



Feature Selection Problem and Metaheuristics: A Systematic Literature Review about Its Formulation, Evaluation and Applications

José Barrera-García ¹, Felipe Cisternas-Caneo ¹, Broderick Crawford ^{1,*}, Mariam Gómez Sánchez ² and Ricardo Soto ¹

¹ Escuela de Ingeniería Informática, Pontificia Universidad Católica de Valparaíso, Avenida Brasil 2241, Valparaíso 2362807, Chile; jose.barrera@pucv.cl (J.B.-G.); felipe.cisternas.c@mail.pucv.cl (F.C.-C.); ricardo.soto@pucv.cl (R.S.)

² Departamento de Electrotecnia e Informática, Universidad Técnica Federico Santa María, Federico Santa María 6090, Viña del Mar 2520000, Chile; mariam.gomez@usm.cl

* Correspondence: broderick.crawford@pucv.cl

Abstract: Feature selection is becoming a relevant problem within the field of machine learning. The feature selection problem focuses on the selection of the small, necessary, and sufficient subset of features that represent the general set of features, eliminating redundant and irrelevant information. Given the importance of the topic, in recent years there has been a boom in the study of the problem, generating a large number of related investigations. Given this, this work analyzes 161 articles published between 2019 and 2023 (20 April 2023), emphasizing the formulation of the problem and performance measures, and proposing classifications for the objective functions and evaluation metrics. Furthermore, an in-depth description and analysis of metaheuristics, benchmark datasets, and practical real-world applications are presented. Finally, in light of recent advances, this review paper provides future research opportunities.

Keywords: feature selection problem; optimization; metaheuristics; classifier; evaluation metrics; objective function; systematic literature review



Citation: Barrera-García, J.; Cisternas-Caneo, F.; Crawford, B.; Gómez Sánchez, M.; Soto, R. Feature Selection Problem and Metaheuristics: A Systematic Literature Review about Its Formulation, Evaluation and Applications. *Biomimetics* **2024**, *9*, 9. <https://doi.org/10.3390/biomimetics9010009>

Academic Editor: Heming Jia

Received: 25 November 2023

Revised: 16 December 2023

Accepted: 18 December 2023

Published: 25 December 2023



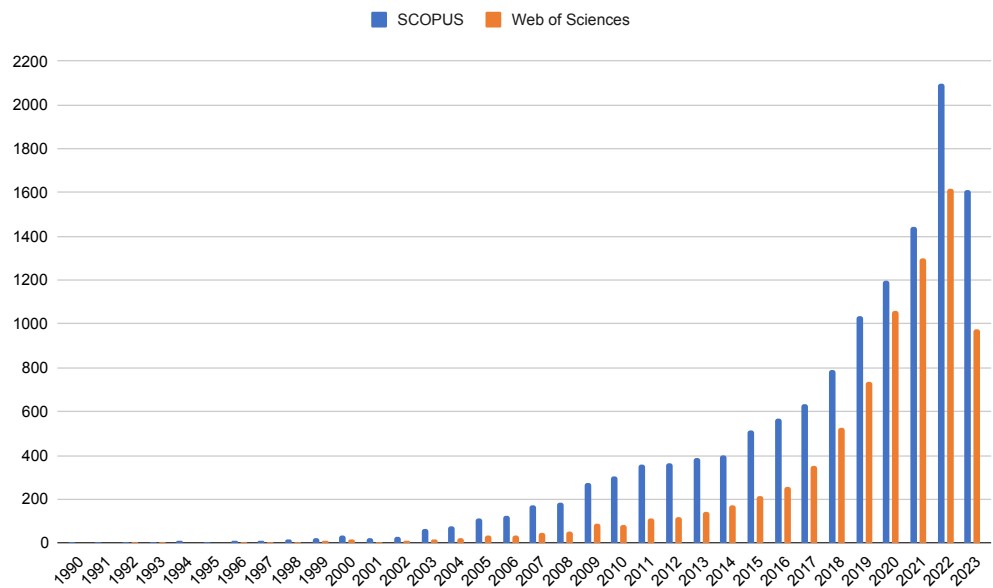
Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Recently, the interest in artificial intelligence, particularly machine learning, has increased. The great success of this technology is due to the extensive computing capacity now available and the vast volumes of existing data. The high level of available data has allowed machine learning algorithms to become increasingly complex and they can be nourished from different sources. This can be a great benefit, but it can also be a problem since the data may be redundant and irrelevant, causing learning errors [1]. In this context, a set of features that describe the problem studied is defined based on existing data. When the datasets used are large, extensive sets of features are generated, and eliminating irrelevant information is of the utmost importance. The feature selection problem consists of finding a subset of features that represents the original dataset with the aim of eliminating irrelevant and redundant information within the dataset to improve the performance of a classification algorithm [2]. This problem is complex since the search space is defined as 2^n , where n corresponds to the number of features that make up the dataset [3].

