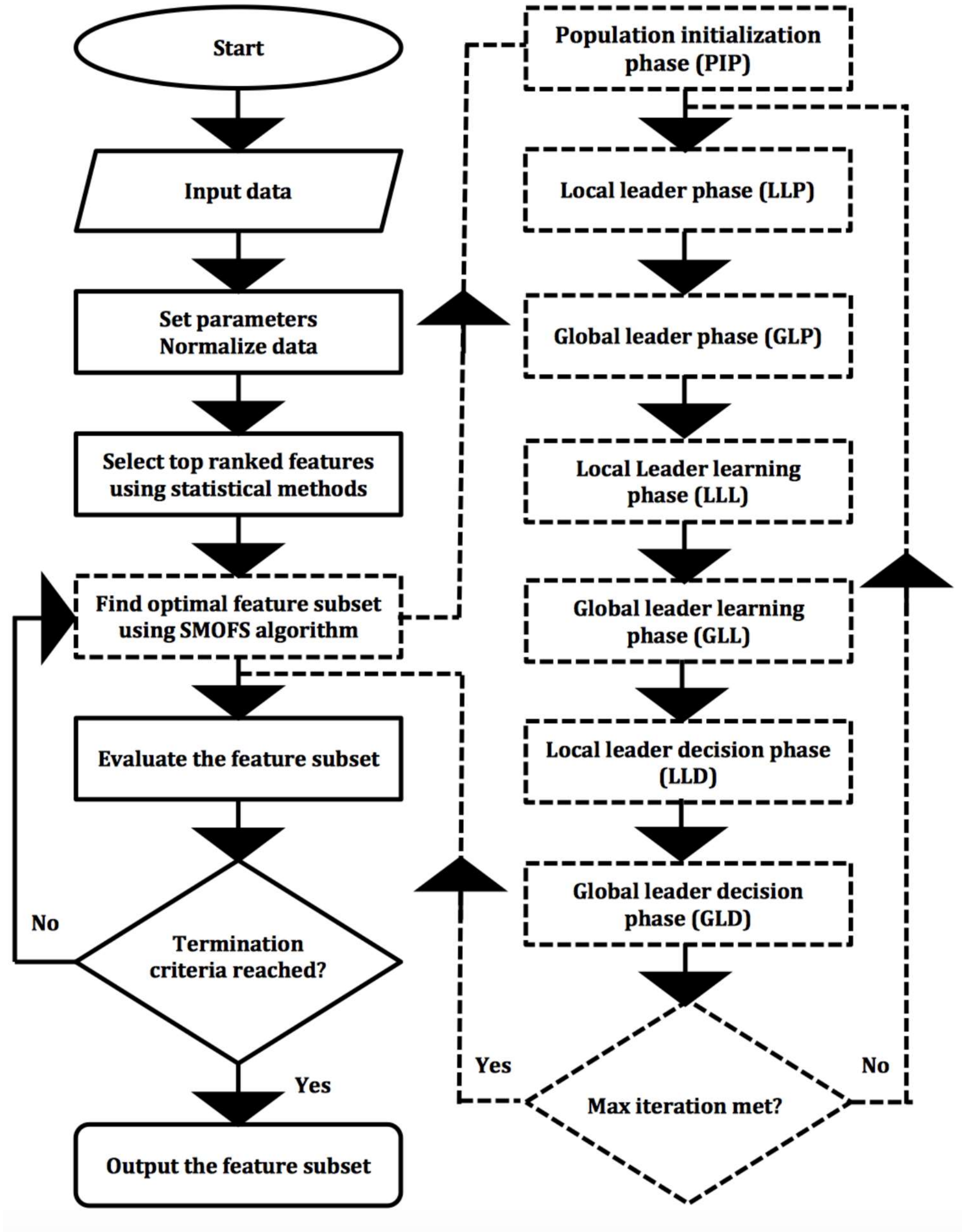


Figure 4.2. Spider Monkey Optimization based Feature Selection Flow

local group's experience [39]; the equation used to modify their positions is,

$$prob_i = (0.9 * Fitness_i / Fitness_{max}) + 0.1 \tag{4.2}$$

$$\begin{aligned} SM_i^P(new) = SM_i^P + \Theta_1 * (GL^P - SM_i^P) + \Theta_2 * (SM_j^P - SM_i^P) \\ \text{where } \Theta_1 \in [0,1] \text{ and } \Theta_2 \in [-1,1] \end{aligned} \tag{4.3}$$

Local Leader Learning (LLL) In this phase, the local leader position is updated by an algorithmic paradigm, greedy selection, which is making the optimal solution in the population at each stage. The LocalLimitCount value is incremented by 1 due to the comparison of the updated position of the local leader with its previous position [39].

Global Leader Learning (GLL) In this phase, global leader is updated using same strategy as LLL phase. Furthermore, the GlobalLimitCount threshold increases by 1 whether the position of global leader is updated or not [39].

Local Leader Decision (LLD) During this phase, decision taken upon the updating of any local leader position. If it is not updated up to the LocalLimitCount value, then all members within the group modernize their positions according to the global leader and local leader experience [39].

$$\begin{aligned} SM_i^P(new) = SM_{min}^P + \Theta * (SM_{max}^P - SM_{min}^P) \\ \text{where } \Theta \in [0,1] \end{aligned} \tag{4.4}$$

$$\begin{aligned} SM_i^P(new) = SM_i^P + \Theta * (GL^P - SM_i^P) + \Theta * (SM_j^P - LL_k^P) \\ \text{where } \Theta \in [0,1] \end{aligned} \tag{4.5}$$

Global Leader Decision (GLD) If the position of global leader is not updated up to a specific iterations value, known as GlobalLimitCount, then the population is divided into smaller groups based on global leader's decision [39].

Once the expression data are generated, our SMOFS algorithm obtains optimal subsets of representative genes that are offered to the biologist and specialist as the genes responsible for the cause of cancer. In Figure 4.1, the process of selecting optimal subset of genes is shown. The reported results by SMOFS are evaluated by means of their classification accuracy using cross-validation and multiple classifiers. The main contribution of our approach is notable, and it is easy to interpret. It offers an enhancement on existing state of the art algorithms in terms of computational effort and classification accuracy. Furthermore, the gene ensembles found by this methodology can be biologically meaningful, not just computationally.

4.4 Data and Experimental Settings

We collect 5 different data sets from the UCI data repository [101, 15, 106, 57, 7] and the summary of data sets is shown in Table 4.2. In Table 4.2, we gather 3 data sets that involve a fewer number of features but a fairly large number of patient samples. Two of the data sets, Wisconsin Breast Cancer and SPECT Heart Data, have imbalanced classes. For training, validation and testing purposes, we apply 5-fold cross-validation resampling procedure, each data-set is equally splitted into 5 portions as training and test sets. The experimental settings for the different feature selection methods are described in Table 4.3.

Table 4.2. Statistics of Data Sets

Data	Features	Samples (+/-)	Classes	Source
Wang Breast Cancer	22283	209 (138/71)	2	[101]
Lung Adenocarcinoma	7129	86 (62/24)	2	[15]
Wisconsin Breast Cancer	10	699 (458/241)	2	[106]
SPECT Heart Data	22	267 (212/55)	2	[57]
Diabetic Retinopathy Debrecen	20	1150 (611/539)	2	[7]

Note: positive sample (+), negative sample (-).

Table 4.3. Parameter Settings

No.	Parameter	Value
1	Number of agents	20
2	Number of iteration	100
3	Problem dimension	number of features
4	Inertia factor of PSO	0.1
5	Acceleration factor of PSO	0.1

4.5 Evaluation Criteria

All methods are examined by total of 10 runs on MATLAB R2018a environment to test the convergence ability and the statistical significance. The detailed evaluation criteria are listed below:

- Classification accuracy: the correct predictions of a classifier with a given feature set,

$$Accuracy = \frac{1}{M}\sum_{M}^{j=1}\sqrt{\frac{1}{N}\sum_{N}^{i=1}(True_i - Pred_i)^2} \tag{4.6}$$

  where M and N are the number of experimental runs and the number of test samples, respectively. $True_i$ and $Pred_i$ indicate the true and predicted class labels for the i^{th} data point.

- Standard deviation (std): the variation of the obtained best solutions originate for running an optimizer over many times,

$$Std = \sqrt{\frac{\sum_{i=1}^{N}(x_i - \overline{x})^2}{N-1}} \tag{4.7}$$

- Precision: the measure of exactness or quality. It refers to the percentage of the outcomes which are relevant [38].

$$Precision = \frac{TruePositive}{(TruePositive + FalsePositive)} \tag{4.8}$$

- Recall: the measure of completeness or quantity. It refers to the percentage of relevant outcomes correctly classified [38].

$$Recall = \frac{TruePositive}{(TruePositive + FalseNegative)} \tag{4.9}$$

- Fisher score (f1): a measure of test's accuracy, calculating for individual features given the class labels, as determined by the equation below [87],

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{4.10}$$

4.6 Results and Discussion

In this subsection, we apply the SMO algorithm adaptively to find the optimal feature subse, known as SMOFS, that maximizing the classification performance in terms of accuracy, recall, precision, and f1 score. We compare the performance criterion of various classification methods involving Random Forests (RF), K-Nearest Neighbors (KNN) and Support Vector Machine (SVM) to determine the best approach combining Swarm Intelligence based feature selectors. One of the reasons for selecting KNN algorithm is because it has the capacity to learn nonlinear relationships between genes/features [79]. These algorithms are chosen because they are frequently applied in the disease classification and have potential to yield promising results. We apply cross-validation resampling method to avoid overfitting, measure the unbiased estimator, and compare the prediction models. Table 4.4 and Table 4.5 show comparison between our SMOFS and PSOFS in literature for the datasets Wang Breast Cancer, Lung Adenocarcinoma Data, Wisconsin Breast Cancer, SPECT Heart Data, and Diabetic Retinopathy Debrecen. Our SMOFS approach combining all three classifiers achieved the best accuracy scores on most of the cancer data sets. The stability are evaluated on the standard deviation of multiple experiments, f1 score and the best number of features selected. For the Wang Breast Cancer data, the feature selection level is 5 (genes) for the PSOFS approach and 25 (genes) for the SMOFS approach. With slightly more genes chosen, our SMO based feature selector has achieved higher accuracy in all cases (classifiers). Similarly, the Lung Adenocarcinoma data shows better results across all classifiers with a feature selection level of 15 (genes) in SMOFS compare to the selection level of 10 (genes) using the PSOFS approach. In all other data sets considering both the classification accuracy and the selected feature size, our SMOFS approach performed comparatively better.

