International Journal of Theoretical & Applied Computational Intelligence
https://ijtaci.com

Research Article

Enhanced Feature Selection for Imbalanced Microarray Cancer Gene Classification Using Chaotic Salp Swarm Algorithm

Bashar Shehu Aliyu[1], Jeremiah Isuwa[2], Abdulrazaq Abdulrahim[2], Mohammed Abdullahi[2] Ibrahim Hayatu Hassan[1]*, Tanbin Sikder Momi[3]

[1]Department of Computer Science, Ahmadu Bello University, Zaria-Nigeria

[2]Computer Science Department, Federal University of Kashere, Gombe-Nigeria

[3]School of Clinical Medicine, Zhengzhou University, Zhengzhou, Henan RP, China

[*]Corresponding Author: Ibrahim Hayatu Hassan. Email: ihhassan@abu.edu.ng

https://orcid.org/0000-0001-5522-8465

Received: 23/7/2025; 14/8/2025: Accepted: 03/09/2025; Published: 11/10/2025

https://doi.org/10.65278/IJTACI.2025.16

Abstract: The healthcare sector requires intelligent solutions to manage vast microarray data, but challenges like high dimensionality, data imbalance, and computational complexity persist. This study addresses these by handling imbalanced microarray cancer gene datasets using Synthetic Minority Oversampling Technique (SMOTE) and enhancing the Salp Swarm Algorithm (SSA) with sinusoidal chaotic map initialization for improved population and control parameter diversity. The enhanced SSA is combined with Chi-square and Mutual Information filter methods to select top-performing genes from Ovarian, Colon, and Leukemia datasets, followed by refinement based on minimal error. Key contributions include chaotic initialization for better exploration, SMOTE for balanced classification, and a novel minimal-error gene subset selection. Compared to state-of-the-art methods, proposed approach achieves competitive performance, with 100% accuracy and F1 score across datasets while reducing gene counts (e.g., 4 genes for Colon). This promises to enhance cancer diagnosis and treatment, enabling targeted therapies and personalized medicine for improved patient outcomes.

Keywords: Salp Swarm Algorithm (SSA); Filter Methods; Microarray Cancer Gene Selection; Imbalance Data; High Dimensional Data

1. Introduction

Microarray gene expression analysis stands at the forefront of modern molecular biology, offering a broad view into the detailed mechanisms governing genetic activity [1,2]. It allows researchers to examine the expression levels of thousands of genes simultaneously, unraveling the complex interplay between genetic material and cellular behavior [3]. However, amidst this wealth of data lies a daunting challenge: the need to discern signal from noise, and relevance from redundancy [4]. Here, microarray gene selection emerges as a valuable method, guiding researchers through the overwhelming path of gene expression profiles toward the identification of key attributes contributing to biological processes [5]. By strategically extracting vast arrays of genetic information into concise subsets of genes with predictive power, microarray gene selection not only illuminates the molecular underpinnings of diseases like cancer but also paves the way for targeted therapies and personalized medicine [6]. In this domain where precision is paramount and insights are invaluable, the quest for optimal gene subsets becomes not just a scientific endeavor but a transformative journey toward unlocking the mysteries encoded within our DNA.

Traditional methods such as brute force and heuristic approaches have been employed to tackle the microarray gene selection problem. However, despite their widespread use, these methods often fall short of providing optimal results [7]. Brute force methods exhaustively search through all possible combinations of genes, making them computationally expensive and impractical for high-dimensional datasets [29,50]. Additionally, they are prone to the curse of dimensionality, where the number of features increases exponentially with the dataset size, leading to sparse data distributions and increased computational complexity [8,9]. On the other hand, heuristic methods, while more computationally efficient, rely on predefined rules or strategies that may not always capture the intricate relationships and interactions present in biological datasets [10]. Overall, the limitations of these traditional methods highlight the need for more advanced and adaptable approaches to microarray gene selection.

The introduction of statistical filter methods marked a significant advancement in microarray gene selection, offering a more systematic and data-driven approach compared to traditional methods. This filter methods operate by evaluating each feature independently based on statistical measures of relevance or importance [11,12]. For example, methods like Chi-square [13], Relief [14], and Mutual Information [15] quantify the association between each gene and the target variable, such as disease status. By ranking genes according to their statistical scores, filter methods can identify the most informative features for downstream analysis [16]. However, despite their simplicity and computational efficiency, statistical filter methods have notable limitations [17]. One major drawback is their inability to capture feature dependencies or interactions, as they assess genes in isolation. This can lead to the selection of redundant or irrelevant features, diminishing the effectiveness of subsequent analysis [18].

To address this issue, wrapper methods have been introduced, which consider feature dependencies by evaluating subsets of genes rather than individual features. Unlike filter methods, wrapper methods assess the contribution of gene subsets collectively, taking into account feature interactions [19,20]. While wrapper methods offer greater flexibility and potentially higher accuracy compared to filter methods, they are computationally expensive, especially in high-dimensional datasets.

Authors in [4] Considering these drawbacks, the advent of Swarm Intelligence (SI) algorithms revolutionized the field of Feature Selection (FS) by mimicking the collective behaviors of social organisms to find optimal solutions [21]. These algorithms, including the Salp Swarm Algorithm (SSA) [22] Particle Swarm Optimization (PSO) [23], Genetic Algorithm (GA), Ant Colony Optimization (ACO) [24], and Bat Algorithm (BA) [25] among others, operate on the principle of collaboration among individual agents to

search the solution space efficiently. They have garnered widespread acceptance across various domains, ranging from engineering and civil engineering to biology and finance. In the context of microarray gene selection, SI algorithms have proven highly effective due to their ability to handle high-dimensional and complex datasets. These algorithms can efficiently explore the vast search space of gene combinations to identify optimal subsets that best discriminate between different classes, such as cancer subtypes or treatment responses. Furthermore, SI algorithms offer advantages such as robustness to noise, adaptability to dynamic environments, and parallel processing capabilities, making them well-suited for addressing the challenges inherent in microarray data analysis [26].

Among these SI algorithms, the SSA stands out for its simplicity, efficiency, and effectiveness in optimization tasks [27]. Inspired by the collective behavior of Salps, SSA employs simple rules for individual agents to adjust their positions based on local and global information exchange. This decentralized approach allows SSA to efficiently explore the solution space while maintaining diversity among solutions [28]

Nonetheless, despite its wide acceptance and recognition in solving large-scale optimization problems, SSA exhibits a limitation in its exploration ability, primarily attributed to the poor initialization of its population and control parameters using pseudo-random numbers [25]. This poor initialization hinders the algorithm's exploration capability, as it may lead to solutions trapped in local optima or premature convergence to suboptimal solutions [29]. Random initialization using pseudo-random numbers often results in sequences with discernible patterns or correlations, limiting the algorithm's ability to thoroughly explore the solution space [30].

However, the use of Chaotic Maps (CM) has revolutionized this aspect of SSA by offering a more effective approach to initializing both population and control parameters [2,14]. CMs, renowned for their inherent unpredictability, generate diverse and irregular sequences of numbers that promote exploration of the solution space [2]. By leveraging CM for initialization, SSA can overcome the limitations of random initialization and enhance its global search ability. This integration of CM not only facilitates a more thorough exploration of the solution space but also improves the algorithm's convergence behavior and solution quality [31].

Research endeavors are increasingly focusing on incorporating the strengths of filtering methods with the global search capabilities of SI algorithms to enhance the effectiveness of gene selection processes [9]. The integration of filtering methods and SI algorithms holds great promise in gene selection tasks. Filtering methods serve as effective preprocessing steps to identify promising gene subsets based on relevant criteria such as statistical significance or information gain [32]. These filtered subsets are then passed on to SI algorithms, which further explore the solution space to refine the selection and identify the most optimal gene subset. This combination leverages the strengths of both approaches, allowing for a more comprehensive search while mitigating the risk of getting stuck in local optima.

However, one of the challenges in this approach lies in determining the optimal size of the gene subset to pass to the SI algorithm after the filtering step. Selecting an inadequate subset size may limit the search space and overlook potentially relevant genes, leading to suboptimal solutions. Conversely, choosing an excessively large subset size may increase computational costs and hinder the efficiency of the SI algorithm. Thus, finding the right balance in subset size selection is crucial for maximizing the effectiveness of the combined filtering and SI approach in gene selection tasks.

Managing high-dimensional data is undoubtedly crucial; however, the challenge of dealing with imbalanced data is equally significant and requires attention. The issue of imbalanced data is prevalent in

various domains, including microarray gene expression data, and poses significant challenges for Machine Learning (ML) algorithms. Imbalanced data occurs when the distribution of classes in a dataset is highly skewed, with one class (the minority class) being significantly underrepresented compared to the others (the majority class) [33]. This imbalance can arise due to various factors such as the natural scarcity of certain events or errors in data collection processes. The imbalanced nature of the data can severely impact the performance of ML algorithms, particularly those designed to optimize overall accuracy [34]. In such scenarios, algorithms tend to exhibit a bias towards the majority class, leading to poor predictive performance for the minority class. As a result, the model may struggle to accurately identify and classify instances belonging to the minority class, which is often of greater interest in real-world applications, such as detecting rare diseases or identifying anomalies [35].

To address the challenges posed by imbalanced data, researchers have developed various techniques aimed at rebalancing the class distribution and improving the performance of ML algorithms [54]. One common approach is resampling, which involves either oversampling the minority class or undersampling the majority class [3]. Additionally, synthetic data generation techniques, such as SMOTE (Synthetic Minority Over-sampling Technique), create synthetic instances for the minority class to balance the class distribution. Other techniques include cost-sensitive learning, where different misclassification costs are assigned to different classes to encourage the model to prioritize the correct classification of the minority class [36].