To initially analyze the information related to the feature selection problem, we performed a simple search on Web of Science and Scopus using the terms “Feature Selection” and “Optimization”, obtaining 8016 and 12,908 research papers, respectively. Figure 1 shows the number of publications per year, evidencing the increase in the study of the problem, especially in recent years.

Paper published by years indexed SCOPUS and Web of Sciences

**Figure 1.** Papers published by year indexed Scopus and Web of Science.

Given the extensive number of investigations on the topic, many methods for solving the problem have emerged. Following the extensive related literature [3–14], the solution methods for the feature selection problem can be classified as shown below:

- Filter methods identify the optimal set of features by focusing on the specificities of the problem within the dataset without considering the classification algorithm to be used. These methods use statistical analysis to explore the connection between each input and target variable, assigning a relevance value to each feature. They stand out for their speed and computational efficiency. Examples of these methods include the correlation coefficient, the chi-squared test, and the Fisher score.
- Wrapper methods approach the feature selection iteratively, continuously adjusting the subset of features based on the training phase of the machine learning model. These methods offer a set of features ideally suited to the needs of the model and often performance improvement. Among its most well-known categories are forward selection, backward elimination, exhaustive selection, and metaheuristics.
- Embedded methods were introduced to overcome the difficulties filter and wrapper methods face. The purpose is to obtain quick results and with greater accuracy. Examples include lasso regression, decision trees, and random forest algorithms.

Wrapper methods are computationally more expensive than filter methods; however, the former delivers better results. Metaheuristics stand out within the wrapper methods.

Metaheuristics are a general purpose algorithm that with few modifications can solve different optimization problems. They are algorithms with stochastic behavior whose optimization process is performed by balancing the exploration of the search space and the exploitation of promising regions [15], and these features make metaheuristics deliver high-quality results in a reasonable time. In the literature, we can find different metaheuristics that are inspired by physical phenomena, evolutionary theory, and animal social behavior [16]. The great variety of existing metaheuristics is due to the no free lunch theorem, which indicates that there is no optimization algorithm capable of solving all existing optimization problems [17–19]. In other words, this theorem motivates us to continue innovating and experimenting with new metaheuristics, making hybridizations, and developing modifications to metaheuristics. In this sense, Becerra-Rozas et al. [20] reviewed the literature related to the binarization of continuous metaheuristics to solve combinatorial

problems, finding that the feature selection problem is highly studied and providing inspiration to continue exploring the field of metaheuristic binarization. In addition, in the related literature, several literature reviews address the use of metaheuristics to solve the problem. These works mainly focus on aspects related to the objective function, evaluation metrics, metaheuristics, classifiers, benchmarks, and real-world applications. Table 1 shows a summary of the contributions found since 2015 in the literature ordered by year of publication. The "✓" indicates that the literature review analyzed includes the field seen in the column. The last row indicates the contributions we made in our literature review.

Table 1. Comparison with other systematic literature reviews.