In Figure 4.3, we compute the misclassification probability for out-of-bag observations in the training data of the Baseline, PSOFS and SMOFS approaches. The baseline approach includes all the features from the input data sets. In the literature, the baseline accuracy level of the Wisconsin Breast Cancer data is 96.84% with 9 out of 10 features; the SPECT Heart data claims a slightly lower accuracy level of 74.0% with all the features; and the Diabetic Retinopathy Debrecen data has a more comparative accuracy level of 98.90% including all the features as well. For each estimator, we observe a total number of 50 trees in the ensemble.

Table 4.4. Results on Various Data Sets

Methods		Evaluation Metrics		
		Accuracy (%)	F1 Score	Features Selected
	Wang Breast Cancer			
Random Forests	PSOFS	64.16 (0.20)	**0.5905**	5
	SMOFS	**68.49 (0.08)**	0.4698	25
K-Nearest Neighbors	PSOFS	63.72 (0.12)	0.4023	5
	SMOFS	67.81 (0.04)	0.3210	25
Support Vector Machine	PSOFS	63.50 (0.10)	0.3279	5
	SMOFS	67.33 (0.02)	0.2766	25
	Lung Adenocarcinoma Data			
Random Forests	PSOFS	63.14 (0.94)	0.4188	10
	SMOFS	65.38 (0.29)	0.4545	15
K-Nearest Neighbors	PSOFS	61.27 (0.63)	0.4176	10
	SMOFS	67.19 (0.22)	**0.5000**	15
Support Vector Machine	PSOFS	63.50 (0.48)	0.4234	10
	SMOFS	**70.00 (0.28)**	0.3970	15

Note: the numbers in parentheses are absolute standard deviations.

Table 4.5. Results on Various Data Sets (Continue)

Methods		Evaluation Metrics		
		Accuracy (%)	F1 Score	Features Selected
		Wisconsin Breast Cancer		
Random Forests	PSOFS	94.33 (0.29)	0.9610	5
	SMOFS	**95.20 (0.15)**	**0.9631**	5
K-Nearest Neighbors	PSOFS	94.46 (0.22)	0.9563	5
	SMOFS	95.12 (0.35)	0.9609	5
Support Vector Machine	PSOFS	94.51 (0.28)	0.9573	5
	SMOFS	94.60 (0.50)	0.9611	5
		SPECT Heart Data		
Random Forests	PSOFS	76.73 (1.53)	0.6890	6
	SMOFS	**78.38 (0.73)**	**0.7222**	6
K-Nearest Neighbors	PSOFS	76.83 (1.50)	0.1167	6
	SMOFS	78.29 (0.86)	0.2124	6
Support Vector Machine	PSOFS	76.94 (1.50)	0.5584	6
	SMOFS	78.33 (0.97)	0.6084	6
		Diabetic Retinopathy Debrecen		
Random Forests	PSOFS	97.49 (0.39)	0.9340	15
	SMOFS	**98.30 (0.21)**	0.9385	15
K-Nearest Neighbors	PSOFS	97.54 (0.14)	0.9762	15
	SMOFS	97.36 (0.06)	0.9787	15
Support Vector Machine	PSOFS	97.98 (0.06)	0.9781	15
	SMOFS	97.84 (0.01)	**0.9811**	15

Note: the numbers in parentheses are absolute standard deviations.

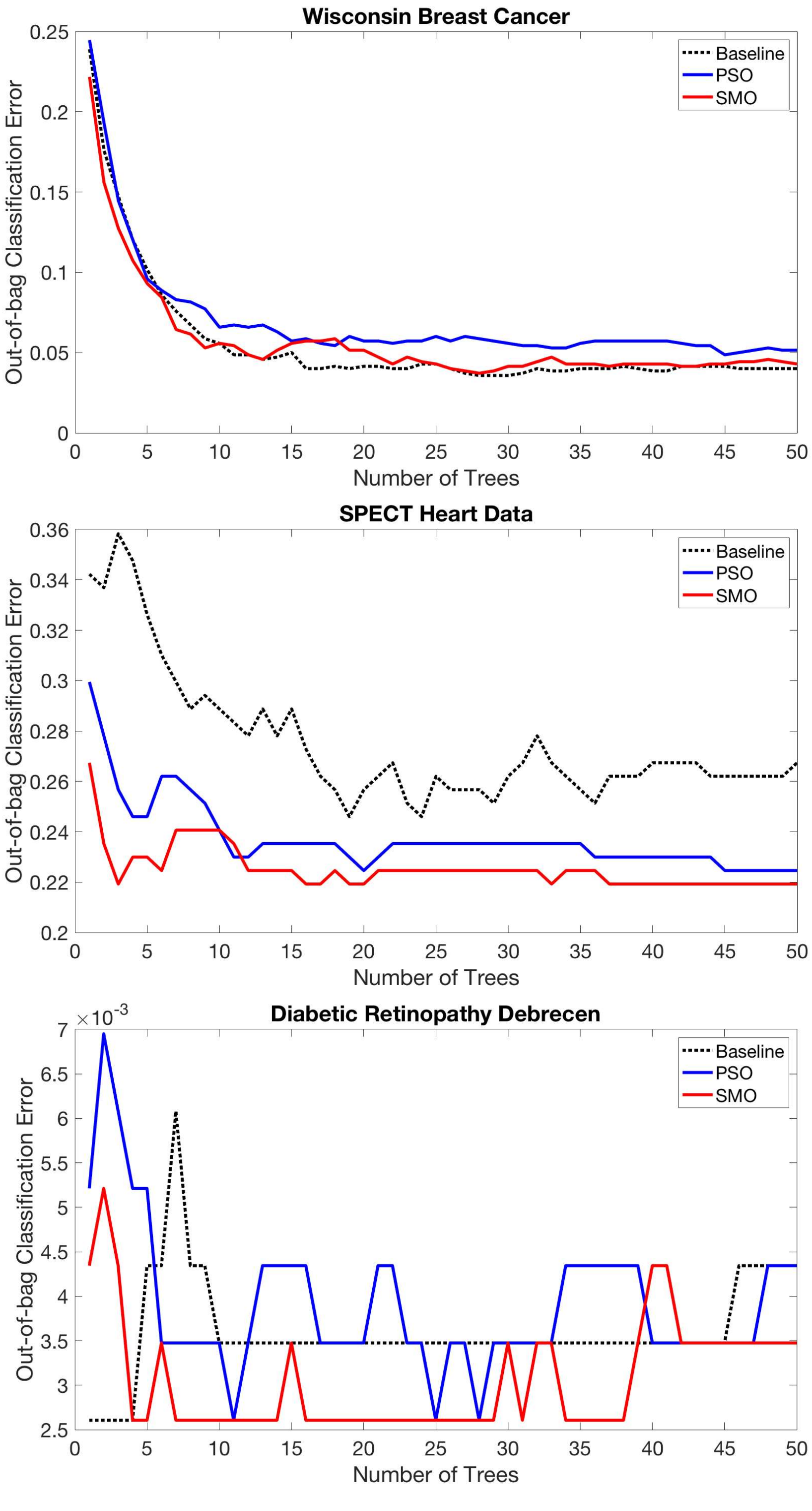


Figure 4.3. Misclassification Rates for Different Number of Trees (RF)

The performance results of the SMOFS algorithm proves its capability to balance between the exploration and exploitation throughout iterations of the optimization. As per the results obtained, the performance of the WOA algorithm is proved on the large data sets as well as on the smaller size data sets. The two datasets - Wang Breast Cancer and Lung Adenocarcinoma data sets are relatively large and the F1 scores of the proposed approach is clearly higher than PSOFS approach. We can also see that our SMOFS algorithm outperforms the PSOFS in terms of the best and worst obtained solution.

CHAPTER 5

SPIDER MONKEY OPTIMIZATION IN WIRELESS SENSOR NETWORKS

In this section, we aim to extend the ability of SMO in the field of Wireless Sensor Networks by finding the optimal route for sending information from the clusters to the base station through sensors. The goal of our approach, cluster-based Spider Monkey Optimization (SMO-C), is to improve the network performance and reduce energy consumption. By designing a new protocol approach, it allows us to space out lifespan of the sensor node due to its limited battery life and other resource constraints. In cluster-based protocol, cluster head is assigned to collect data from its surrounding nodes and passes it on to the base station as shown in Fig. 5.1.

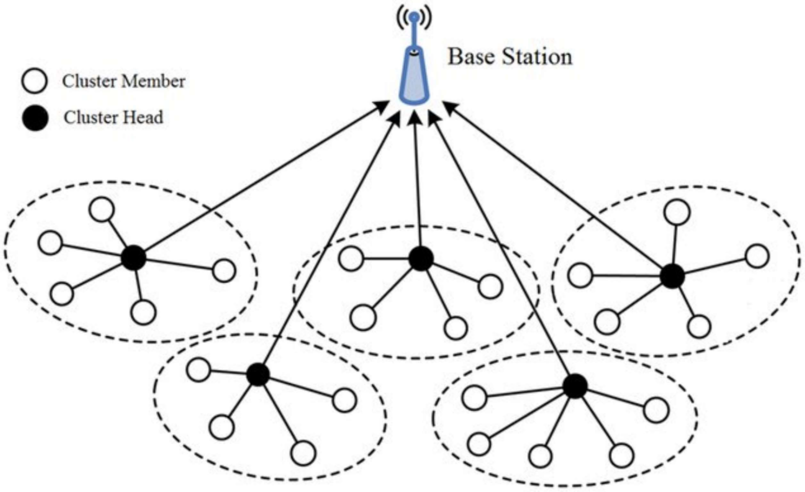


Figure 5.1. Cluster-based wireless sensor networks (WSNs)

5.1 Design Challenges

Saleem et al. [86] proposed a general list of essential factors for wireless sensor network routing protocols like scalability, self-organization, memory requirements, sensor localization, fault tolerance, energy efficiency and security. Due to the large amount of sensor nodes, unstable energy sources and unpredictable operating environment, these facts present unique challenges on the architecutral design and application development of WSNs. Particularly, major concerns in routing protocols developed by monkey-inspired optimization are robustness, reachability, scalability, simplicity, coverage, routing strategies, flexibility and quality of service. In this section, a detailed explanation on some of these factors is provided below:

1. Scalability: it is a challenging problem in WSNs – large amount of sensor nodes are expected to be heterogeneously deployed with long transmission path [86].