The novelty of proposed approach lies in the synergistic integration of three components: (1) sinusoidal chaotic map initialization for both SSA population and control parameters, enhancing diversity and exploration to avoid local optima; (2) SMOTE for robust handling of imbalanced data, improving minority class prediction; and (3) a filter-based method selecting gene subsets with minimal error before chaotic SSA refinement, capturing comprehensive gene interactions. Unlike prior works [37], which used pseudo-random initialization and fixed top-100 gene selection), this hybrid framework addresses high-dimensionality and imbalance simultaneously, yielding superior accuracy and reduced gene counts.

Based on the preceding discussions, this paper aims to improve microarray gene selection by tackling the obstacles posed by high dimensionality and imbalanced data. It seeks to combine SSA with chi-square and mutual information filter methods, complemented by a chaotic map, to facilitate precise and effective gene subset identification. Furthermore, the study aims to introduce an approach for determining and selecting reduced gene subsets founded on minimal error. Through an emphasis on intricate gene interactions, the proposed methodology endeavors to heighten accuracy while minimizing gene subset dimensions. To achieve overarching objective, we specifically undertake the following actions:

1. Address the data imbalanced problem in microarray cancer gene datasets using the SMOTE technique.
2. Develop a filter-based method for determining and selecting reduced and top-performing gene subsets based on minimum error.
3. Design an enhanced initialization method for the SSA’s population and control parameters, integrating the sinusoidal chaotic map to improve the algorithm's convergence behavior and solution quality, for an improved gene selection task.
4. Assess the effectiveness of the proposed approach in comparison to existing methods in the literature using benchmark Ovarian, Leukemia, and Colon cancer datasets with popular metrics such as accuracy, gene counts, F1-score, standard deviation, and error rate.

The subsequent sections of this paper are organized as follows: Section 2 offers background information and an in-depth literature review. Section 3 discusses the proposed approach. Section 4 details the

experiments, comparisons, and analysis of results. Finally, Section 5 concludes with a summary of insights and prospects for future research.

2. Background

In this section, we present an overview of key concepts related to this research, including microarray data, dimensionality reduction, swarm intelligence algorithms, chaotic maps, imbalanced data, and machine learning algorithms. We also discuss state-of-the-art works in the literature relevant to this study.

2.1 Microarray Data

Microarray data refers to an HD dataset generated by microarray technology; a powerful tool used in molecular biology to measure the expression levels of thousands of genes simultaneously [38]. This technology allows researchers to examine the expression patterns of genes under different conditions, such as disease states or drug treatments. Analyzing microarray data can provide valuable insights into gene function, and disease mechanisms. As outlined by [39], there are four primary steps involved in acquiring microarray data, which include cell analysis, genetic material separation, identification of relevant genes, and compilation of a list containing the identified genes. A depiction of these procedures is illustrated in Figure 1, providing a visual representation of the microarray analysis process.

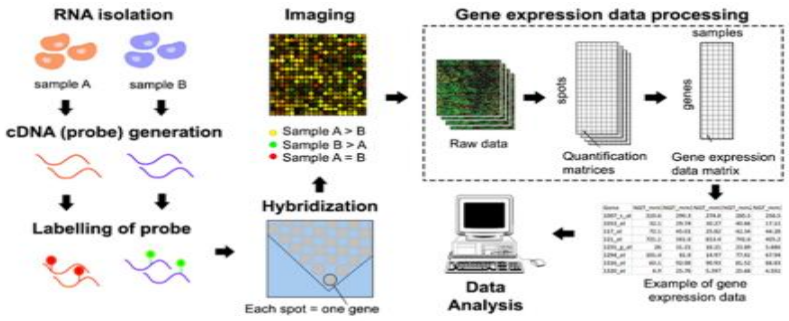

Figure 1: The microarray analysis process

2.2 Features selection (FS)

FS is a crucial step in ML and data analysis that involves identifying and selecting a subset of relevant features from the original dataset [40]. This process helps to reduce dimensionality, improve model performance, and enhance interpretability. There are three major types of FS techniques: filter methods, wrapper methods, and embedded methods. Figure 2 illustrates the core operations of FS. It commences with passing the complete feature set into the chosen FS technique, such as filter, wrapper, or embedded methods. Subsequently, a set of potentially optimal features is selected. Evaluation of the chosen feature subset is carried out using an ML algorithm to assess its effectiveness. This iterative process continues until a stopping criterion is satisfied, which may include reaching the maximum predefined number of iterations, achieving the desired performance level, reaching the maximum predefined runtime, maintaining a consistent accuracy level, or user intervention, among other factors.

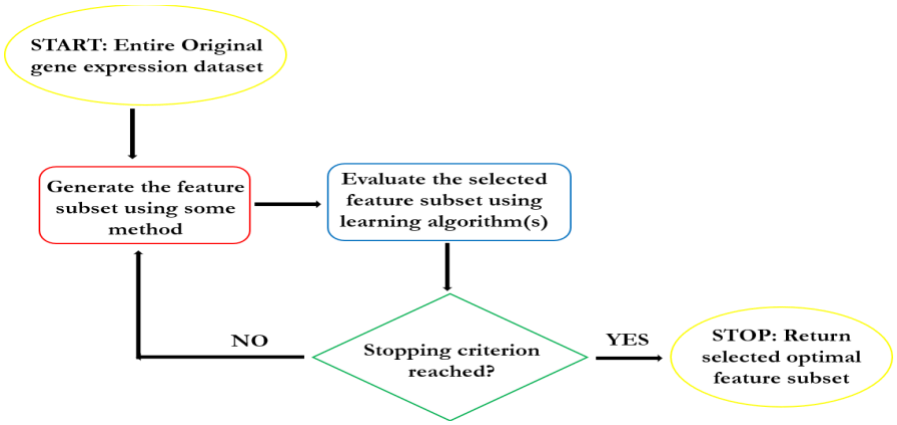

Figure 2: The overall process of feature selection

Filter methods evaluate the relevance of features independently of the learning algorithm. They typically rely on statistical measures, such as correlation, to rank features based on their predictive power [41]. These methods are computationally efficient and can handle HD datasets effectively. They are often used as a preprocessing step before applying more complex algorithms [42]. However, filter methods may overlook the interactions between features and their combined predictive power. Two of its most popular techniques include Mutual Information [43] and Chi-square [44].

XZXMutual information (MI) serves as a measure quantifying the correlation between two random variables, (representing the feature) and (indicating the associated class label), as elaborated by [15]. Originating from Shannon entropy principles, MI aims to gauge the level of uncertainty inherent in the distribution of events associated with feature. Lower entropy values denote reduced uncertainty when a particular event occurs more frequently. Conversely, when all events possess equal probabilities, entropy reaches its peak, indicating a lack of certainty regarding any specific outcome. Defined mathematically by [86] MI includes these concepts.

$$I(x;y) = \sum_{i=1}^{n}\sum_{j=1}^{n} p\big(x(i),y(j)\big).\log\left(\frac{p(x(i),y(j))}{p(x(i)).p(y(j))}\right) \quad (1)$$

$x\ yp\big(x(i),y(j)\big) = p\big(x(i)\big).p\big(y(j)\big)$. When MI equals zero, it implies that variables and are statistically independent, meaning the joint probability equals the product of their probabilities i.e., Eq. (1) and (2) illustrate the linear connection between MI and the entropies of the variables. Figure 3 employs a Venn diagram to visually represent the interplay among these variables [45].

$$I(x;y) = \begin{cases} H(x) - H(x|y) \\ H(y) - H(y|x) \\ H(x) + H(y) - H(x,y). \end{cases} \quad (2)$$

The conditions of each of the expressions holding are:

1. $I(x;y) = H(x) - H(x|y)$: Holds when MI between x and y equals the entropy of x minus the conditional entropy of x given y.
2. $I(x;y) = H(y) - H(y|x)$: Holds when MI between x and y equals the entropy of y minus the conditional entropy of y given x.
3. $I(x;y) = H(x) + H(y) - H(x,y)$: Holds when MI between x and y equals the sum of the entropies of x of y minus their joint entropy.

4. Suppose z represents a discrete random variable. To assess its interaction with the other variables, x and y, conditional MI can be utilized. This measure is articulated as follows:

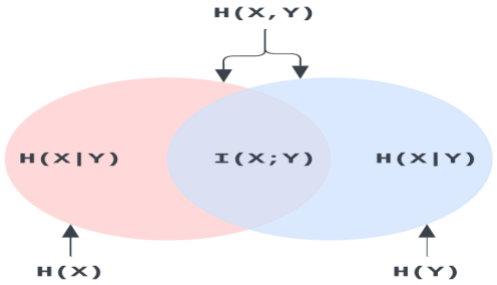

Figure 3: Venn diagram showing how the entropies of the variables and MI are related

$$I(x;y|z) = \sum_{i=1}^{n} p(z(i))I(x;y|z = z(i)) \quad (3)$$

$I(x;y|z = z(i))xyz = z(i)$.where is the MI between and when Conditional mutual information enables the quantification of information between two variables within the context of a third variable, but it does not allow for the measurement of information among all three variables simultaneously. In addition, the MI filter approach has several advantages that help explain why it works well for FS. Its capacity to capture nonlinear interactions, independence on label labels, and lack of assumptions about linearity or distribution are a few of these important strengths [46].