Paper	Year	Objective Function	Evaluation Metrics	Optimization Techniques	Classifier	Benchmark Application	Real-World Application
[4]	2023	✓		✓	✓		
[5]	2023			✓			
[6]	2023		✓	✓	✓		✓
[7]	2023	✓	✓	✓	✓	✓	
[8]	2022			✓			
[9]	2022	✓		✓	✓		✓
[10]	2022	✓	✓	✓	✓	✓	
[11]	2022	✓		✓			✓
[12]	2022			✓	✓		
[13]	2021		✓	✓		✓	
[3]	2021		✓	✓	✓	✓	
[14]	2020	✓		✓	✓	✓	
Our Work		✓	✓	✓	✓	✓	✓

In most literature reviews, the authors do not detail the objective functions pursued in the corresponding compiled works. Similarly, the evaluation metrics used by the authors are only defined in detail in [3,13]. On the other hand, the metaheuristics used to solve the problem are the common aspects studied in the mentioned reviews. The authors address the binarization of metaheuristics in [4–6,10]. For example, in [5], the authors detail whether the metaheuristics were binarized or modified with chaotic maps. In addition, different classifications based on behavior, hybridization, the main modifications carried out, and real-world applications, among others, are presented in [4–6,8,9,11,13]. Also, in investigations such as [3,7–9,12,13], the authors indicate the contribution made by the metaheuristic to the field of feature selection. The works that address the different classifiers commonly define each classifier used or at least the most common ones and, for example, in [6], the authors additionally indicate the statistical tests used to validate the obtained results. Regarding the datasets used, classifications are also presented. For example, in [7], the authors classify the datasets used according to size, field, and number of classes. Finally, applications in the real world are also studied; these span across various fields, including, but not limited to, healthcare [21], cybersecurity [22], environmental monitoring [23], and text classification.

As a result of the analysis, we found several aspects not addressed in detail by the existing reviews. This study conducts a systematic literature review presenting a comprehensive taxonomy of objective functions, categorized into single-objective and multi-objective functions. Similarly, a classification is proposed for metrics based on four categories: classifier, metaheuristics, features, and statistical test. In addition, the metaheuristics used to solve the problem are analyzed in depth, emphasizing the implementation details and hybridization. Also, regarding benchmarks and real-world applications, a thorough categorization of repositories is presented, providing standardized, pertinent dataset information.

Based on this, the contributions of this research are the following:

- An updated review of the literature analyzing and discussing objective functions proposed for the feature selection problem as well as metrics, classifiers, and meta-

heuristics used to solve the problem and benchmarks or real-world applications to validate the results obtained;

- A detailed classification of the objective functions and evaluation metrics provides a better understanding of the status of several aspects.
- A deep analysis of the metaheuristics used by researchers, following different criteria.

The remainder of this document is structured as follows. In Section 2, the applied methodology is presented, also detailing the research questions. In Section 3, a bibliometric analysis of the selected research papers is presented. The research questions proposed in Section 2 are answered in Section 4. Finally, in Section 5, the conclusions of the research and some lines of future work are presented.

2. Methodology

The methodology is crucial in ensuring a robust and comprehensive research analysis. This research was conducted following the systematic literature review (SLR) framework [24]. In an SLR, one of the main steps is the definition of the research questions. These questions serve as a compass, directing our exploration and analysis in the area. In this work, the research questions were defined as follows:

- RQ1. How is the objective function of the feature selection problem formulated?
- RQ2. What metrics are used to analyze the performance of the feature selection problem?
- RQ3. What machine learning techniques have been used to calculate fitness in the feature selection problem?
- RQ4. What metaheuristics have been used to solve the feature selection problem?
- RQ5. Which datasets are commonly used as benchmarks, and which are derived from real-world applications?

For the literature review search process, we used six databases well known to the scientific community: Scopus, Web of Science, IEEE Xplore, ScienceDirect by Elsevier, Wiley, and SpringerLink. Our date range was defined between 2019 and 2023 (20 April 2023), and our initial keyword was “Feature Selection”, focusing primarily on titles. Table 2 shows the information regarding the search process for each database. The database is shown in column 1. Columns 2–3 show the query performed and the number of investigations obtained in each database. Owing to the differences in the search and filtering capabilities of these databases, some manual processes were required after obtaining the search results. This involved a refinement process, applying manual year filtering in the case of IEEE Xplore and SpringerLink, and expanding our search criteria.

Table 2. Summary of database searches for feature selection literature (2019–2023).