2. Self Organization: monkey inspired protocols must be flexible to predict variations, in static and dynamic situations [80].

3. Multipath Routing: adopting a multiple path strategy can extend network lifetime. It can be considered as a backup when the initial path fails [54].

4. Localized Interaction: full locality awareness, restrition of interactions to neighboring sensor nodes, is a main charactertic of self-organization. Types of communication in WSNs are aggregation, distribution, and broadcast [65].

5. Failure Detection: the communication among sensor nodes can be affected by battery life, location, weather condition, obstacles, antenna and others [64].

6. Memory Requirements: Given limited on-board memory, the routing algorithms developed require minimal processing overhead in order to have an efficient execution of functionalities [24].

7. Energy Efficiency: Due to limited battery life on sensor nodes, the power usage effectiveness is a critical challenge in WSNs to support longer network operational time [24].

8. Robustness: Each sensor node needs to be constructed to be as robust as possible in case of battery life shortening. The robustness can be improved through the use of multi-channel and multi-radio [23].

5.2 Our Methodology

5.2.1 Cluster Head Selection

In SMO-C, a cluster head is selected based on the residual energy of the sensor node. This election approach can be visualized in a group of spider monkeys, containing both females and males. In nature, the female monkey is always the leader; it ranks by its fitness and fertility [58]. If the current leader dies, the authority will pass to the next female monkey. For redundant cluster heads without any nodes attached to them, they will be automatically assigned to the nearest cluster. This feature is inspired from the fission-fusion behavioral structure of spider monkeys.

Cluster head (CH) selection process is dynamic because the duty of cluster-head rotates. SMO-C protocol chooses cluster head by the user-specific threshold shown below,

$$CH_{prob} = P_i * P_{CH} \tag{5.1}$$

where P_i and P_{CH} are the probability of i_{th} solution for every group member and percentage of cluster heads in each iteration, respectively. According to LEACH [45], some of nodes cannot be nominated as cluster heads in the initialization stage. The SMO-C approach is designed at selecting the cluster heads with better location to extend the overall network performance in terms of energy loss and fewer dead nodes. The execution flow of the cluster formation in SMO-C is shown in Fig. 5.2.

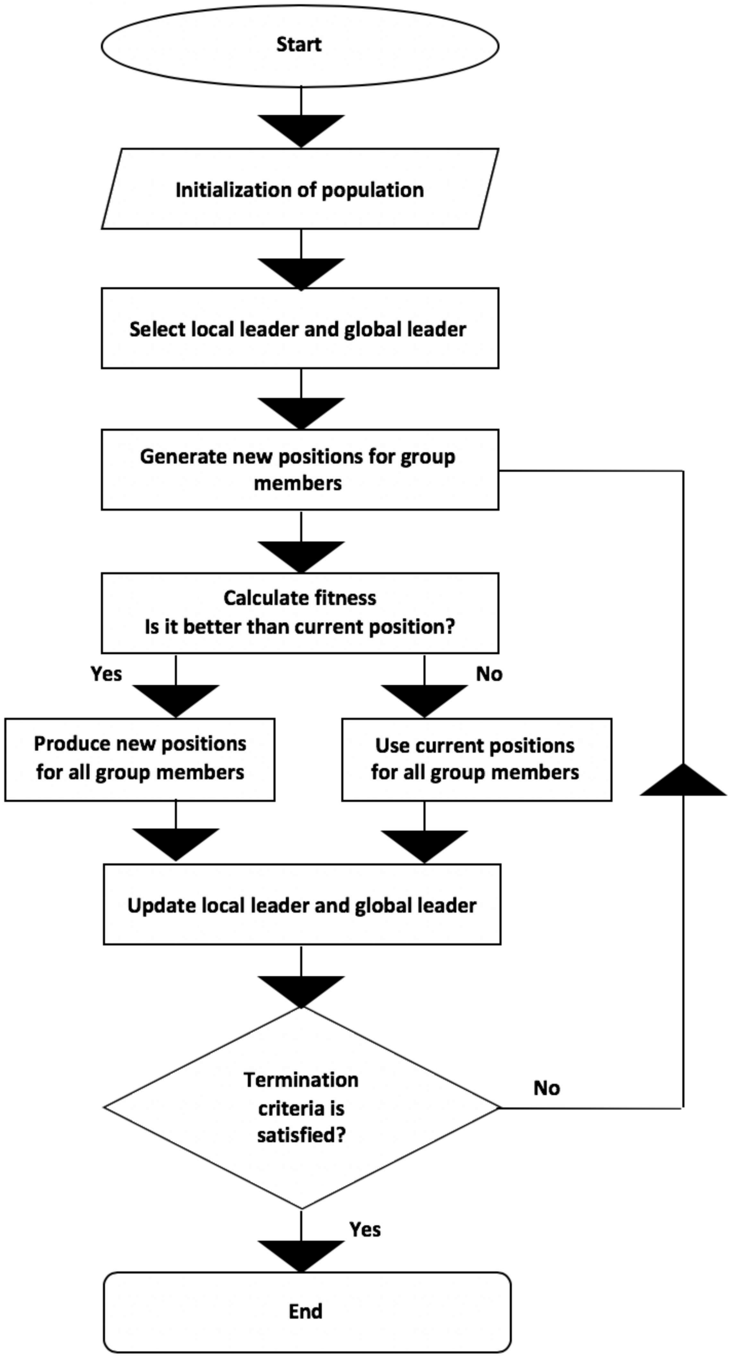

Start
Initialization of population
Select local leader and global leader
Generate new positions for group members
Calculate fitness
Is it better than current position?
Yes
No
Produce new positions for all group members
Use current positions for all group members
Update local leader and global leader
Termination criteria is satisfied?
No
Yes
End

Figure 5.2. Flowchart of SMO-C Cluster Formation [39]

In order to optimize the effectiveness, we present an improved cluster head selection scheme (SMOCH) to maximize network lifetime and minimize energy dissipation of our SMO-C protocol. Fig. 5.3 shows an illustration of how SMOCH works, where it updates the position of individuals based on their fitness scores to increase the convergence performance and exploitation capability. In this way a better-positioned node can have a higher chance to be chosen as cluster head. SMOCH extends the ability of the original Spider Monkey Optimization algorithm and LEACH by nominating the cluster heads with better location in order to improve overall network performance in terms of fewer dead nodes and energy consumption. The procedure of position update process in SMOCH algorithm is shown below:

Algorithm 7 cluster head selection for SMO based routing protocol

```
Initialize population with $n$ spider monkeys $SM$, $k$ random cluster centers and cluster head
    probability $CH_{prob}$
while max number of iteration is not reached do
    for each $SM$ do
        Calculate Euclidean distance of $SM$ with all cluster centroids
        Assign $SM$ to the cluster that has the nearest centrism
    end for
    Calculate the fitness and $CH_{prob}$
    if $Uniform(0, 1) < CH_{prob}$ then
        Update the cluster centroids based on local best
    end if
end while
```

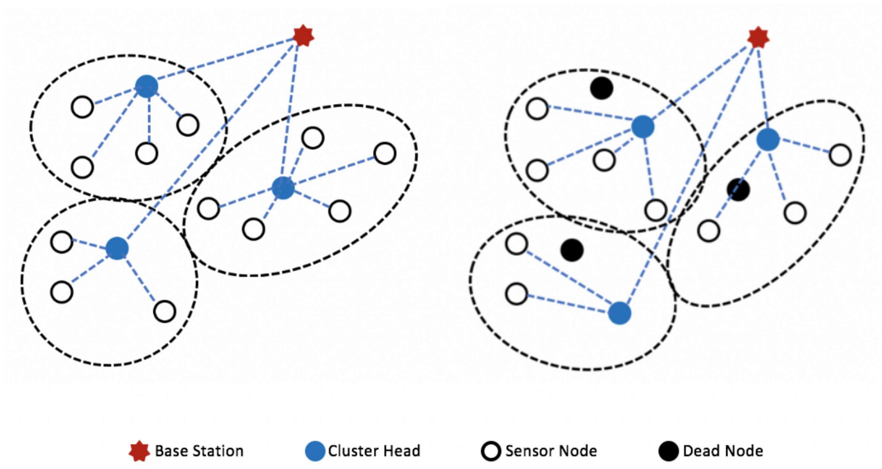


Figure 5.3. SMO: The Process of New Cluster Heads Election

5.2.2 First Order Radio Model

First order radio model is used to estimate energy consumption of the nodes. Assume each sensor node does not consume any energy when it is not receiving or sending any packet. We present the low-energy radio model, which was adapted in the original LEACH [45] protocol as well as the EAMMH [74] cluster algorithm. Different protocols take advantage of different assumptions of the radio model, such as energy dissipation in transmit and receive modes. We assume a simple model where the radio dissipates ET_x = 50 nJ/bit to run the transmitter or receiver and the amplifier losses of the sending node is set as E_{amp} = 100 pJ/bit/m2 . The energy consumed by the node to receive or send 1 bit packet is E_{elec}. When the condition satisfies the communication distance d and the energy consumed to send a k bit packet, the radio expands.

The assumption that the radio channel is symmetric is made that the energy required to transmit a data packet from node SN_i to SN_j is the same as the energy required to transmit from SN_j to SN_i. Furthermore, all nodes are sensing the environment at a fixed rate, which is a data-driven simulation.

5.2.3 Energy Utilization

There are many network routing protocols proposed for wireless sensor networks. We examine two of these protocols as our baseline approaches, LEACH [45] and EAMMH [74]. In SMO-C, we consider a low energy routing protocol. There are several power aware protocols, nodes route messages through intermediate nodes instead of direct communication through the cluster head. In this case, the intermediate nodes are known as our local leaders within a group of spider monkeys and the current cluster head is labeled as global leader. The intermediate nodes are chosen such that the distance to the cluster head and the base station achieve the best fitness scores. For future SMO-C versions, we will consider an event-driven sensor activation scenario, where sensors only transmit data if some event occurs, as well as a combination of minimum- transmission-energy (MTE) routing with our fitness-constraint approach in determining the routes.