$f\ (X^2)$According to [7], the chi-square statistic measures the degree of independence or interdependence between two categorical variables. In particular, it evaluates the departure from the expected distribution based on the supposition that the feature is not affected by the class label. The chi-square test was given a mathematical expression in [47], in which the anticipated range is split up into intervals. Eq. (4) is used to determine the feature chi-square value.

$$X^2 f = \sum_{j=1}^{r}\sum_{s=1}^{c} \frac{(n_{js} - \mu_{js})^2}{\mu_{js}} \quad (4)$$

$rcn_{js}j^{th}s^{th}\mu_{js} = \frac{n_{*s}n_{j*}}{n}n_{j*}j^{th}n_{*s}\,s^{th}$In this context, denotes the count of dissimilar values within the feature, represents the number of dissimilar values within a class, signifies the frequency of the element within the class, and, where represents the frequency of the element, and indicates the total number of elements within the class. In essence, higher chi-square values indicate increased significance or importance. Because of its many advantages, the Chi-Square filter method is a well-regarded option for FS in a variety of data-driven fields, including microarray analysis. Easy interpretability, nonparametric nature, robustness to outliers, simplicity, and efficiency are some of its main advantages [4]

However, wrapper methods evaluate the performance of feature subsets by directly using a specific learning algorithm to train and evaluate models with different feature combinations. These methods search through the space of possible feature subsets using a predefined search strategy, such as forward selection, backward elimination, or exhaustive search. Wrapper methods can capture feature interactions and select subsets that are tailored to the learning algorithm. However, they are more computationally intensive and may be prone to overfitting, especially with large feature spaces [48]. Wrapper methods, particularly those employing SI techniques, have garnered significant attention in recent years for their efficacy in FS. These

methods harness the collective behavior of agents inspired by natural phenomena like the flocking of birds or the foraging of ants to optimize feature subsets. Among the most trending SI wrapper methods is the SSA. These methods iteratively explore the feature space, dynamically adjusting feature subsets to improve classification performance based on feedback from a specified evaluation criterion as exhibited in Figure 4. Its ability to efficiently navigate complex search spaces and identify relevant features makes it highly attractive for various applications, including bioinformatics, image processing, and financial analysis.

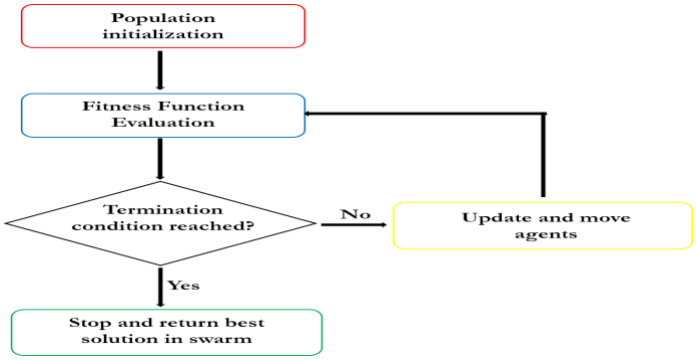

Figure 4: Operational principle of SI algorithms

Embedded methods incorporate FS into the model-building process itself. These methods typically use regularization techniques, such as Lasso regression or decision tree pruning, to simultaneously learn the model and select relevant features [49]. By integrating FS directly into the learning algorithm, embedded methods can automatically identify the most important features while building the model. This approach often leads to more robust and interpretable models. Embedded methods strike a balance between the efficiency of filter methods and the effectiveness of wrapper methods. They are particularly useful for datasets with a large number of features and limited computational resources [50].

2.3 Salp Swarm Algorithm (SSA)

Salps are cylindrical-bodied marine animals that are known to live in groups or chains in the ocean while purposefully drifting. Research has shown that Salp swarms can locate food more efficiently and accomplish greater mobility [51]. The idea of SSA was first presented by and has been applied to several optimization issues. To find the target or food supply and navigate the search space, all salps in SSA assemble into a structure resembling a chain. The swarm is split into two components to quantitatively depict the chain generation of salps: the leader and the followers. Always taking the lead, the leader Salp directs the other members in the chain.

Consider a collection of n Salps denoted as $Y = \{Y_1, Y_2, \ldots, Y_i, \ldots . Y_n\}$. Each Salp is presented as a d-dimensional vector $(Y_i = y_1, y_2, \ldots, y_d)$. F_s represents the target vector or food source. Salp's leadership status is updated following Eq. (5).

$$Y_i \begin{cases} F_s + \alpha 1\left((Y_{max} - Y_{min})\alpha 2 + Y_{\min}\right)\alpha 3 \geq 0.5 \\ F_s - \alpha 1\left((Y_{max} - Y_{min})\alpha 2 + Y_{min}\right)\alpha 3 < 0.5 \end{cases} \quad (5)$$

With Y_{i} denoting the leading Salp's location, we have random values $\alpha 1, \alpha 2$, and $\alpha 3$ in this configuration. The upper and lower bounds for each Salp are indicated by the variables Y_{max}, and Y_{min} respectively. In SSA, $\alpha 1$ controls the ratio of exploration to exploitation, and Eq. (6) is used to change its value at each cycle.

$$\alpha 1 = 2e^{-\left(\frac{4*c_{iter}}{Max_{iter}}\right)^2} \quad (6)$$

The variables Max_{iter} and c_{iter} in this context denote the total number of iterations and the current iteration, respectively. Using Newton's law of motion, the positions of the follower Salps aside from Y_1 are improved, as outlined in Eq. (7).

$$Y_j(i) = \frac{1}{2}at^2 + v_o t \quad (7)$$

In this case, j can be any value between 2 and n, and $Y_j(i)$ denotes the i^{th} dimension of the j^{th} Salp. The initial velocity, time, and acceleration are represented by v_o, t, and a respectively, and are computed using Eq. (8).

$$a = \frac{v_{end}}{v_o} \; where \; v = \frac{y - y_o}{t} \quad (8)$$

The initial velocity is set to 0 and the term "time" denotes the number of iterations in the context of optimization issues. As a result, the followers' positions are updated using a modified equation given in Eq. (9).

$$Y_j(i) = \frac{1}{2}\,(Y_j(i) + Y_{J-1}(i) \quad (9)$$

According to Eq. (9), i must be assigned a value of $i \geq 2$, indicating it represents a follower; conversely, a value of 1 signifies it is a leader. The Salp that best fits the situation is chosen as the food supply, where the initial population is created at random. After that, the remaining salps move near this food source. With every iteration, the food source (F_s) position is updated.

2.4 Chaotic maps

Chaotic maps have gained attention in the field of SI as an effective method for initializing the search process [14]. These maps introduce an element of randomness and complexity into the initial positions of swarm individuals, such as particles in PSO or agents in Artificial Bee Colony (ABC) algorithms [23]. This initialization strategy aims to enhance the exploration capability of SI algorithms by introducing unpredictable, yet controlled, behaviors at the start of the optimization process [14]. Chaotic maps exhibit sensitive dependence on initial conditions, which means that small variations in the initial positions can lead to significantly different trajectories. This property is harnessed to ensure diverse exploration of the search space, helping SI algorithms avoid local optima and discover more promising solutions [11]. Therefore, chaotic map-based initialization methods have become valuable tools for improving the performance and robustness of SI algorithms in various optimization tasks. Examples of chaotic maps include the Logistic map, sine map, tent map, iterative map, sinusoidal map, and singer map [52].

The sinusoidal chaotic map is a mathematical function used in chaotic systems and dynamic systems theory. It is characterized by its sinusoidal nature, which introduces nonlinearity and unpredictability into the system. The map iterates a given initial value through a series of calculations involving sine functions, resulting in a sequence of values that exhibit chaotic behavior. This chaotic map has been employed in various fields, including cryptography and optimization algorithms. Its unique properties make it suitable for applications where randomness and complexity are desired, such as in the initialization of optimization algorithms like the SSA for solving optimization problems efficiently. The sinusoidal map is defined as in Eq. (10)

$$x_{k+1} = P.x_k^2 \sin(\pi x_k) \quad (10)$$

where x_k is the current value, x_{k+1} represents the next value, P stands for the controlling parameter for the chaotic map. The $\sin(\pi x_k)$ represents the sine function

2.4 Imbalanced data

Imbalanced data refers to datasets where the distribution of classes or categories is heavily skewed, with one or more classes significantly outnumbering others [53]. This imbalance poses challenges for ML algorithms, as they tend to be biased towards the majority class, leading to suboptimal performance in predicting the minority class. Imbalanced data can be categorized into two basic types: algorithm-driven approaches and data-driven approaches. Both approaches have their advantages and limitations, and the choice between them depends on factors such as the specific characteristics of the dataset, computational resources available, and the desired balance between predictive performance and interpretability [54]. Ultimately, addressing imbalanced data is crucial for building robust and reliable ML models in various domains, including healthcare, finance, and fraud detection. The resampling (random over and under sampling) and SMOTE techniques are the two most common data imbalance handling techniques.

2.4.1 Algorithm-driven approach

Algorithm-driven approaches involve modifying the learning algorithm to handle class imbalance more effectively [3]. Techniques such as cost-sensitive learning, where misclassification costs are adjusted to penalize errors on the minority class more heavily, fall under this category [3]. Another approach is to use ensemble methods like bagging and boosting, which combine multiple models to improve predictive performance on imbalanced datasets.

2.4.2 Data-driven approach

On the other hand, data-driven approaches focus on modifying the dataset itself to rebalance the class distribution [83]. One common method is resampling, which involves either oversampling the minority class to increase its representation or under-sampling the majority class to reduce its dominance [55]. Oversampling techniques include Random Oversampling and Synthetic Minority Over-sampling Technique (SMOTE), while under sampling methods include Random under sampling and Near Miss [56].

2.4.3 Oversampling

This technique involves increasing the number of instances in the minority class to match the majority class [80]. While ROS is simple and easy to implement, it can lead to overfitting and does not introduce new information.

2.4.4 Under sampling

Conversely, under sampling reduces the number of instances in the majority class to balance class distributions [80]. Random Under sampling (RUS) randomly removes instances from the majority class until the class balance is achieved. While under sampling can help reduce computational complexity and processing time, it may lead to loss of valuable information present in the majority class, resulting in underfitting and reduced model performance [13].