Database	Query	#Result
• IEEE Xplore	(“Document Title”:“Feature Selection”) and Filters Applied: 2019 - 2023	2204
• ScienceDirect by Elsevier	Title field: “Feature Selection” and Year field: “2019–2023”	1388
• Scopus	TITLE (“feature selection”) AND PUBYEAR > 2018 AND PUBYEAR < 2024 AND (LIMIT-TO (LANGUAGE,“English”)	8812
• SpringerLink	Title field: “Feature Selection”	3006
• Web of Sciences	(TI=(“feature selection”)) AND (DT==(“ARTICLE” OR “REVIEW”) AND LA==(“ENGLISH”) AND PY==(“2023” OR “2022” OR “2021” OR “2020” OR “2019”))	4713
• Wiley	[Publication Title: “feature selection”] AND [Earliest: (01/01/2019 TO 04/20/2023)]	220

Our process of refining the information obtained was based on two main phases, the first making use of the tools provided by the databases, using as inclusion criteria manuscripts that (a) present the title, doi, and abstract, (b) are not duplicates, (c) are published in journals (not in conferences or book chapters), and (d) containing specific words in the abstract, in this case, the phrase “Feature selection problem”. Subsequently, we carried out a manual filtering process on the 190 papers obtained where we verified the

previous inclusion criteria as well as manuscripts (e) written in English, (f) within the scope of this research, and (g) not classified as a survey, review, or SLR. Note that this research's scope refers to manuscripts that use a metaheuristic and present metrics to measure the performance of the proposals and/or the classifier used and/or the optimized objective function and/or the reference datasets/real-world data used. After this process, the final number of manuscripts analyzed in this literature review was 161. Figure 2 shows the process of filtering the collected literature.

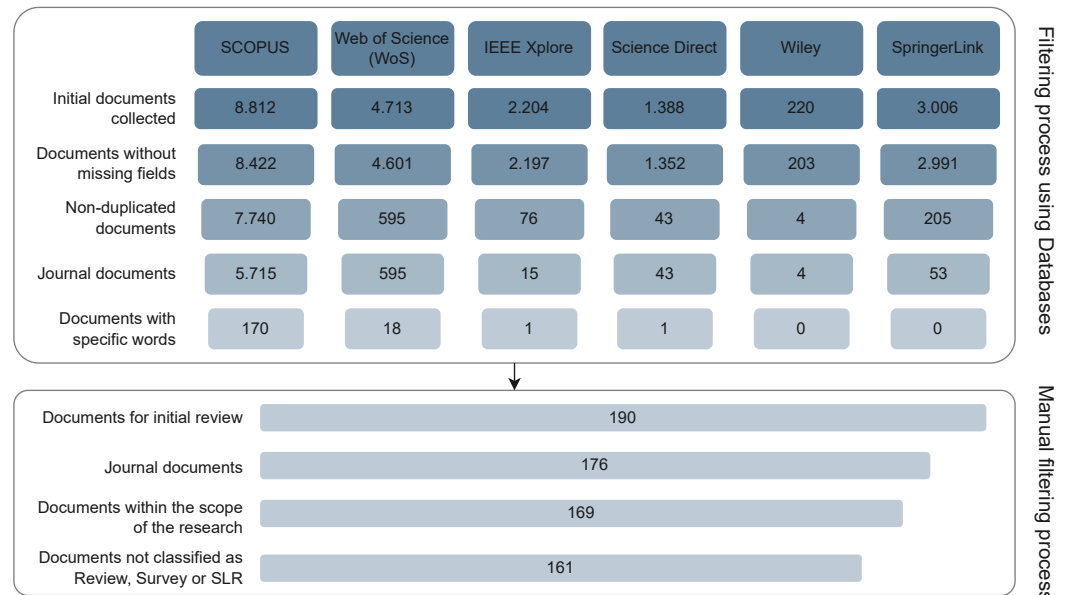


Figure 2. Flowchart of filtering process for the systematic literature review.