5.3 Baseline Protocols

LEACH [45] is a low-energy adaptive clustering hierarchy for wireless sensor networks. The LEACH operation is divided into rounds. For each round, it consists of two phases,

Set-up Phase: selection of cluster head using Eq. 5.2 and cluster formation.

$$T(n) = \begin{cases} \frac{P_{CH}}{1-P_{CH}\times(rmod(P_{CH}^{-1}))} & , n \in N \\ 0 & , otherwise \end{cases} \tag{5.2}$$

where P_{CH} denotes the user specific percentage of cluster heads, r denotes the number of round in current, and N is the set of nodes that has potential to be elected as cluster heads in the future rounds. In each cluster, every individual sensor has equal chance to become the cluster head.

Steady State: this phase includes data collection, data aggregation, and data transmission to the sink. LEACH applies MTE transmission, which performances better than direct transmission to the base station. The transmit route is selected if and only if both Eq. 5.3

and 5.4 are true,

$$E_{amp}(k, d(SN_i, SN_j)) + E_{amp}(k, d(SN_j, SN_t)) < E_{amp}(k, d(SN_i, SN_t)) \tag{5.3}$$

$$d(SN_i, SN_j)^2 + d(SN_j, SN_t)^2 < d(SN_i, SN_t)^2 \tag{5.4}$$

EAMMH [74] is an energy aware multi-hop multi-path hierarchical protocol for WSNs. In EAMMH, the cluster heads are elected using the initial energy level in order to equalize the magnitude for energy consumption. Competing with the probabilistic distribution in the LEACH protocol, the deployment of cluster heads in EAMMH is more consistent. The intra-cluster multi-hop strategy is adapted due to the fact that some nodes may consume larger amount of energy through long-distance transmission in terms of data volume and node location [74]. The energy consumed for any cluster member node SN_i to its cluster head SN_{CH} is represented in Eq. 5.5. EAMMH adopts a free space propagation channel model to deliver k-bit packets from node SN_i to another node SN_j, which can communicate with the SN_{CH} as follows,

$$E(SN_i, SN_j) = E_{Tx}(k, d(SN_i, SN_j)) + E_{Rx}(k) + E_{Tx}(k, d(SN_j, SN_{CH})) \tag{5.5}$$

thus the node with smallest value of energy cost, $E(SN_i, SN_j)$, will act as the intermediate node.

5.4 Evaluation Environment

In our evaluation environment, we simulated each monkey inspired optimization technique with MATLAB by randomly distributing all sensor nodes in an area of $100 \times 100(m^2)$. The base station, which acts like a gateway between sensor nodes and the end user, is placed in a permanent location in the sensing area. Experimental parameters are summarized in TABLE 5.1

Table 5.1. Experimental Settings

Parameter	**Symbol**	**Value**
Number of nodes	n	100
Number of iterations	r	500, 1000
Base station	(x, y)	(150, 50)
Distance threshold	d	$\sqrt{(}E_{fs}/E_{amp})m$
Percentage of cluster heads	P	20%
Transmitter electronics	E_{tx}	50 nJ/bit
Receiver electronics	E_{rx}	50 nJ/bit
Transmit Amplifier type 1	E_{fs}	10 pJ/bit/m^2
Transmit Amplifier type 2	E_{amp}	0.0013 pJ/bit/m^2
Data aggregation energy	E_{da}	5 nJ/bit
Initial energy	E_{init}	(0.1, 0.25, 0.5, 1.0) J/node

5.5 Results and Discussion

5.5.1 Performance Comparison with LEACH and EAMMH Approaches

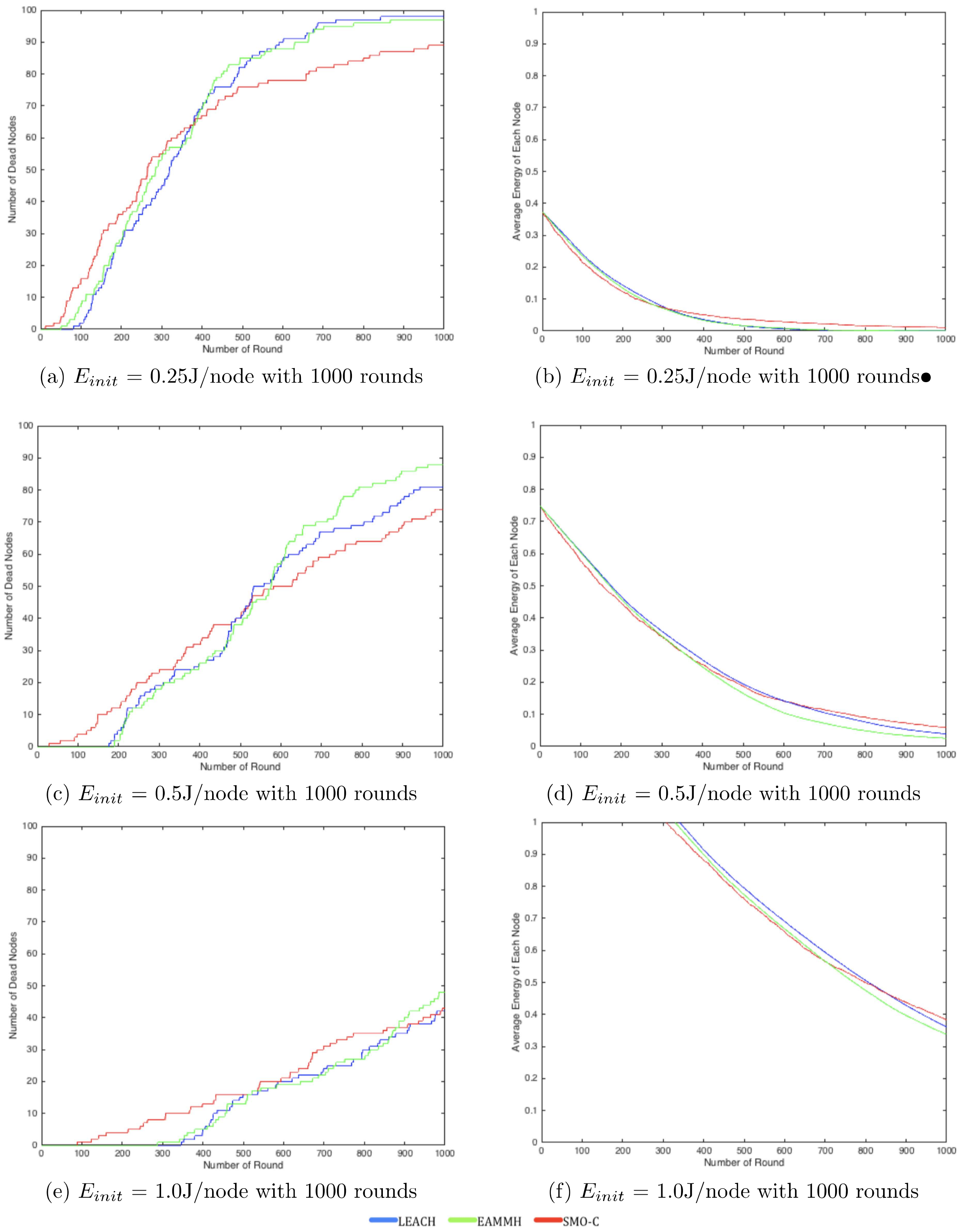
SMO-C is a self-organizing, easy to interpret and adaptive clustering protocol. The associate cluster heads are the ones with best fitness scores and located within the organized local clusters. In addition, SMO-C performs local data fusion and minimum energy transmission to send data from the clusters to the base station with the optimal route, further enhancing more alive nodes and reducing energy dissipation.

In a previous study, the cluster heads in LEACH are unevenly distributed. In some areas, there is lack of cluster head coverage and some of the sensor nodes are placed far away from the cluster heads. To achieve better routes, this situation has been considered in SMO-C. During our selection of cluster heads, we can get better solutions compared with LEACH and EAMMH. Our experimental results show that with better cluster-head locations we can obviously increase the node's lifetime in a larger number of rounds and decrease the energy loss of communication. Figure 5.4(a)(c)(e) indicates the duration of most nodes is prolonged in SMO-C; it has the lowest number of dead nodes in various initial energy settings. The sensor nodes can last longer when given limited initial energy and prolonged overall network lifetime. It also can be inferred from Figure 5.4(b)(d)(f) that the average energy of each node remained in SMO-C is higher than LEACH and EAMMH after multiple iterations. In Spider Monkey social behaviors, monkeys do not maintain fixed size of clusters throughout their foraging process, which means the percentage of cluster heads (P_{CH}) in routing protocols will not affect the performance in SMO-C. Hence, our experiments carry a specific amount ($P_{CH} = 10\%$) of cluster heads.

5.5.2 Performance Comparison with RMO Approach

In literature, Rhesus Macaque Optimization (RMO) is mainly designed based on the LEACH strategy. In some sensing areas, the nodes are positioned at a great distance from the cluster heads and cannot be replaced or recharged regularly. To improve the full network performance such situations have been considered in SMOCH, where the sensor nodes elevate themselves as cluster heads according to the state-of-the-art Spider Monkey Optimization formulation. The associated cluster heads are located within the structured local clusters with best fitness scores. SMOCH is a decentralized, self-organizing and interpretable clustering protocol. Moreover, SMOCH performs local data fusion and minimum energy transmission, further enhancing more alive nodes and reducing energy dissipation [39].

Figure 5.5 indicates the average retained energy of each node in SMOCH is greater than RMO in each round. It also can be inferred from Figure 5.6 that the duration of



(a) $E_{init} = 0.25$J/node with 1000 rounds

(b) $E_{init} = 0.25$J/node with 1000 rounds●

(c) $E_{init} = 0.5$J/node with 1000 rounds

(d) $E_{init} = 0.5$J/node with 1000 rounds

(e) $E_{init} = 1.0$J/node with 1000 rounds

(f) $E_{init} = 1.0$J/node with 1000 rounds

Figure 5.4. Performance comparison on LEACH, EAMMH, and SMO-C protocols with different initial energy for the sensors.