2.4.5 SMOTE (Synthetic Minority Over-sampling Technique)

SMOTE is a popular oversampling technique designed to address the limitations of simple oversampling methods like ROS [13]. Instead of replicating minority class instances, SMOTE generates synthetic instances by interpolating between existing minority class samples [13]. It selects a random minority class instance and then selects its k nearest neighbors. A new instance is then generated by selecting a random

point along the line connecting the chosen instance and one of its neighbors [57]. This process is repeated until the desired balance between the minority and majority classes is achieved.

2.5 Related works

This section provides an overview of recent and pertinent literature relevant to the present study. Firstly, it examines discussions on research endeavors that combine the filtering strengths of filter methods with the global search capabilities of SI algorithms to tackle microarray gene selection challenges. This discourse extends beyond the Salp Swarm Algorithm to include other prominent SI algorithms documented in the literature, such as Particle Swarm Optimization (PSO), Genetic Algorithms (GA), and Month Flame Optimization (MFO), among others. Secondly, it delves into literature exploring enhancements in SI algorithms' initialization through the integration of chaotic maps. Lastly, the section addresses recent studies focused on addressing imbalanced data challenges inherent in microarray cancer datasets.

2.5.1 Filter-based SI methods for microarray gene selection

In the field of bioinformatics, the utilization of filter-based FS methods for microarray gene selection has emerged as a pivotal area of research. Microarray technology has revolutionized our ability to analyze gene expression data on a large scale, offering invaluable insights into complex biological processes. Within this context, filter-based FS methods play a crucial role in identifying informative genes amidst vast datasets, facilitating the extraction of meaningful biological knowledge. This literature review aims to explore the landscape of filter-based FS techniques tailored specifically for microarray gene selection, delving into their methodologies and applications, in advancing our understanding of biological systems. [20] introduced a gene selection approach for multiple HD microarray cancer datasets. They combined the Relief, Correlation, ANOVA, Information Gain (IG), and IG gain ratio filter methods in a technique termed TOPSIS to select top genes. These selected genes were then subjected to the Jaya Algorithm (JA) for final refinement. For classification, the NB machine learning algorithm was employed. The TOPSIS technique was found to be 10 times faster than the compared methods. However, the model may suffer from issues related to scalability compounded by the time required to evaluate each alternative solution (combination of features). This is due to the increasing computational complexity, larger search space, potential convergence challenges, higher resource consumption, and longer evaluation times as the problem size grows. These factors can limit the practical applicability of the method to large-scale feature selection problems, especially in microarray datasets. [21] introduced a gene selection technique that merges Conditional Mutual Information (CMI) with Moth Flame Optimization (MFO). They utilized multiple HD microarray cancer gene datasets as benchmark datasets and employed SVM as the classifier. The integration of CMI with MFO offers a unique approach to gene selection with benefits such as enhanced feature relevance and improved convergence. However, the traditional MFO for feature selection suffers from slow convergence, premature convergence, and limited exploration in high-dimensional spaces, often requiring careful parameter tuning and facing scalability challenges with large datasets. Additionally, no effort was taken to rectify the severe imbalance present in most of the datasets employed in the study. This has significantly affected their model's efficacy and overall generalizability.

Authors in [7] conducted a study employing three filter methods i.e. Chi-Square, Information gain, and Relief F for initial gene selection across six microarray cancer datasets. Each of the filter methods was employed to independently select and rank the top-performing genes based on their scores. Next, using the mean of the scores, the gene scores from each filter are combined into a single gene ranking list. Following this preliminary selection, the final subset of selected genes underwent further refinement through PSO.

For classification tasks, the study utilized three ML algorithms: Decision Tree (DT), K-Nearest Neighbor (KNN), Support Vector Machine (SVM), and Naïve Bayes (NB). Nevertheless, no effort was taken to rectify the severe imbalance present in four of the six datasets included in the study. This has significantly affected their model's efficacy and generalizability. [42], employed SSA in combination with the Chi-Square and Mutual Information filter methods for gene selection in microarray cancer datasets of Ovarian and Colon cancers. Referred to as the combination approach, the authors strategically combined the top genes selected from each filter method, selecting only the top 100 best-performing genes, which were then passed to the SSA for further intelligent selection. The learning algorithm utilized was KNN. Experimental results revealed that when applied to the Ovarian cancer dataset, their method achieved an accuracy and F1 score of 98.00% and 97.00% respectively. Similarly, for the Colon cancer dataset, the method achieved accuracy and F1 scores of 94.20% and 93.60% respectively, all while significantly reducing the number of genes.

2.5.2 SI initialization enhancement using chaotic maps

In the literature, several chaotic maps like sine, tent, logistic, and sinusoidal, among others, have been utilized to initialize both the population and various control parameters of MAs. This approach aims to enhance the convergence of solutions towards optimality. For example, [11] employed the Grasshopper Optimization Algorithm (GOA) in conjunction with ten chaotic maps. Each chaotic map was utilized to initialize the $c1$ and $c2$ control parameters of the GOA. However, their models were evaluated on 10 benchmark test functions rather than on a FS task. Authors in [58] introduced a binary chaotic GA for FS, utilizing ten different chaotic maps to initialize the population and mutation operators of the GA. Their method was evaluated using two large healthcare datasets with extensive feature spaces, demonstrating promising performance in identifying optimal feature subsets with improved fitness values. However, the algorithm's effectiveness in the exploration and exploitation phases, particularly in dynamic search spaces where chaotic behavior might offer additional benefits, may be limited by the study's restriction of the use of chaotic maps to only the mutation and population initialization stages. This could potentially limit the algorithm's ability to fully leverage chaotic dynamics throughout the optimization process.

In their study, [27] proposed an enhanced version of the Reptile Search Algorithm (RSA) by integrating it with the Simulated Annealing (SA) local search technique to improve its exploitation capabilities. Additionally, they employed the circle map chaotic map for population initialization to enhance RSA's solution diversity. Their approach was evaluated using more than 20 medical datasets sourced from the University of California in Irvine (UCI) repository. The findings demonstrated the superiority of IRSA over the original RSA algorithm and other optimized algorithms across the majority of the medical datasets. Nonetheless, the study restricted the use of the circle map chaotic map to only the population initialization stage. By doing this, the algorithm may not be able to fully leverage chaotic dynamics throughout the optimization process. This could limit the algorithm's effectiveness in the exploration and exploitation phases, particularly in dynamic search spaces where chaotic behavior might offer additional benefits. Recently, [43] introduced an enhanced version of the Sparrow Search Algorithm (SSA) by integrating 10 chaotic maps to enhance the algorithm's overall efficacy. Their improvements targeted three key aspects of SSA. Firstly, they optimized population initialization to enhance diversity and convergence among search agents. Secondly, chaotic maps were utilized to initialize two critical random numbers governing SSA's convergence. Lastly, these chaotic maps were employed to confine sparrows within the search range. Performance evaluation conducted on benchmark datasets from the UCI repository, three microarray cancer datasets, and standard benchmark test functions showcased the method's effectiveness compared to existing

literature studies. However, the study does not fully discuss how each chaotic map contributes to the overall improvement in SSA's performance, which makes it difficult to determine the precise effect of each chaotic map on the optimization process and may further hinder opportunities to optimize the algorithm's parameters for improved performance.

2.5.3 Imbalance data handling for microarray gene selection

In this section, we delve into strategies to address the imbalanced nature of microarray datasets, crucial for effective gene selection. Techniques such as RUS, ROS, and SMOTE are explored to mitigate class imbalances, enhancing the performance of gene selection algorithms. For example, in [59] introduced a Wasserstein Generative Adversarial Network (WGAN) approach to mitigate imbalanced learning challenges across three HD cancer datasets and facilitate subsequent classification. The WGAN facilitates the generation of new samples from the minority class, effectively addressing the imbalance issue at the data level. The results indicated that achieving a balanced data distribution and expanding the sample size significantly improved prediction accuracy across all three datasets compared to conventional RUS and ROS techniques. Nonetheless, because deep learning-based methods like WGAN rely heavily on computer resources, researchers with limited computational capabilities may find it more difficult to use. Furthermore, even though WGANs show promise in addressing data imbalance and investigating data features, problems with model interpretability and generalization may still arise, especially in complicated and diverse cancer gene expression datasets.

Authors in [13] applied various supervised ML algorithms, including SVM, DT, KNN, NB, and MLP, for the classification of a multiclass lung cancer dataset. The dataset comprised four classes with instance percentages of 0.68, 0.08, 0.02, 0.10, and 0.09 for each class. To mitigate imbalanced class distribution, the authors employed ROS, RUS, and SMOTE. Comparative analysis revealed significant enhancement in the predictive capabilities of the supervised learning algorithms following dataset resampling, with the SVM-SMOTE model demonstrating superior performance. The analysis, however, is restricted to a single, very specific lung cancer dataset, which may limit the findings' generalizability to other datasets or domains and raise doubts about the suggested method's resilience and suitability in a range of data types and clinical settings. Authors in [60] employed the SMOTE to address the imbalanced nature inherent in seven HD microarray cancer datasets. Subsequently, they applied Principal Component Analysis (PCA) as a DR technique to reduce the dimensions of these datasets before subjecting them to six supervised ML algorithms, each evaluated using 5-fold cross-validation, for the classification of cancer types into distinct classes. The study did not, however, assess how well the SMOTE technique performed in comparison to other popular resampling techniques for unbalanced data. Furthermore, a detailed analysis of the effects of various parameter settings or modifications to the SMOTE approach, like altering the value of K, is lacking.

3. Proposed gene selection method

This section proceeds by elaborating on the specific design and implementation aspects of the proposed gene selection method. It encompasses discussions on various components, including the utilization of chaotic methods for the SSA population and the α_2 and c_3 initialization. Additionally, it outlines the Transfer Function (TF) employed for converting continuous values to binary, the fitness function utilized for evaluating the selected gene subsets, and the evaluation metrics employed. Furthermore, it provides insights into the microarray cancer datasets utilized, and the chosen learning algorithm, and presents a detailed description of the proposed method's architecture, encompassing materials, parameters, and settings. Additionally, it outlines the design procedure for conducting experiments and presenting results.