3. Bibliometric Analysis

To perform a bibliometric analysis, the keywords, year of publication, journal, number of citations, authors, and country of the institution represented by each author were extracted from the selected articles. We used Biblioshiny by [25], a Bibliometrix application developed in R that is open access.

Figure 3 shows a network graph plotting the keywords found in the collected papers. Each node symbolizes a keyword, and the size of each node reflects the frequency of occurrence of the corresponding keyword. In the center of this graph are two central nodes, “feature selection” and “classification”, linked together, which means there is a strong correlation between these topics. Additionally, the graph is organized into four main groups, indicated by a different color: blue, red, green, and purple. This codification of colors is an effective network graph technique for visually differentiating groups of nodes that often interact or are related. These clusters suggest a typical grouping pattern of these keywords, revealing underlying connections and thematic consistencies within the research field.

IEEE Access is the journal with the most research related to the feature selection problem, followed by Expert System with Applications. These data are in accordance with the most cited papers on the feature selection problem. Figure 4 shows the ten journals with the most research on the feature selection problem. Figure 5 shows the top ten cited papers, with the second and third places corresponding to articles published in IEEE Access, and five of the ten most cited papers having been published in Expert System with Applications. In both figures, blue and violet represent the information relating to IEEE Access and Expert System with Applications, respectively.

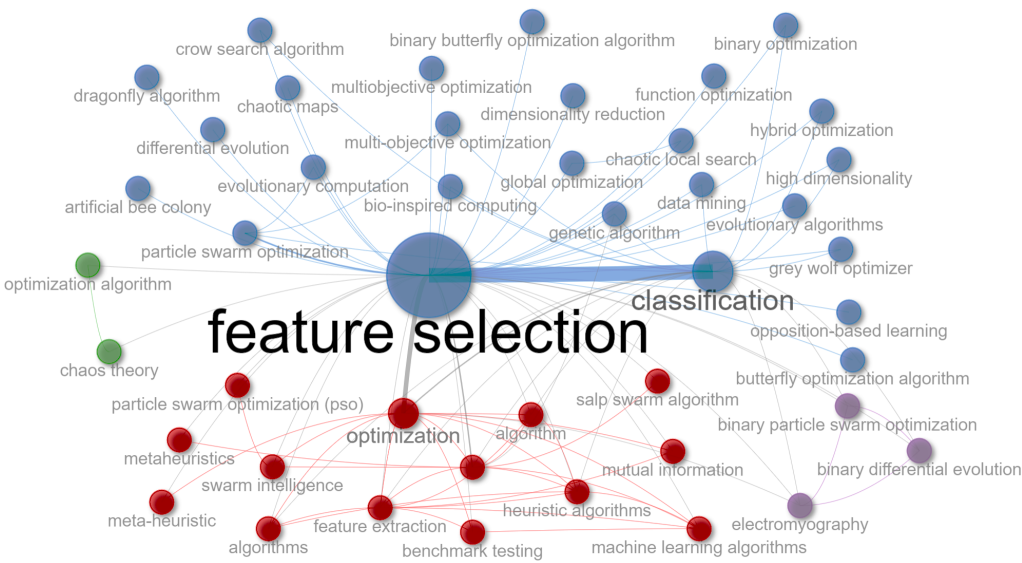


Figure 3. Network map of the keywords found.

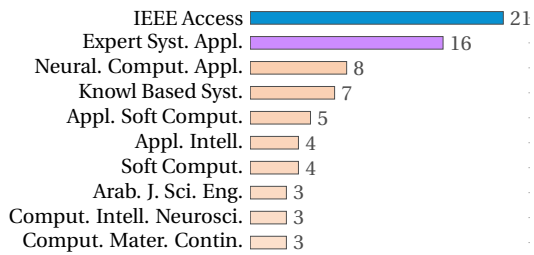


Figure 4. Research by journal.

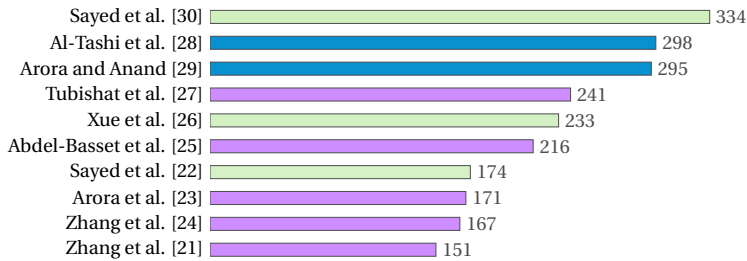


Figure 5. Citations by author [21–30].