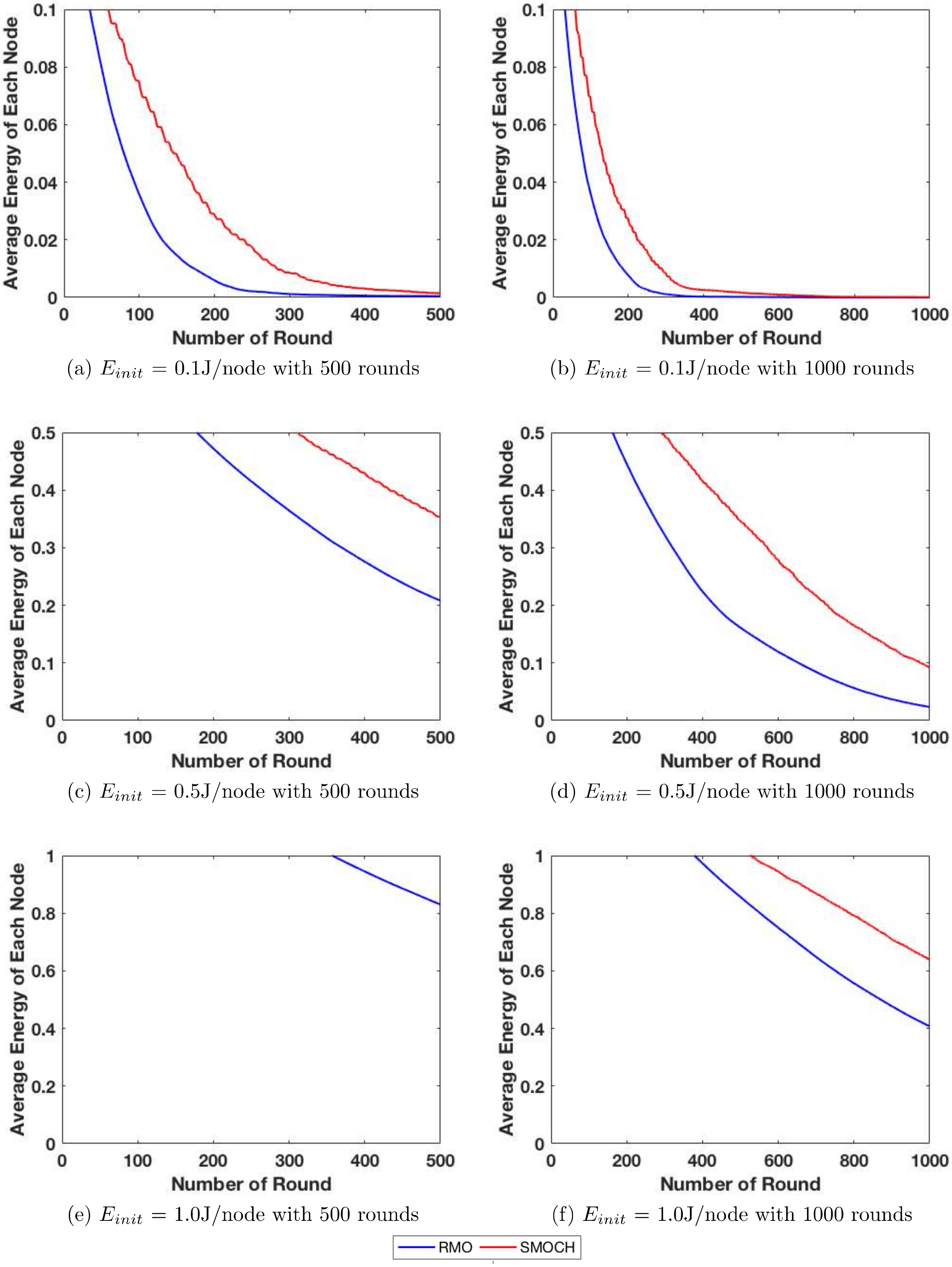
most sensor nodes is prolonged in the proposed SMOCH and it has the lowest number of dead nodes in the long run. The sensor nodes can last longer when given limited initial energy and extend the overall network lifetime. During the cluster head selection step, we achieve better results compared with Rhesus Macaque Optimization (RMO). Our simulation outcomes show that with improved cluster-head locations, we can observably decrease the energy consumption for communication and increase the network lifetime in a larger number of rounds. Contrasting with the other clustering protocols, SMOCH and RMO protocols focus on decreasing the energy consumed at the set-up phase not the steady phase. The experimental results show that the proposed SMOCH approach is self-organized, scalable and can be easily adapted in wireless sensor networks.

The monkey-inspired optimization algorithms require less computation time than conventional algorithms [25]. Our proposed SMOCH approach improves the exploration and exploitation capabilities of the search space. We believe the SMO based routing protocol will provide better performance due to the following reasons:

1. Easy to design and interpret: it is a non-parameteric optimization algorithm which means no manual parameter setting required [59].

2. Capability in multi-path model: swarms are divided into multiple groups; all groups exchange information and intelligence to optimize the routing solutions [26].

3. Self-organization behavior: all swarms act at the same time and there is no central coordinator [40].



(a) $E_{init} = 0.1$J/node with 500 rounds

(b) $E_{init} = 0.1$J/node with 1000 rounds

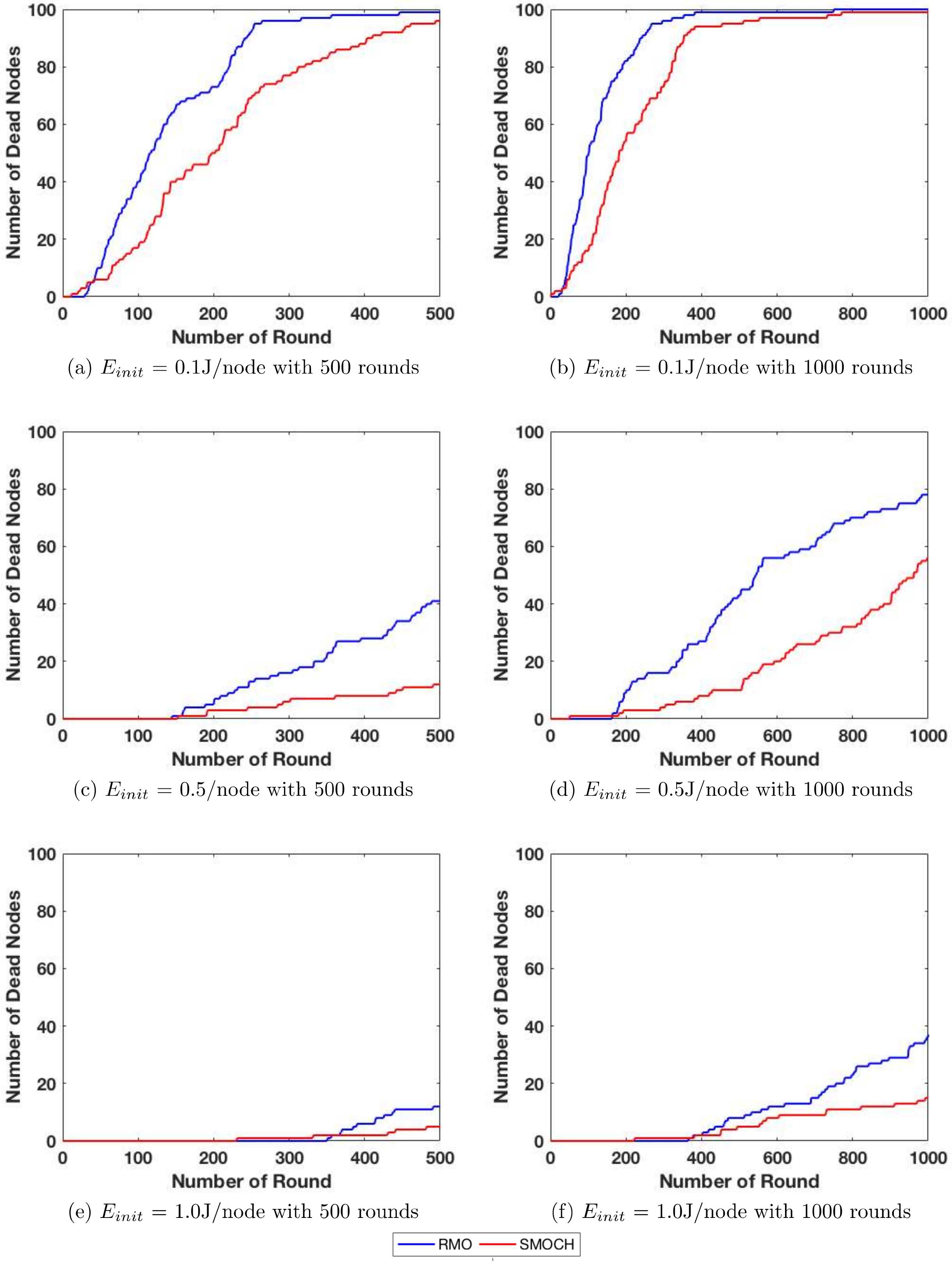
(c) $E_{init} = 0.5$J/node with 500 rounds

(d) $E_{init} = 0.5$J/node with 1000 rounds

(e) $E_{init} = 1.0$J/node with 500 rounds

(f) $E_{init} = 1.0$J/node with 1000 rounds

Figure 5.5. Average Remained Energy of RMO and SMOCH



(a) E_{init} = 0.1J/node with 500 rounds

(b) E_{init} = 0.1J/node with 1000 rounds

(c) E_{init} = 0.5/node with 500 rounds

(d) E_{init} = 0.5J/node with 1000 rounds

(e) E_{init} = 1.0J/node with 500 rounds

(f) E_{init} = 1.0J/node with 1000 rounds

Figure 5.6. Number of Dead Nodes of RMO and SMOCH

CHAPTER 6

CONCLUSION

In cancer classification and prediction, feature selection is an important process – selecting the optimal subset of relevant features or useful data for further study and prediction. Biologically inspired computing has successfully been used in many tasks that need simplicity in computation, optimized intelligent search and machine learning techniques. In this dissertation, we first propose a rule based feature selection and elimination approach, Top Discriminating Pairs (TDP); which aims to reveal which features are highly ranked according to their discrimination power. We compare the proposed approach with the traditional Top Scoring Pairs (TSP) method as the baseline on various artificial and real datasets. This work provides a new effective method for feature selection and dimensionality reduction in machine learning.