3.1 Chaotic method of SSA initialization

In this study, we have adopted the Sinusoidal chaotic map for initializing both the SSA population and the $\alpha 2$ and $\alpha 3$ control parameters. The Sinusoidal chaotic map has demonstrated superior effectiveness in microarray gene selection by generating sequences of numbers characterized by intricate and unpredictable behaviors, driven by sensitivity to initial conditions [61]. This contrast with the use of pseudo-random numbers allows for a more comprehensive exploration of the solution space, facilitating the discovery of potential solutions and mitigating the risk of being trapped in local optima. Moreover, employing the Sinusoidal chaotic map for initializing the parameters of SSA achieves a balanced approach between exploration and exploitation, dynamically adapting to the difficulties of the optimization problem, which surpasses the capabilities of traditional pseudo-random initialization methods

3.2 Transfer function

This study utilizes the sigmoid TF to convert continuous values into binary representations, as introduced by [45]. The sigmoid TF is a member of the S-shape TF family and is selected for its straightforward interpretation in terms of probability or likelihood, along with its robustness against noise and outliers. It serves to differentiate between selected and non-selected genes effectively. Mathematically, the sigmoid TF is expressed as follows:

$$S\left(x_i^d\right) = \frac{1}{1+e^{-x_i^d}} \tag{11}$$

$$x_i^d = \begin{cases} 1, if\ S\left(x_i^d\right) > \alpha, \\ 0, \text{ and} otherwise \end{cases} \tag{12}$$

where $\alpha \sim U(0,1)$, $x \epsilon \mathbb{R}$ denotes the possible solution and d stands for the gene's continuous value at a time.

3.3 Fitness function

The fitness function is a mathematical technique used for evaluating the fitness of the SSA's solution. It provides candidate solutions with a fitness score based on how effective it is in satisfying the objectives of the optimization task. Equation 13 presents the fitness function employed in this work [39]:

$$fitness\ function = \alpha \Delta_R(D) + \beta \frac{|Y|}{|T|} \tag{13}$$

where $\Delta_R(D)$ is the error rate of the classifier. $|Y|$ is the size of the chosen feature subset and $|T|$ is the total number of features in the dataset. The $'\alpha' \in [0,1]$ is the classifier's error rate's weight. $'\beta' = (1-\alpha)$ is the significance of the features reduction.

3.4 Datasets description

Table 1 displays the three microarray cancer datasets utilized in this study, providing a comprehensive breakdown of their dimensions [38]. These datasets are publicly available and play a vital role in providing a standardized framework for evaluating algorithms, fostering reproducibility, ensuring real-world relevance, enabling performance comparison, and promoting collaboration among researchers and practitioners. These datasets are designed for binary classification, meaning instances are categorized as either cancerous or non-cancerous.

Table 1: Description of the three microarray cancer datasets used

Datasets	No. Instances	No. Features	No. Class
Colon Cancer	62	2000	2 (40,22)
Leukemia Cancer	72	3572	2 (25,27)
Ovarian Cancer	253	15155	(162,91)

3.5 Imbalanced data handling

Examining the microarray cancer datasets utilized in this research, as depicted in Table 1, it is evident that both the Ovarian and Colon cancer datasets exhibit an imbalance, with a class ratio of nearly 2:1 for cancerous and non-cancerous instances. These result in several problems such as biased model performance toward the majority class, poor generalization to unseen data, misleading evaluation metrics such as accuracy, increased false negatives, and model skewness towards the majority class [13,59]. Thus, in this study, the SMOTE technique of imbalanced data handling is employed to address these issues.

The basic idea behind SMOTE involves generating synthetic samples by interpolating between existing minority class samples. Given a minority class sample x_i, SMOTE selects one of its k-neighbors x_{zi} and generates a synthetic sample x_{new} by combining x_i and x_{zi} according to the formula:

$$x_{new} = x_i + \lambda * (x_{zi} - x_i) \quad (14)$$

where λ is a random value between 0 and 1. This process is repeated for each minority class sample to create synthetic samples, thereby balancing the class distribution.

4. Proposed gene selection method

Figure 5 presents the overall system architecture of the proposed gene selection method. The architecture is divided into four distinct stages aimed at selecting an optimal gene subset. These stages include:

Stage1: Preliminary data pre-processing

Stage2: Data imbalance handling using SMOTE

Stage3: Filter-based method to determine and select gene subsets with minimal error

Stage4: Chaotic SSA Initialization

4.1 Preliminary data pre-processing

In this stage, we conduct target variable encoding, data normalization, and data cleaning on the microarray cancer datasets utilized. These steps aim to enhance the effectiveness of the learning algorithm i.e., the KNN which relies on distance metrics, mitigates bias towards specific genes, and facilitates easier data analysis.

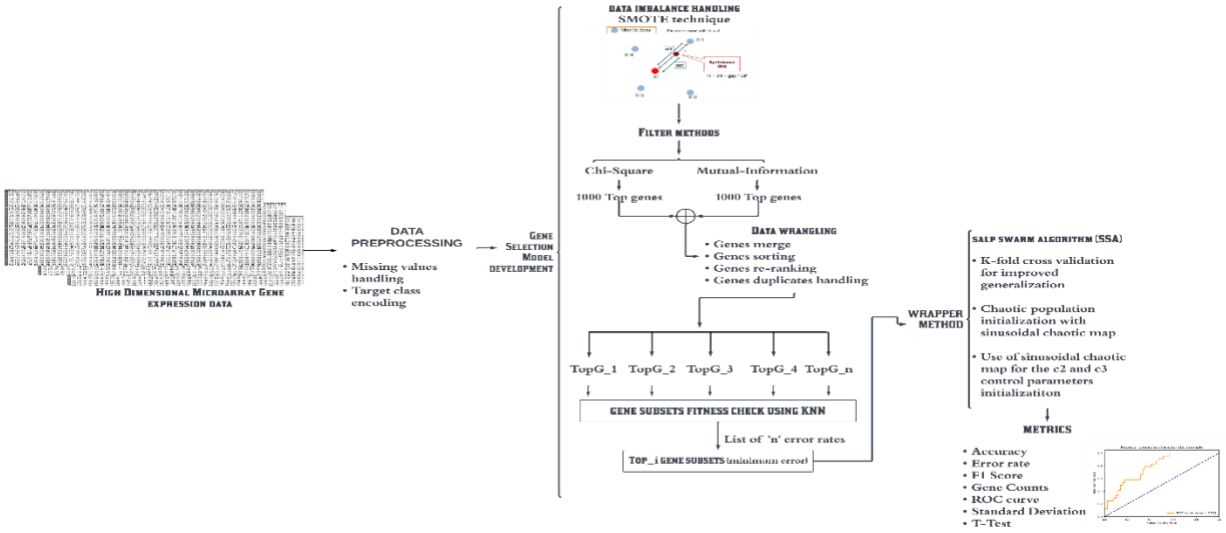

Figure 5: The proposed system architecture

4.2 Data imbalance handling using SMOTE

After the employed datasets are preprocessed from the first stage, the SMOTE technique is applied to address the imbalanced nature of the data. This technique generates synthetic samples for the minority class to balance the class distribution and improve the performance of the learning algorithms. Note that only the Ovarian and Colon cancer datasets are considered as imbalanced in this study.

4.3 Filter-based method to determine and select gene subsets with minimal error

In the study by [42], both the Chi-Square and MI filter methods were individually applied to microarray cancer datasets to select the top 100 best-performing genes. Subsequently, the selected genes were merged, and duplicates with lower scores were eliminated, resulting in a final selection of the top 100 genes for further selection using SSA. Although this approach yielded a reduced number of selected genes with reasonably accurate classification, it may have overlooked crucial disease-relevant information, potentially leading to suboptimal results.

In contrast, the proposed method expands the selection of genes from each filter method to 1000 based on previous studies demonstrating good classification performance with subsets of even fewer than 1000 genes [9,38]. Hence, we contend that it is likely that the few best-performing genes to be selected in subsequent stages should be included therein. After merging the 1000 genes from each filter method to create a subset of 2000 genes, duplicates were resolved by retaining the gene with the higher score. Additionally, our method introduces a hyperparameter 'n' to determine multiple gene subsets of various predetermined sizes from the reduced gene subset. This is denoted as 'TopG_n' in Figure 3. For instance, n = 5, which means five gene subsets with different predetermined sizes. Subsequently, error rates for each gene subset are computed using the proposed learning algorithm, and the subset with the minimal error rate is automatically chosen for further selection by the chaotic SSA algorithm. This approach addresses the limitations of [42] by ensuring a more comprehensive selection of genes, considering potential dependencies and interactions among genes, and ultimately aiming for improved performance

4.4 Chaotic SSA initialization

In this study, the sinusoidal chaotic map will serve a dual purpose in initializing the population of Salps and determining the values of the $\alpha 2\ and\ \alpha 3$ control parameters. The sinusoidal map's unique properties, characterized by its sensitivity to initial conditions and unpredictability, make it an ideal choice for introducing diversity and complexity into the initial population of Salps. Moreover, its application in initializing control parameters ensures adaptability and exploration in the optimization process. By leveraging the sinusoidal map for both population initialization and control parameter assignment, this study aims to enhance the exploration capability and convergence speed of the SSA, ultimately improving its performance in solving HD microarray gene selection problems. Algorithm 1 shows the sinusoidal chaotic map-based SSA initialization.