On the other hand, when analyzing the contributions by country, we found that China presents the most significant number of published papers related to the feature selection problem. This is in accordance with the contributions by authors, where five of the ten most active authors represent China, four represent Malaysia, and one represents Australia. Note that the three countries represented by the ten most contributing authors in the area are among the ten countries that present the most significant number of publications. Figure 6 shows the number of investigations on the feature selection problem for the ten most prominent countries. Figure 7 shows the annual contribution of the ten most representative authors in the area. The colors blue, orange, green, red, and gray in the bars represent the papers published from 2019 to 2023, respectively. The color of the name of the author is related to the country represented

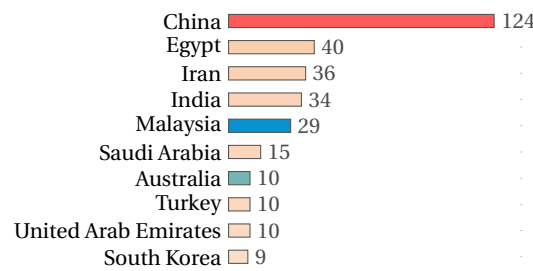


Figure 6. Research by country.

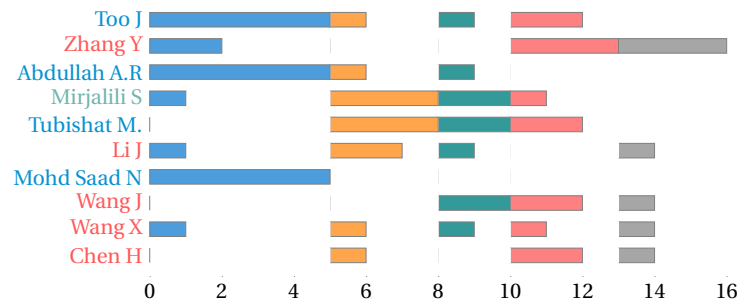


Figure 7. Research by authors by years.

4. Discussion

4.1. How Is the Objective Function of the Feature Selection Problem Formulated?

Optimization problems are composed of an objective function subject to constraints. The objective functions can be classified into two main categories: single-objective, focused on optimizing only one objective; and multi-objective, focused on optimizing several objective functions at the same time. There are two ways of representing multi-objective optimization problems: (1) a pure multi-objective function, and (2) a weighted multi-objective function. Figure 8 summarizes the classification of the objective functions found in the papers collected.