Next, we considered Swarm Intelligence based methods which mimic the social behaviors of natural insects or artificial systems. We presented a comprehensive study of the recent applications of Swarm Intelligence (SI) for optimizing feature selection process in Microarray data for human cancer classification and prediction. Then we introduced our Spider Monkey Optimization (SMO) based feature selection approach that has the advantages of in reducing irrelevant genes and improving classification accuracy. The results show that our SMO feature selector combining all three classifiers achieved the best accuracy scores on most of the test data sets.

Furthermore, we extended the ability of SMO in other fields such as Wireless Sensor Networks. WSN Our approach shows great potential and possibilities to provide optimization strategies, handle large-scale networks and avoid resource constraints. In this study, we successfully formulated the mathematical model according to spider monkeys' social behavior patterns and improved the traditional routing protocols in term of low-energy consumption and overall system quality of the network. The experimental results show that our approach is self-organized, scalable and can be easily adapted to the wireless sensor networks.

Our main contributions are,

1. We present a new method in machine learning that improves the prediction accuracy with less number of features involved while still maintaining robustness and interpretability.

2. We prove that swarm intelligence based algorithms are able to avoid local minimums and search for global optimal solution more efficiently.

3. We construct spider monkey optimization based routing protocol that advances wireless sensor networks mechanism in the direction of optimal solution.

4. We adopt spider monkey optimization algorithm that advances feature space reduction in classification and prediction.

5. We provide evidence from real-world applications that our methods provide significant advantages in accuracy and interpretability.

To this end we believe that our methods presented in this dissertation would be helpful in medical diagnosis as well as for further research. We hope that this work will motivate algorithm developers and scientists to take various techniques into account when working on solving optimization problems in different fields.

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] H. A. Abbass. Mbo: Marriage in honey bees optimization-a haplometrosis polygynous swarming approach. In *Evolutionary Computation, 2001. Proceedings of the 2001 Congress on*, volume 1, pages 207–214. IEEE, 2001.

[2] R. Agrawal, R. Srikant, et al. Fast algorithms for mining association rules. In *Proc. 20th int. conf. very large data bases, VLDB*, volume 1215, pages 487–499, 1994.

[3] A. S. Ahmed, M. A. Attia, N. M. Hamed, and A. Y. Abdelaziz. Comparison between genetic algorithm and whale optimization algorithm in fault location estimation in power systems. In *Power Systems Conference (MEPCON), 2017 Nineteenth International Middle East*, pages 631–637. IEEE, 2017.

[4] M. M. Ahmed, E. H. Houssein, A. E. Hassanien, A. Taha, and E. Hassanien. Maximizing lifetime of wireless sensor networks based on whale optimization algorithm. In *International Conference on Advanced Intelligent Systems and Informatics*, pages 724–733. Springer, 2017.

[5] A. A. Al-Azza, A. A. Al-Jodah, and F. J. Harackiewicz. Spider monkey optimization: A novel technique for antenna optimization. *IEEE Antennas and Wireless Propagation Letters*, 15:1016–1019, 2016.

[6] W. Andaru, I. Syarif, and A. R. Barakbah. Feature selection software development using artificial bee colony on dna microarray data. In *Knowledge Creation and Intelligent Computing (IES-KCIC), 2017 International Electronics Symposium on*, pages 6–11. IEEE, 2017.

[7] B. Antal and A. Hajdu. An ensemble-based system for automatic screening of diabetic retinopathy. *Knowledge-based systems*, 60:20–27, 2014.

[8] A. M. Anter, A. E. Hassanien, M. A. ElSoud, and T.-H. Kim. Feature selection approach based on social spider algorithm: case study on abdominal ct liver tumor. In *2015 Seventh International Conference on Advanced Communication and Networking (ACN)*, pages 89–94. IEEE, 2015.

[9] A. M. Anter and A. E. Hassenian. Normalized multiple features fusion based on pca and multiple classifiers voting in ct liver tumor recognition. In *Advances in Soft Computing and Machine Learning in Image Processing*, pages 113–129. Springer, 2018.

[10] A. Askarzadeh and A. Rezazadeh. A new heuristic optimization algorithm for modeling of proton exchange membrane fuel cell: bird mating optimizer. *International Journal of Energy Research*, 37(10):1196–1204, 2013.

[11] T. Back. *Evolutionary algorithms in theory and practice: evolution strategies, evolutionary programming, genetic algorithms.* Oxford university press, 1996.

[12] J. C. Bansal, H. Sharma, S. S. Jadon, and M. Clerc. Spider monkey optimization algorithm for numerical optimization. *Memetic computing*, 6(1):31–47, 2014.

[13] I. Barandiaran. The random subspace method for constructing decision forests. *IEEE transactions on pattern analysis and machine intelligence*, 20(8), 1998.

[14] B. Basturk. An artificial bee colony (abc) algorithm for numeric function optimization. In *IEEE Swarm Intelligence Symposium, Indianapolis, IN, USA, 2006*, 2006.

[15] D. G. Beer, S. L. Kardia, C.-C. Huang, T. J. Giordano, A. M. Levin, D. E. Misek, L. Lin, G. Chen, T. G. Gharib, D. G. Thomas, et al. Gene-expression profiles predict survival of patients with lung adenocarcinoma. *Nature medicine*, 8(8):816, 2002.

[16] L. Bi, Y. Li, X.-Q. Di, and Y. Zhang. Analyzing behavior of the social media users through swarm intelligence perspective. In *Computer Science and Network Technology (ICCSNT), 2016 5th International Conference on*, pages 340–344. IEEE, 2016.

[17] M. Canayaz and M. Demir. Feature selection with the whale optimization algorithm and artificial neural network. In *Artificial Intelligence and Data Processing Symposium (IDAP), 2017 International*, pages 1–5. IEEE, 2017.

[18] V. Chaurasia, S. Pal, and B. Tiwari. Prediction of benign and malignant breast cancer using data mining techniques. *Journal of Algorithms & Computational Technology*, 12(2):119–126, 2018.

[19] C. A. C. Coello, G. B. Lamont, D. A. Van Veldhuizen, et al. *Evolutionary algorithms for solving multi-objective problems*, volume 5. Springer, 2007.

[20] C. Cortes and V. Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.

[21] C. Creighton and S. Hanash. Mining gene expression databases for association rules. *Bioinformatics*, 19(1):79–86, 2003.

[22] E. Cuevas, M. Cienfuegos, D. Zaldívar, and M. Pérez-Cisneros. A swarm optimization algorithm inspired in the behavior of the social-spider. *Expert Systems with Applications*, 40(16):6374–6384, 2013.

[23] P. Cui, Y. Dong, H. Liu, D. Rajan, E. Olinick, and J. Camp. Whitemesh: Leveraging white spaces in wireless mesh networks. In *Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt), 2016 14th International Symposium on*, pages 1–7. IEEE, 2016.

[24] A. Datta and S. Nandakumar. A survey on bio inspired meta heuristic based clustering protocols for wireless sensor networks. *IOP Conference Series: Materials Science and Engineering*, 263(5), 2017.

[25] R. Devi and S. Sathya. Monkey behavior based algorithms-a survey. *International Journal of Intelligent Systems and Applications*, 9(12), 2017.

[26] J. Dhar and S. Arora. Designing fuzzy rule base using spider monkey optimization algorithm in cooperative framework. *Future Computing and Informatics Journal*, 2(1):31–38, 2017.

[27] M. Dorigo and M. Birattari. Ant colony optimization. In *Encyclopedia of machine learning*, pages 36–39. Springer, 2011.

[28] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern classification.* John Wiley & Sons, 2012.

[29] R. Eberhart and J. Kennedy. A new optimizer using particle swarm theory. In *Micro Machine and Human Science, 1995. MHS'95., Proceedings of the Sixth International Symposium on*, pages 39–43. IEEE, 1995.

[30] E. M. El Houby, N. I. Yassin, and S. Omran. A hybrid approach from ant colony optimization and k-nearest neighbor for classifying datasets using selected features. *Informatica*, 41(4), 2017.

[31] E. Emary, H. M. Zawbaa, and A. E. Hassanien. Binary grey wolf optimization approaches for feature selection. *Neurocomputing*, 172:371–381, 2016.

[32] J. Faber and L. M. Fonseca. How sample size influences research outcomes. *Dental press journal of orthodontics*, 19(4):27–29, 2014.

[33] A. Frank and A. Asuncion. UCI machine learning repository, 2010.

[34] M. Galar, A. Fernandez, E. Barrenechea, H. Bustince, and F. Herrera. A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42(4):463–484, 2012.

[35] A. H. Gandomi and A. H. Alavi. Krill herd: a new bio-inspired optimization algorithm. *Communications in Nonlinear Science and Numerical Simulation*, 17(12):4831–4845, 2012.

[36] L. Gao, M. Ye, and C. Wu. Cancer classification based on support vector machine optimized by particle swarm optimization and artificial bee colony. *Molecules*, 22(12):2086, 2017.

[37] D. Geman, C. d'Avignon, D. Q. Naiman, and R. L. Winslow. Classifying gene expression profiles from pairwise mrna comparisons. *Statistical applications in genetics and molecular biology*, 3(1):1–19, 2004.

[38] C. Goutte and E. Gaussier. A probabilistic interpretation of precision, recall and f-score, with implication for evaluation. In *European Conference on Information Retrieval*, pages 345–359. Springer, 2005.

[39] T. Gui, C. Ma, F. Wang, J. Li, and D. E. Wilkins. A novel cluster-based routing protocol wireless sensor networks using spider monkey optimization. In *Industrial Electronics Society, IECON 2016-42nd Annual Conference of the IEEE*, pages 5657–5662. IEEE, 2016.

[40] T. Gui, C. Ma, F. Wang, and D. E. Wilkins. Survey on swarm intelligence based routing protocols for wireless sensor networks: An extensive study. In *Industrial Technology (ICIT), 2016 IEEE International Conference on*, pages 1944–1949. IEEE, 2016.