Algorithm 1: Sinusoidal-Chaotic Salp Swarm Optimization

Input: population of salps N ; Max_iter, sinusoidal map parameters i.e initial value (x_0) and control parameter P
$F(x)$: objective function
Output: F_S: best solution;
Generate the initial population (Y_i), where $i = 1,2,\ldots,N$ using Sinusoidal-chaotic initialization Eq. (2.11)
t = 0;
While ($t < Max\_iter$) do
    Compute Salp's fitness and find the current best
    Calculate α_1 using Eq. (2.7)
    For each Salp (Y_i) In the chain do
        if ($j == 1$) then
            Modify the position of leader Salp using Eq. (2.6)
        Else
            Modify locations of followers Salps according to Eq. (2.10)
        end if
    end for
    $t = t + 1$
end while

4.5 Evaluation metrics

To evaluate the efficacy of the proposed gene selection method in comparison with other methods, the following metrics will be used accordingly.

1. Classification accuracy: measures the overall correctness of the model's predictions by calculating the proportion of correctly classified instances out of the total number of instances in the dataset.
2. Error rate**:** represents the proportion of incorrectly classified instances with the total number of instances in the dataset.
3. F1-Score: The F1 score is often considered one of the best metrics for evaluating performance in imbalanced datasets because it considers both precision and recall, enabling a balanced assessment of its predictive capabilities
4. Selected Gene Count: returns the number of selected genes by fittest candidate solution.
5. Standard Deviation (SD)**:** quantifies the amount of variation or dispersion in a dataset.

4.6 Parameters values and settings

Table 2 outlines the parameter values and configurations employed in our proposed gene selection methodology. Parameters related to SSA, including population size and the number of iterations, were established in line with the settings used by [42] to ensure a fair comparison. Note that the SSA does not have an additional parameter that needs initialization aside from the α1, α2, and α3. The parameter values for the sinusoidal map, such as the initial start value and control parameter, were selected based on the approaches outlined by [11]. Additionally, other settings, including the search boundary, value in KNN, value of cross-validation, and alpha-beta values in the fitness function, were determined based on widely recognized values from various literature sources over the years.

Table 2: Parameters values and settings for the proposed method

	Hyperparameters	**Parameter values**
	k-fold cross validation	10
	$'\alpha'$	0.99
	$'\beta'$	0.01
	Dimension (D)	Gene count in datasets
	K value in KNN	5
	Search boundary	0 -1
	Number of runs (P)	10
Sinusoidal	x_0	0.7
	P	2.3
SSA	Population Size	20
	Number of generation/iterations	20
	α2, and α3	Chaotic initialization

4.7 Experimental design and result presentation

The experiments and result presentation in this study will be conducted in two distinct forms.

i. Firstly, concerning the hyperparameter $'n'$ which will subsequently be compared to determine the superior performing approach. i.e.
   a) $'n'$ will be initialized to 5, with gene counts 100, 200, 300, 400, and 500 respectively.
   b) $'n'$ will be initialized to 10, with gene counts 10, 20, 30, 40, 50, 60, 70, 80, 90, and 100

Secondly, the method from (i) demonstrating the best performance will then undergo further evaluation against existing studies in the literature to assess its efficacy and relevance in the broader context of the research.

5 Results analysis and discussion

All experiments in this study are carried out using the Python programming language within a Jupyter Notebook integrated development environment. The experiments are executed on a computer equipped with an Intel(R) Core (TM) i7-6600U CPU running at 2.80 GHz and 8.00 GB of RAM. This section presents the results of the experiments conducted following the experimental design outlined in prior section. We

explored two values of the $'n'$ parameter, namely 5 and 10, each representing subsets with varying numbers of genes. It is important to note that the selection of these $'n'$ values was intuitive and not based on existing literature, suggesting potential avenues for future research exploring different $'n'$ values and corresponding gene counts. Subsequently, the $'n'$ value demonstrating superior performance based on the utilized metrics was chosen for comparison with existing literature.

5.1 Comparing the performance of the proposed method with two values of 'n'

In the discussion of the results obtained from experiments conducted with different values of $'n'$ across three benchmark datasets (Ovarian, Colon, and Leukemia), Table 3 demonstrates notable variations in performance metrics and selected gene counts. Note that results with better performance are presented in bold. Starting with the Ovarian dataset, it is evident that setting $'n'$ to 10 led to superior results across all metrics compared to $'n'$ set to 5. Despite achieving perfect scores in accuracy, F1 score, and error rates for both 'n' values, the model selected significantly fewer genes (46) when $'n'$ was set to 10, in contrast to 95 genes with 'n' set to 5. This suggests that while subsets with smaller gene counts failed to achieve improved error rates, the model's decision with $'n'$ = 10 reflects a balance between minimal gene counts and enhanced gene classification accuracy.

Similarly, for the Colon cancer dataset, the model achieved perfect accuracy, F1 score, and error rates with both $'n'$ values, but notably selected fewer genes (4) when $'n'$ was set to 10 compared to 48 genes with $'n'$ set to 5. Likewise, in the case of Leukemia cancer, perfect performance metrics were achieved with both $'n'$ values, but the model selected fewer genes (40) with $'n'$ set to 10 compared to 47 genes with $'n'$ set to 5.

The interesting aspect of the experiment lies in the selection of fewer genes with $'n'$ = 10 across all datasets. This phenomenon suggests that the model likely favored gene subsets containing at least 50 genes, indicating that subsets comprising 10, 20, 30, and 40 genes failed to achieve improved error rates. Therefore, the decision to select fewer genes with $'n'$ = 10 reflects a strategic balance between gene count and classification accuracy. Overall, the performance improvement observed with $'n'$ = 10 can be attributed to the model's ability to strike a balance between exploring diverse gene subsets and selecting those that contribute most significantly to improved classification accuracy, thereby mitigating the risk of overfitting and enhancing generalization capability. Due to the model's superior performance when $'n'$ is set to 10 across all three datasets, it has been chosen for subsequent comparison with existing studies outlined in earlier sections.

Table 3: Performance comparison of the proposed method with different values of 'n'

Dataset	Algorithms	No. Genes	Accuracy and SD	F1-Score	Error Rate
Ovarian Cancer	$n = 5$ (100,200,300,400,500)	95 ± 3.31	100.00 ± 0.00	100.00 ± 0.00	0.00
	n = 10 (10,20,30,40,50,60,70,80,90,100)	**46 ± 3.69**	**100.00 ± 0.00**	**100.00 ± 0.00**	**0.00**
Colon Cancer	$n = 5$ (100,200,300,400,500)	48 ± 3.69	100.00 ± 0.00	100.00 ± 0.00	0.00
	n = 10 (10,20,30,40,50,60,70,80,90,100)	**4 ± 1.49**	**100.00 ± 0.00**	**100.00 ± 0.00**	**0.00**
Leukemia Cancer	$n = 5$ (100,200,300,400,500)	47 ± 3.48	100.00 ± 0.00	100.00 ± 0.00	0.00
	n = 10 (10,20,30,40,50,60,70,80,90,100)	**40 ± 4.45**	**100.00 ± 0.00**	**100.00 ± 0.00**	**0.00**

5.2 Comparing the performance of the proposed method with existing methods in literature

As outlined in previous sections, the proposed method with $'n'$ set to 10, will be employed for comparative analysis with five recent studies in the literature, owing to its superior performance across all metrics compared to its counterpart. These existing studies have been extensively discussed in the literature. Note that as can be observe from Table 4, there are instances of missing results/values, particularly in gene counts for some existing methods. This might hinder the accurate interpretation of subsequent graphs. As a result, the studies with missing results are not included in the graphs presented.

The comparison basis between our proposed method and the five existing studies lies in the utilization of SI algorithms integrated with one or more filter methods for microarray cancer gene selection. The comparison metrics encompass accuracy, F1 score, and gene counts. This selection is due to the absence of error rate as a metric in most of the existing studies. Additionally, not all the studies utilized all three datasets as in our study. Consequently, for each of the existing studies, we only utilize datasets that overlap with ours for comparison purposes. Table 4 presents the comparison results for each of the three benchmark datasets. Similar to Table 3, each row corresponds to a specific dataset utilized.

Upon comparing the performance of the proposed method with existing studies, particularly concerning the Ovarian dataset, noteworthy insights emerge. The proposed method, alongside the work of [7], exhibited superior performances, achieving perfect scores of 100% in both accuracy and F1 metrics. However, it is worth noting that while [7] selected a significantly lower number of genes (13), our proposed method selected more genes, totaling 46. In the broader context of competing methods, specifically in the gene count metric, our proposed approach performs rather poorly by selecting the highest number of genes (46). Nonetheless, considering the vast pool of genes in the dataset (15,155), this selection can be deemed fair. Moreover, the ability of our method to achieve superior classification accuracy and F1 score underscores its capability to strike a balance between accuracy and the number of selected genes. The utilization of a chaotic method for both population and control parameter initialization, coupled with the integration of the SMOTE technique to address the imbalanced nature of the data, significantly contributed to this superior performance. The chaotic initialization facilitated a diverse exploration of the solution space, enhancing the

algorithm's ability to converge to optimal solutions. Concurrently, the SMOTE technique ensured that the KNN learning algorithm was trained on a more balanced dataset, mitigating bias and enhancing generalization capability.