[41] T. Gui, F. Wang, C. Ma, and D. E. Wilkins. On cluster head selection in monkey-inspired optimization based routing protocol for wsns. In *International Conference on Computing, Networking and Communications, ICNC 2019, Honolulu, HI, USA, February 18-21, 2019*, pages 126–130. IEEE, 2019.

[42] K. Gupta, K. Deep, and J. C. Bansal. Spider monkey optimization algorithm for constrained optimization problems. *Soft Computing*, 21(23):6933–6962, 2017.

[43] A. I. Hafez, A. E. Hassanien, H. M. Zawbaa, and E. Emary. Hybrid monkey algorithm with krill herd algorithm optimization for feature selection. In *Computer Engineering Conference (ICENCO), 2015 11th International*, pages 273–277. IEEE, 2015.

[44] Z. He and W. Yu. Stable feature selection for biomarker discovery. *Computational biology and chemistry*, 34(4):215–225, 2010.

[45] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan. Energy-efficient communication protocol for wireless microsensor networks. In *System sciences, 2000. Proceedings of the 33rd annual Hawaii international conference on*, pages 10–pp. IEEE, 2000.

[46] S. M. K. Heris.

[47] T. K. Ho. Random decision forests. In *Document analysis and recognition, 1995., proceedings of the third international conference on*, volume 1, pages 278–282. IEEE, 1995.

[48] T. K. Ho and M. Basu. Complexity measures of supervised classification problems. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (3):289–300, 2002.

[49] R. Jaswal et al. Earth observation and satellite imagery using spider monkey optimization (smo). *International Journal of Advanced Research in Computer Science*, 8(7), 2017.

[50] H. Jemal, Z. Kechaou, and M. B. Ayed. Swarm intelligence and multi agent system in healthcare. In *Soft Computing and Pattern Recognition (SoCPaR), 2014 6th International Conference of*, pages 423–427. IEEE, 2014.

[51] C. Jin and S.-W. Jin. Gene selection approach based on improved swarm intelligent optimisation algorithm for tumour classification. *IET systems biology*, 10(3):107–115, 2016.

[52] A. R. Kammerdiner, A. Mucherino, and P. M. Pardalos. Application of monkey search meta-heuristic to solving instances of the multidimensional assignment problem. In *Optimization and Cooperative Control Strategies*, pages 385–397. Springer, 2009.

[53] D. Karaboga. An idea based on honey bee swarm for numerical optimization. Technical report, Technical report-tr06, Erciyes university, engineering faculty, computer engineering department, 2005.

[54] I. Kassabalidis, M. El-Sharkawi, R. Marks, P. Arabshahi, and A. Gray. Swarm intelligence for routing in communication networks. In *Global Telecommunications Conference, 2001. GLOBECOM'01. IEEE*, volume 6, pages 3613–3617. IEEE, 2001.

[55] A. Kaveh and N. Farhoudi. A new optimization method: dolphin echolocation. *Advances in Engineering Software*, 59:53–70, 2013.

[56] J. Kennedy. Particle swarm optimization. In *Encyclopedia of machine learning*, pages 760–766. Springer, 2011.

[57] L. A. K. Krzysztof J. Cios. UCI machine learning repository, 2001.

[58] S. Kumar and S. Kusuma. Clustering protocol for wireless sensor networks based on rhesus macaque (macaca mulatta) animal's social behavior. *International Journal of Computer Applications*, 87(8), 2014.

[59] S. Kumar, V. K. Sharma, and R. Kumari. Self-adaptive spider monkey optimization algorithm for engineering optimization problems. *JIMS8I-International Journal of Information Communication and Computing Technology*, 2(2):96–107, 2014.

[60] K. Lenin, B. R. Reddy, and M. S. Kalavathi. Modified monkey optimization algorithm for solving optimal reactive power dispatch problem. *Indonesian Journal of Electrical Engineering and Informatics (IJEEI)*, 3(2):55–62, 2015.

[61] J. Li and H. Liu. Challenges of feature selection for big data analytics. *IEEE Intelligent Systems*, 32(2):9–15, 2017.

[62] X. Li. A new intelligent optimization-artificial fish swarm algorithm. *Doctor thesis, Zhejiang University of Zhejiang, China*, 2003.

[63] X. Li, M. Li, and M. Yin. Multiobjective ranking binary artificial bee colony for gene selection problems using microarray datasets. *IEEE/CAA Journal of Automatica Sinica*, 2016.

[64] H. Liu, A. Nayak, and I. Stojmenović. Fault-tolerant algorithms/protocols in wireless sensor networks. In *Guide to Wireless Sensor Networks*, pages 261–291. Springer, 2009.

[65] Y. Liu, J. Chen, and Y.-j. Zhan. Local patches alignment embedding based localization for wireless sensor networks. *Wireless personal communications*, 70(1):373–389, 2013.

[66] X. Lu and Y. Zhou. A novel global convergence algorithm: bee collecting pollen algorithm. In *International Conference on Intelligent Computing*, pages 518–525. Springer, 2008.

[67] S. Ma, X. Song, and J. Huang. Supervised group lasso with applications to microarray data analysis. *BMC bioinformatics*, 8(1):60, 2007.

[68] S. Mirjalili and A. Lewis. The whale optimization algorithm. *Advances in Engineering Software*, 95:51–67, 2016.

[69] S. Mirjalili, S. M. Mirjalili, and A. Lewis. Grey wolf optimizer. *Advances in engineering software*, 69:46–61, 2014.

[70] F. Mohamed, M. AbdelNasser, K. Mahmoud, and S. Kamel. Accurate economic dispatch solution using hybrid whale-wolf optimization method. In *Power Systems Conference (MEPCON), 2017 Nineteenth International Middle East*, pages 922–927. IEEE, 2017.

[71] S. Montani and R. Bellazzi. Supporting decisions in medical applications: the knowledge management perspective. *International journal of medical informatics*, 68(1-3):79–90, 2002.

[72] J. M. Moosa, R. Shakur, M. Kaykobad, and M. S. Rahman. Gene selection for cancer classification with the help of bees. *BMC medical genomics*, 9(2):47, 2016.

[73] A. Mucherino and O. Seref. Monkey search: a novel metaheuristic search for global optimization. In *AIP conference proceedings*, volume 953, pages 162–173. AIP, 2007.

[74] M. R. Mundada, V. CyrilRaj, and T. Bhuvaneswari. Energy aware multi-hop multipath hierarchical (eammh) routing protocol for wireless sensor networks. *European Journal of Scientific Research*, 88(4):520–530, 2012.

[75] N. Nayak, M. S. Mahali, I. Majumder, and R. K. Jena. Dynamic stability improvement of vsc-hvdc connected multi machine power system by spider monkey optimization based pi controller. In *Electrical, Electronics, and Optimization Techniques (ICEEOT), International Conference on*, pages 152–157. IEEE, 2016.

[76] J. Ni, L. Wu, X. Fan, and S. X. Yang. Bioinspired intelligent algorithm and its applications for mobile robot control: a survey. *Computational intelligence and neuroscience*, 2016:1, 2016.

[77] R. Oftadeh, M. Mahjoob, and M. Shariatpanahi. A novel meta-heuristic optimization algorithm inspired by group hunting of animals: Hunting search. *Computers & Mathematics with Applications*, 60(7):2087–2098, 2010.

[78] W.-T. Pan. A new fruit fly optimization algorithm: taking the financial distress model as an example. *Knowledge-Based Systems*, 26:69–74, 2012.

[79] R. Parry, W. Jones, T. Stokes, J. Phan, R. Moffitt, H. Fang, L. Shi, A. Oberthuer, M. Fischer, W. Tong, et al. k-nearest neighbor models for microarray gene expression analysis and clinical outcome prediction. *The pharmacogenomics journal*, 10(4):292, 2010.

[80] A.-S. K. Pathan, H.-W. Lee, and C. S. Hong. Security in wireless sensor networks: issues and challenges. In *Advanced Communication Technology, 2006. ICACT 2006. The 8th International Conference*, volume 2, pages 6–pp. IEEE, 2006.

[81] G. Piatetsky-Shapiro. Discovery, analysis, and presentation of strong rules. *Knowledge discovery in databases*, pages 229–238, 1991.

[82] P. C. Pinto, T. A. Runkler, and J. M. Sousa. Wasp swarm algorithm for dynamic max-sat problems. In *International Conference on Adaptive and Natural Computing Algorithms*, pages 350–357. Springer, 2007.

[83] J. Prentzas and I. Hatzilygeroudis. Categorizing approaches combining rule-based and case-based reasoning. *Expert Systems*, 24(2):97–122, 2007.

[84] J. R. Quinlan. Simplifying decision trees. *International journal of man-machine studies*, 27(3):221–234, 1987.

[85] M. Roth. Termite: A swarm intelligent routing algorithm for mobile wireless ad-hoc networks. 2005.

[86] M. Saleem, G. A. Di Caro, and M. Farooq. Swarm intelligence based routing protocol for wireless sensor networks: Survey and future directions. *Information Sciences*, 181(20):4597–4624, 2011.

[87] Y. Sasaki et al. The truth of the f-measure. 2007.

[88] M. Sharawi, H. M. Zawbaa, and E. Emary. Feature selection approach based on whale optimization algorithm. In *Advanced Computational Intelligence (ICACI), 2017 Ninth International Conference on*, pages 163–168. IEEE, 2017.

[89] F. V. Sharbaf, S. Mosafer, and M. H. Moattar. A hybrid gene selection approach for microarray data classification using cellular learning automata and ant colony optimization. *Genomics*, 107(6):231–238, 2016.

[90] P. Shi, S. Ray, Q. Zhu, and M. A. Kon. Top scoring pairs for feature selection in machine learning and applications to cancer outcome prediction. *BMC bioinformatics*, 12(1):375, 2011.

[91] Y. Shiqin, J. Jianjun, and Y. Guangxing. A dolphin partner optimization. In *Intelligent Systems, 2009. GCIS'09. WRI Global Congress on*, volume 1, pages 124–128. IEEE, 2009.

[92] D. Storcheus, A. Rostamizadeh, and S. Kumar. A survey of modern questions and challenges in feature extraction. In *Feature Extraction: Modern Questions and Challenges*, pages 1–18, 2015.