Table 4: Performance comparison of the proposed method with existing studies in literature

Datasets	Citations/Methods	Accuracy Score	F1-Score	No. Genes
	Proposed Method	**100.00**	**100.00**	46
Ovarian Cancer	(Jeremiah et al., [40])	98.00	97.00	18
	(Isuwa et al., [38])	93.55	-	41
	(Ke et al., [46)	98.42	-	-
	(Alrefai and Ibrahim, [7]	**100.00**	**100.00**	**13**
	(Chaudhuri and Sahu, [20])	99.52	-	18.50
	Proposed Method	**100.00**	**100.00**	**4**
Colon Cancer	(Jeremiah et al., [40])	94.20	93.60	16
	(Isuwa et al., [38])	97.37	-	46
	(Ke et al.,[46])	91.90	-	-
	(Alrefai and Ibrahim, [7])	92.86	93.00	41
	(Chaudhuri and Sahu, 2021)	97.76	-	18.90
	Proposed Method	**100.00**	100.00	40
Leukemia Cancer	(Jeremiah et al., [40])	-	-	-
	(Isuwa et al., [38])	**100.00**		44
	(Ke et al., [46])	-	-	-
	(Alrefai and Ibrahim, [7])	**100.00**	**-**	**26**
	(Chaudhuri and Sahu, [20])	**-**	**-**	**-**

In essence, the combined influence of chaotic initialization and imbalanced data handling mechanisms played a pivotal role in the proposed method's superior performance. By effectively balancing gene selection with classification accuracy and leveraging innovative techniques for data handling and initialization, our approach demonstrates robustness and efficacy in addressing the challenges posed by high-dimensional datasets in the Ovarian cancer dataset.

When evaluating the Colon cancer dataset, our proposed method demonstrated superior performance across all metrics compared to its competitors. Notably, it achieved perfect scores of 100% in both accuracy and F1 score, surpassing the performance of other methods. What is particularly notable is that our method achieved these exceptional results while utilizing significantly fewer genes, only 4 in total. There are several reasons why our method outperformed its competitors in this context. Firstly, the thorough exploration of the gene space facilitated by our approach allows for the identification of the most discriminative features relevant to Colon cancer classification. By efficiently selecting a concise subset of genes, our method minimizes redundancy and focuses on the most informative genetic markers associated with the disease. Furthermore, the utilization of a sophisticated method for gene subset selection and refinement, as well as the strategic integration of filter methods, enhances the robustness and effectiveness of our approach. This

ensures that the selected gene subset not only maximizes classification performance but also generalizes well to unseen data, reducing the risk of overfitting.

Moreover, the emphasis on simplicity and interpretability in gene selection is a key factor contributing to our method's success. By prioritizing a smaller number of genes without compromising performance, our approach not only improves computational efficiency but also facilitates biological interpretation, enabling researchers to pinpoint the specific genes implicated in Colon cancer pathogenesis. In the Leukemia cancer dataset, our proposed method, along with the work of [7], once again demonstrated the highest performance, achieving a perfect accuracy score of 100%. However, it is noteworthy that [7] method selected a notably smaller number of genes, totaling 26, compared to our method, which selected a total of 40 genes.

The exceptional performance of both our method and Alrefai and Ibrahim's [7] approach in achieving perfect accuracy underscores the effectiveness of the methodologies employed. Despite the disparity in the number of selected genes, both methods successfully identified informative gene subsets crucial for accurate classification of Leukemia cancer samples. Proposed method's selection of a larger gene set may indicate a more comprehensive representation of relevant genetic features associated with Leukemia cancer. However, their ability to achieve comparable performance with fewer genes suggests a more efficient yet effective gene selection strategy.

In conclusion, the results of our comparative analysis across the Ovarian, Colon, and Leukemia cancer datasets highlight the efficacy of the proposed method in achieving superior performance in terms of accuracy and gene selection. The utilization of hybrid filter-wrapper methods, comprehensive gene space exploration, and efficient gene subset selection contribute to the success of our approach. However, it is essential to acknowledge the No-Free-Lunch theorem, which suggests that no single optimization method is universally superior across all problem domains. While the method demonstrates strong performance in these specific datasets, its applicability and effectiveness may vary in other contexts. Thus, further research and validation across diverse datasets are needed to fully assess its potential and limitations.

6 Conclusion and future works

The study aimed to address challenges associated with high-dimensional data, such as the curse of dimensionality, feature redundancy, and data imbalance, by integrating filter methods with the SSA enhanced with a chaotic map. The discussion highlights notable variations in performance metrics across three benchmark datasets (Ovarian, Colon, and Leukemia) when different values of $'n'$ were employed. Results indicate that setting $'n'$ to 10 consistently led to superior performance compared to $'n'$ set to 5, achieving a strategic balance between minimal gene counts and enhanced gene classification accuracy. Furthermore, comparisons with existing studies demonstrate the performance competitiveness of the proposed method, particularly in terms of accuracy and F1 score, across all datasets

Based on the findings of the study, several recommendations are proposed for future research endeavors in this domain. Firstly, experimenting with different values of $'n'$ could enrich the analysis and provide deeper insights into the behavior and performance of the proposed method across a wider range of scenarios. By varying $'n'$ systematically and conducting thorough experimentation, researchers can assess how changes in this parameter affect the algorithm's convergence behavior, solution quality, and overall performance. This comprehensive analysis could help identify an optimal value of $'n'$ that maximizes performance metrics such as accuracy, F1 score, and gene counts across different datasets. Furthermore,

experimenting with different values of $'n'$ could also shed light on the trade-off between computational efficiency and solution quality. Lower values of $'n'$ may lead to faster convergence but could result in suboptimal solutions, while higher values of $'n'$ may enhance solution quality but at the expense of increased computational time and resource requirements. Secondly, further investigation is warranted to explore the applicability and effectiveness of the proposed method across diverse datasets and problem domains. Additionally, efforts should be directed toward enhancing the scalability and computational efficiency of the method to handle larger datasets. Moreover, exploring alternative optimization techniques and hybrid approaches may provide valuable insights into improving gene selection performance. Finally, collaboration with domain experts and biologists is essential to validate the biological relevance of the selected gene subsets and facilitate translational research efforts in cancer diagnosis and treatment.

Funding Statement: The author received no specific funding for this study.

Data Availability: The data that support the findings of this study is available in Ref. [40] of this article.

Conflicts of Interest: The authors declare no conflicts of interest regarding this study.

Authors contributions. Conceptualization: BSA, JI, AA; data curation and methodology: MA, AHH, TSM; validation and visualization: BSA, TSM; writing—original draft preparation: BSA, JI, AHH; writing—review and editing: BSA, TSM; supervision and project administration: BSA, JI, AA. The author had approved the final version.

AI use declaration

During the preparation of this work, generative AI tools were used solely for language editing purposes. The authors reviewed and edited all content and take full responsibility for the final manuscript.

References

[1] Abd-Elnaby, M., Alfonse, M., and Roushdy, M. (2021). "Classification of breast cancer using microarray gene expression data: A survey". *Journal of Biomedical Informatics*, 117, 103764.

[2] Adamu, A., Abdullahi, M., Junaidu, S. B., et al., (2021). "An hybrid particle swarm optimization with crow search algorithm for feature selection". *Machine Learning with Applications*, 6, 100108.

[3] Aguiar, G., Krawczyk, B., and Cano, A. (2024). "A survey on learning from imbalanced data streams: taxonomy, challenges, empirical study, and reproducible experimental framework". *Machine Learning*, 113(7), 4165-4243.

[4] Ali, W., and Saeed, F. (2023). "Hybrid filter and genetic algorithm-based feature selection for improving cancer classification in high-dimensional microarray data". *Processes*, 11(2), 562.

[5] Almazrua, H., and Alshamlan, H. (2022). "A comprehensive survey of recent hybrid feature selection methods in cancer microarray gene expression data". *IEEE ACCESS*, 10, 71427-71449.

[6] Alomari, O. A., Makhadmeh, S. N., Al-Betar, M. A., et al., (2021). "Gene selection for microarray data classification based on Gray Wolf Optimizer enhanced with TRIZ-inspired operators". *Knowledge-Based Systems*, 223, 107034.

[7] Alrefai, N., and Ibrahim, O. (2022). "Optimized feature selection method using particle swarm intelligence with ensemble learning for cancer classification based on microarray datasets". *NEURAL Computing and Applications*, 34(16), 13513-13528.

[8] Alshareef, A. M., Alsini, R., Alsieni, M., et al., (2022). "Optimal deep learning enabled prostate cancer detection using microarray gene expression". *Journal of Healthcare Engineering*, 2022, 1-12.

[9] Alzaqebah, M., Alrefai, N., Ahmed, E. A. E., et al., (2020). "Neighborhood search methods with Moth Optimization algorithm as a wrapper method for feature selection problems". *International Journal of Electrical and Computer Engineering (IJECE)*, 10(4), 3672.

[10] Araujo, D., Doria Neto, A., Martins, A., et al., "Comparative study on dimension reduction techniques for cluster analysis of microarray data". In The 2011 International Joint Conference on Neural Networks: IEEE 1835-1842, (2011).

[11] Arora, S., and Anand, P. (2019). "Chaotic grasshopper optimization algorithm for global optimization". *NEURAL Computing and Applications*, 31(8), 4385-4405.

[12] Arora, S., and Singh, S. (2017). "An improved butterfly optimization algorithm with chaos". *Journal of Intelligent and AMP; FUZZY Systems*, 32(1), 1079-1088.

[13] Aruna, S. and Nandakishore, L. V., "Empirical analysis of the effect of resampling on supervised learning algorithms in predicting the types of lung cancer on multiclass imbalanced microarray gene expression data", *Eai/springer Innovations in Communication and Computing*, Cham: Springer International Publishing, 15-27, (2022).

[14] Assiri, A. S. (2021). "On the performance improvement of Butterfly Optimization approaches for global optimization and Feature Selection". *PLOS ONE*, 16(1), e0242612.

[15] Alhafedh, M. A. A., and Qasim, O. S. (2019). "Two-stage gene selection in microarray dataset using fuzzy mutual information and binary particle swarm optimization". *INDIAN Journal of Forensic Medicine and AMP; Toxicology*, 13(4), 1162.

[16] Baliarsingh, S. K., and Vipsita, S. (2020). "Chaotic emperor penguin optimised extreme learning machine for microarray cancer classification". *IET Systems Biology*, 14(2), 85-95.

[17] Karimi, M., Karimi, Z., Khosravi, M., et al., (2025). "Feature selection methods in big medical databases: a comprehensive survey". *International Journal of Theoretical & Applied Computational Intelligence*, 181–209

[18] Chakraborty, C., Kishor, A., and Rodrigues, J. J. (2022). "Novel Enhanced-Grey Wolf Optimization hybrid machine learning technique for biomedical data computation". *Computers and Electrical Engineering*, 99, 107778.

[19] Chantar, H., Tubishat, M., Essgaer, M., et al., (2021). "Hybrid binary dragonfly algorithm with simulated annealing for feature selection". *SN Computer Science*, 2(4).

[20] Chaudhuri, A., and Sahu, T. P. (2021). "A hybrid feature selection method based on Binary Jaya algorithm for micro-array data classification". *Computers and AMP; Electrical Engineering*, 90, 106963.

[21] Dabba, A., Tari, A., and Meftali, S. (2021). "Hybridization of moth flame optimization algorithm and quantum computing for gene selection in microarray data". *Journal of Ambient Intelligence and Humanized Computing*, 12(2), 2731-2750.

[22] Dabba, A., Tari, A., Meftali, S., et al., (2021). "Gene selection and classification of microarray data method based on mutual information and moth flame algorithm.". *EXPERT Systems with Applications*, 166, 114012.

[23] Das, S., and Saha, P. (2021). "Performance of swarm intelligence based chaotic meta-heuristic algorithms in civil structural health monitoring". *Measurement*, 169, 108533.

[24] Dash, R., Dash, R., and Rautray, R. (2022). "An evolutionary framework based microarray gene selection and classification approach using binary shuffled frog leaping algorithm". *Journal of King Saud University - Computer and Information Sciences*, 34(3), 880-891.

[25] Dhal, P., and Azad, C. (2022). "A comprehensive survey on feature selection in the various fields of machine learning". *Applied Intelligence*, 52(4), 4543-4581.

[26] Dorigo, M. and Stützle, T., "The Ant Colony Optimization metaheuristic: algorithms, applications, and advances", *International Series in Operations Research and amp; Management Science*, Boston, MA: Springer US, 250-285, (2003).

[27] Elgamal, Z., Sabri, A. Q. M., Tubishat, M., et al., (2022). "Improved reptile search optimization algorithm using chaotic map and simulated annealing for feature selection in medical field". *IEEE Access*, 10, 51428-51446.

[28] Founta, K., Dafou, D., Kanata, E., et al., (2023). "Gene targeting in amyotrophic lateral sclerosis using causality-based feature selection and machine learning". *Molecular Medicine*, 29(1).

[29] Kanti Ghosh, K., Begum, S., Sardar, A., et al., (2021). "Theoretical and empirical analysis of filter ranking methods: Experimental study on benchmark DNA microarray data". *Expert Systems with Applications*, 169, 114485.

[30] Ghosh, M., Guha, R., Alam, I., et al., (2019). "Binary Genetic Swarm Optimization: A combination of ga and pso for feature selection". *Journal of Intelligent Systems*, 29(1), 1598-1610.

[31] Hakak, S., Alazab, M., Khan, S., et al., (2021). "An ensemble machine learning approach through effective feature extraction to classify fake news". *Future Generation Computer Systems*, 117, 47-58.

[32] Hambali, M. A., Oladele, T. O., and Adewole, K. S. (2020). "Microarray cancer feature selection: Review, challenges and research directions". *International Journal of Cognitive Computing in Engineering*, 1, 78-97.

[33] Mohd Hasri, N. N., Wen, N. H., Howe, C. W., et al., (2017). "Improved Support Vector Machine using multiple SVM-RFE for cancer classification". *International Journal on Advanced Science, Engineering and Information Technology*, 7(4-2), 1589.

[34] Hassan, I. H., Abdullahi, M., Isuwa, J., et al., (2023). "Microarray cancer gene selection using pelican optimization algorithm". *The Journal of Contents Computing*, 5(1), 609-620.

[35] Hassan, I. H., Mohammed, A., Ali, Y. S., Jeremiah, I., and Abdulraheem, S. A., "Metaheuristic algorithms in text clustering", *Comprehensive Metaheuristics*, Elsevier, 131-152, (2023).

[36] Huda, R. K., and Banka, H. (2020). "New efficient initialization and updating mechanisms in PSO for feature selection and classification". *NEURAL Computing and Applications*, 32(8), 3283-3294.

[37] Isuwa, J., Abdullahi, M., Sahabi Ali, Y., et al., (2022). "Hybrid particle swarm optimization with sequential one point flipping algorithm for feature selection". *Concurrency and Computation: Practice and Experience*, 34(25).

[38] Isuwa, J., Abdullahi, M., Ali, Y. S., et al., (2023). "Optimizing microarray cancer gene selection using swarm intelligence: Recent developments and an exploratory study". *Egyptian Informatics Journal*, 24(4), 100416.

[39] Nasir, S., Bilal, M., and Khalidi, H. (2025). "Detection and classification of skin cancer by using CNN-enabled cloud storage data access control algorithm based on blockchain technology". *International Journal of Theoretical and Applied Computational Intelligence*, 2025, 145–169.

[40] Jeremiah, I. (2023). "Towards an improved particle swarm optimization for feature selection: a survey". *SLU Journal of Science and Technology*, 59-73.

[41] Jeremiah, I., Abdullahi, M., Yusuf, S. A., et al., "Integrating Local Search Methods in Metaheuristic Algorithms for Combinatorial Optimization: The Traveling Salesman Problem and its Variants". In 2022 IEEE Nigeria 4th International Conference on Disruptive Technologies for Sustainable Development (NIGERCON): IEEE 1-5, (2022).

[42] Jeremiah, I., Rosemary, et al., (2023). "Microarray Cancer Gene Selection using Swarm Intelligence: An Ensemble Approach". *International Conference on Computing and Advances in Information Technology (ICCAIT 2023). Unpublished WORK, November*, 21–23.

[43] Jia, L., Wang, T., Gad, A. G., et al., (2023). "A weighted-sum chaotic sparrow search algorithm for interdisciplinary feature selection and data classification". *Scientific Reports*, 13(1).

[44] Jiang, Z. "Discrete bat algorithm for traveling salesman problem". In 2016 3rd International Conference on Information Science and Control Engineering (ICISCE): IEEE 343-347, (2016).

[45] Kalra, M., Kumar, V., Kaur, M., et al., (2022). "A novel binary emperor penguin optimizer for feature selection tasks". *Computers, Materials and Continua*, 70(3), 6239-6255.

[46] Ke, L., Li, M., Wang, L., et al., (2023). "Improved swarm-optimization-based filter-wrapper gene selection from microarray data for gene expression tumor classification". *Pattern Analysis and Applications*, 26(2), 455-472.

[47] Kennedy, J., and Eberhart, R. "Particle swarm optimization". In Proceedings of ICNN'95 - International Conference on Neural Networks: IEEE 1942-1948, (1995).

[48] Abu Khurma, R., Aljarah, I., Sharieh, A., et al., (2022). "A review of the modification strategies of the nature inspired algorithms for feature selection problem". *Mathematics*, 10(3), 464.

[49] Kumar, M. and Rath, S., "Feature selection and classification of microarray data using machine learning techniques", *Emerging Trends in Applications and Infrastructures for Computational Biology, Bioinformatics, and Systems Biology*, 213-242, (2016).

[50] Kundu, R., Chattopadhyay, S., Cuevas, E., et al., (2022). "AltWOA: Altruistic Whale Optimization Algorithm for feature selection on microarray datasets". *Computers in Biology and Medicine*, 144, 105349.

[51] Layer, R.," G*enetic mutations can be benign or cancerous – a new method to differentiate between them could lead to better treatments*". The Conversation, (2022).

[52] Li, M., Ke, L., Wang, L., et al., (2023). "A novel hybrid gene selection for tumor identification by combining multifilter integration and a recursive flower pollination search algorithm". *Knowledge-Based Systems*, 262, 110250.

[53] Liu, L., Wei, Z., and Xiang, H. (2022). "A novel image encryption algorithm based on compound-coupled logistic chaotic map". *Multimedia TOOLS and Applications*, 81(14), 19999-20019.

[54] Liu, Y., Liu, Y., Yu, B. X., et al., (2023). "Noise-robust oversampling for imbalanced data classification". *Pattern Recognition*, 133, 109008.

[55] Mirjalili, S., Gandomi, A. H., Mirjalili, S. Z., et al., (2017). "Salp Swarm Algorithm: A bio-inspired optimizer for engineering design problems". *Advances in Engineering Software*, 114, 163-191.

[56] Mohd Ali, N., Besar, R., and Ab. Aziz, N. A. (2022). "Hybrid feature selection of breast cancer gene expression microarray data based on metaheuristic methods: a comprehensive review". *Symmetry*, 14(10), 1955.

[57] Naik, R. B., and Singh, U. (2024). "A review on applications of chaotic maps in pseudo-random number generators and encryption". *Annals of Data Science*, 11(1), 25-50.

[58] Nouri-Moghaddam, B., Ghazanfari, M., and Fathian, M. (2023). "A novel bio-inspired hybrid multi-filter wrapper gene selection method with ensemble classifier for microarray data". *Neural Computing and Applications*, 35(16), 11531-11561.

[59] Petinrin, O. O., Saeed, F., Salim, N., et al., (2023). "Dimension reduction and classifier-based feature selection for oversampled gene expression data and cancer classification". *Processes*, 11(7), 1940.

[60] Prabhakar, S. K., and Lee, S. (2020). "Transformation based tri-level feature selection approach using wavelets and swarm computing for prostate cancer classification". *IEEE Access*, 8, 127462-127476.

[61] Ravindran, U., and Gunavathi, C. (2023). "A survey on gene expression data analysis using deep learning methods for cancer diagnosis". *Progress in Biophysics and Molecular Biology*, 177, 1-13.