[93] S. K. Suguna and S. Maheswari. Performance analysis of feature extraction and selection of region of interest by segmentation in mammogram images between the existing metaheuristic algorithms and monkey search optimization (mso). *WSEAS Transactions on Information Science and Applications*, 11:72–82, 2014.

[94] S. Tabakhi, A. Najafi, R. Ranjbar, and P. Moradi. Gene selection for microarray data classification using a novel ant colony optimization. *Neurocomputing*, 168:1024–1036, 2015.

[95] A. C. Tan, D. Q. Naiman, L. Xu, R. L. Winslow, and D. Geman. Simple decision rules for classifying human cancers from gene expression profiles. *Bioinformatics*, 21(20):3896–3904, 2005.

[96] R. Tibshirani, T. Hastie, B. Narasimhan, and G. Chu. Diagnosis of multiple cancer types by shrunken centroids of gene expression. *Proceedings of the National Academy of Sciences*, 99(10):6567–6572, 2002.

[97] G. K. Venayagamoorthy and R. G. Harley. Swarm intelligence for transmission system control. 2007.

[98] P. H. Venkatrao and S. S. Damodar. Hwfusion: Holoentropy and sp-whale optimisation-based fusion model for magnetic resonance imaging multimodal image fusion. *IET Image Processing*, 12(4):572–581, 2017.

[99] G.-G. Wang. Moth search algorithm: a bio-inspired metaheuristic algorithm for global optimization problems. *Memetic Computing*, pages 1–14, 2016.

[100] G.-G. Wang, X. Zhao, and S. Deb. A novel monarch butterfly optimization with greedy strategy and self-adaptive. In *Soft Computing and Machine Intelligence (ISCMI), 2015 Second International Conference on*, pages 45–50. IEEE, 2015.

[101] Y. Wang, J. G. Klijn, Y. Zhang, A. M. Sieuwerts, M. P. Look, F. Yang, D. Talantov, M. Timmermans, M. E. Meijer-van Gelder, J. Yu, et al. Gene-expression profiles to predict distant metastasis of lymph-node-negative primary breast cancer. *The Lancet*, 365(9460):671–679, 2005.

[102] Y. Wang, B. Li, T. Weise, J. Wang, B. Yuan, and Q. Tian. Self-adaptive learning based particle swarm optimization. *Information Sciences*, 181(20):4515–4538, 2011.

[103] Z. Wang and M. Schwager. Force-amplifying n-robot transport system (force-ants) for cooperative planar manipulation without communication. *The International Journal of Robotics Research*, 35(13):1564–1586, 2016.

[104] W. A. Watkins and W. E. Schevill. Aerial observation of feeding behavior in four baleen whales: Eubalaena glacialis, balaenoptera borealis, megaptera novaeangliae, and balaenoptera physalus. *Journal of Mammalogy*, 60(1):155–163, 1979.

[105] F. Wilcoxon. Individual comparisons by ranking methods. *Biometrics bulletin*, 1(6):80–83, 1945.

[106] W. H. Wolberg and O. L. Mangasarian. Multisurface method of pattern separation for medical diagnosis applied to breast cytology. *Proceedings of the national academy of sciences*, 87(23):9193–9196, 1990.

[107] M. Xi, J. Sun, L. Liu, F. Fan, and X. Wu. Cancer feature selection and classification using a binary quantum-behaved particle swarm optimization and support vector machine. *Computational and mathematical Methods in Medicine*, 2016, 2016.

[108] C. Yang, X. Tu, and J. Chen. Algorithm of marriage in honey bees optimization based on the wolf pack search. In *Intelligent Pervasive Computing, 2007. IPC. The 2007 International Conference on*, pages 462–467. IEEE, 2007.

[109] X.-S. Yang. Firefly algorithm, stochastic test functions and design optimisation. *International Journal of Bio-Inspired Computation*, 2(2):78–84, 2010.

[110] X.-S. Yang. A new metaheuristic bat-inspired algorithm. In *Nature inspired cooperative strategies for optimization (NICSO 2010)*, pages 65–74. Springer, 2010.

[111] X.-S. Yang and S. Deb. Cuckoo search via lévy flights. In *Nature & Biologically Inspired Computing, 2009. NaBIC 2009. World Congress on*, pages 210–214. IEEE, 2009.

[112] J. Ye and J. Liu. Sparse methods for biomedical data. *ACM Sigkdd Explorations Newsletter*, 14(1):4–15, 2012.

[113] E. C. Yip, K. S. Powers, and L. Avilés. Cooperative capture of large prey solves scaling challenge faced by spider societies. *Proceedings of the National Academy of Sciences*, 2008.

[114] N. A. Zamri, B. Thangavel, N. A. Ab Aziz, and N. H. A. Aziz. Review on the usage of swarm intelligence in gene expression data. In *International Conference for Innovation in Biomedical Engineering and Life Sciences*, pages 153–160. Springer, 2017.

[115] Z. Zhang, A. Teo, B. C. Ooi, and K.-L. Tan. Mining deterministic biclusters in gene expression data. In *Bioinformatics and Bioengineering, 2004. BIBE 2004. Proceedings. Fourth IEEE Symposium on*, pages 283–290. IEEE, 2004.

[116] Z. A. Zhao and H. Liu. *Spectral feature selection for data mining.* CRC Press, 2011.

[117] M. Zwitter and y. . M. Soklic.

VITA

Tina Gui
Email: tgui@go.olemiss.edu

WORK EXPERIENCE

Bacardi | *Miami, FL*
Senior Manager, Lead Data Scientist — Aug 2021 - Present
Anheuser-Busch InBev | *Boston, MA*
Global Manager, Analytics — Jul 2018 - Jul 2021
Computer Scientist — Aug 2016 - Jun 2018
National Science Foundation | *Taipei, Taiwan*
EAPSI Research Fellow — Jun 2016 - Aug 2016
University of Mississippi | *University, MS*
Graduate Research Assistant — Jun 2013 - Jun 2016
Graduate Instructor — Aug 2012 - Dec 2015
Summer Research Assistant — Jun 2012 - Aug 2012
Graduate Teaching Assistant — Aug 2011 - May 2012

EDUCATION

University of Mississippi | *University, MS* — 2019
Ph.D. in Computer Science — GPA: 3.75/4.0
University of Mississippi | *University, MS* — 2014
M.Sc. in Computer Science — GPA: 3.70/4.0
California State University | *Bakersfield, CA* — 2011
B.Sc. in Computer Science — GPA: 3.29/4.0

AWARDS AND ACHIEVEMENTS

NSF East Asia and Pacific Summer Institutes (EAPSI) Fellowship — 2016
Third Place, Poster Award, 13th annual McBios conference — 2016
Travel Fellowship Award, 11th annual McBios conference — 2015
Travel Fellowship Award, 22nd annual ISMB conference — 2014
First Place, Poster Award, 10th annual McBios conference — 2013
Graduate Fellowship, University of Mississippi — 2011 - 2015
Outstanding and Dean List Student, California State University — 2009 - 2011

PUBLICATIONS

Journal

1. Y. She, X. Wang, T. Gui, and J. Cai, "Lawn Plant Identification and Segmentation based on Least Squares Support Vector Machine and Multifeature Fusion". Journal of Electronic

Imaging, 2019, 28(2) 023034.
2. B. Chen, T. Gui, S. Cheng, and X. Li, ”Research on Operation Cost in Urban Transportation System Based on Stopping Distance and Vehicle Density.” Journal of Simulation. 2019, (01):67-72.
3. X. Wang, Y. Ge, T. Gui, S. Zhang, Qi Wang. ”Research on optimum design of hydraulic flat transporter frame based on response surface method,” China Mechanical Engineering, 2013, (16):2261-2265.
4. X. Wang, P. Zhao, T. Gui, S. Zhang, Z. Li, ”Research of a new brake energy recovery system of the light electric car,” Journal of Jiangsu University of Science and Technology(Natural Science Edition), 2013, (02):129-136.
5. X. Wang, W. Zhu, Q. Li, and T. Gui. ”Application of the GPS technology in the measurement of the vehicle parameters,” Journal of Guangxi University(Natural Science Edition), 2011, (02):241-245.

Conference Papers
1. T. Gui, F. Wang, C. Ma, and D.E. Wilkins, “On Cluster Head Selection in Monkey-inspired Optimization based Routing Protocol for WSNs”. International Conference on Computing, Networking and Communications (ICNC), 2019, pp. 126-130.
2. C. Ma, T. Gui, X. Dang, Y. Chen, and D. Wilkins. ”Integration of Cancer Data through Multiple Mixed Graphical Model.” In Proceedings of the 2018 ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics (BCB), Washington DC, USA, 2018, pp. 341-350.
3. T. Gui, C. Ma, F. Wang, Jinyang Li and D.E. Wilkins, ”A novel cluster-based routing protocol wireless sensor networks using Spider Monkey Optimization,” IECON 2016 - 42nd Annual Conference of the IEEE Industrial Electronics Society, Florence, Italy, 2016, pp. 5657-5662.
4. T. Gui, C. Ma, F. Wang and D.E. Wilkins, ”Survey on swarm intelligence based routing protocols for wireless sensor networks: An extensive study,” 2016 IEEE International Conference on Industrial Technology (ICIT), Taipei, Taiwan, 2016, pp. 1944-1949.
5. C. Ma, Z. Zhao, T. Gui, Y. Chen, X. Dang and D. Wilkins, ”A generative Bayesian model to identify cancer driver genes,” 2015 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), Washington, DC, USA, 2015, pp. 351-356.

Posters and Talks
1. T. Gui, C. Ma, J. Nakarmi, D.E. Wilkins, Y. Chen. 2016. “ARM-B: Mining Biclusters With Association Rules In Gene Expression Data Analysis”. MidSouth Computational Biology and Bioinformatics Society (MCBIOS) Conference.
2. T. Gui, Z. Zhao, DE Wilkins, Y. Chen. 2014. “A Pairwise Feature Selection Method for Gene Data Using Information Gain”. International Conference on Intelligent Systems for Molecular Biology (ISMB).
3. T. Gui, X. Nan, D.E. Wilkins, and Y. Chen. 2013. ”Classification and Feature Selection Using Hybrid Top Scoring Pairs on Microarray Data”. MidSouth Computational Biology and Bioinformatics Society (MCBIOS) Conference.