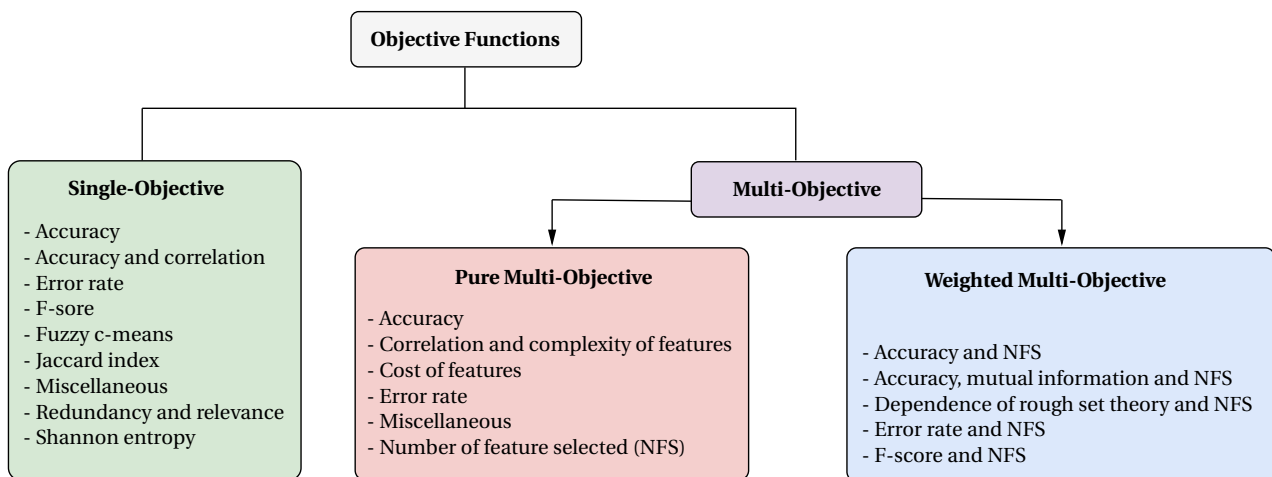


Figure 8. Taxonomy of objective functions.

Multi-objective functions are pursued in 73% of the collected research, with the weighted multi-objective function being the most used objective function classification in the related literature. This trend occurs in general and over the years. Figure 9 shows the number of papers that have pursued different objective function classifications in general (left) and their trend over the years (right).

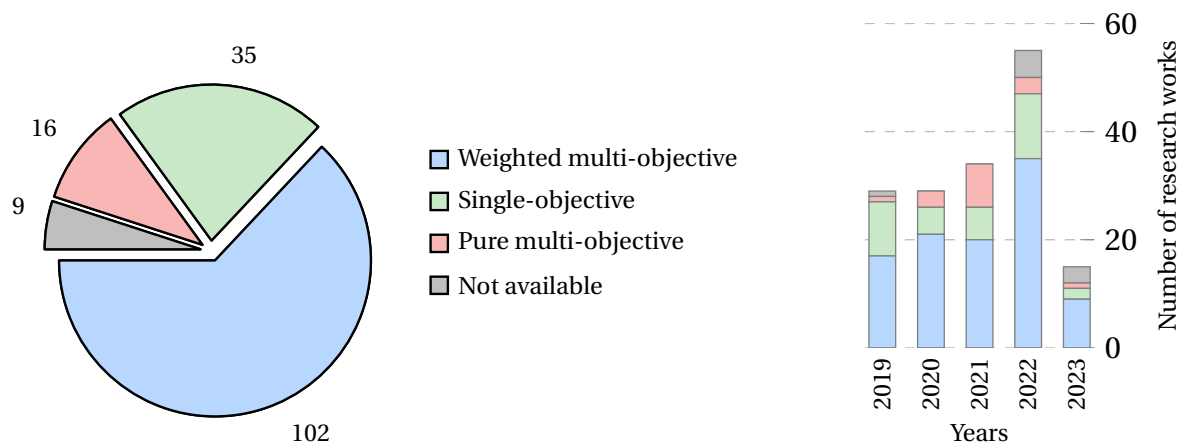


Figure 9. Objective functions by category.

4.1.1. Single-Objective Functions

Single-objective functions are focused on optimizing only one objective function subject to constraints. Mathematically, single-objective optimization problems are modeled as follows [26]:

$$\min \text{ or } \max f(X) \quad (1)$$

Subject to

$$\begin{aligned} g_i(X) &< 0 \quad i = 1, 2, \dots, N_{ieq} \\ h_i(X) &= 0 \quad i = 1, 2, \dots, N_{eq} \end{aligned} \quad (2)$$

where $f(X)$ represents the objective function, X corresponds to the solution vector composed of the decision variables, and $g_i(X)$ and $h_i(X)$ are the inequality and equality constraints, respectively. Within this category, eight different functions were detected:

(a) **Accuracy**: Defined in detail in Section 4.2.1 and mathematically in Equation (22). This objective function was pursued in [21,27–41].

(b) **Error rate**: Defined in detail in Section 4.2.1 and mathematically in Equation (26). This objective function was pursued in [42–51].

(c) **Fuzzy c-means (FCM)**: Clustering algorithm which returns a cost function used to calculate the performance of the metaheuristic [52]. This objective was pursued in [53].

(d) **Redundancy and relevance**: Redundancy has been used to quantify the similarity level between selected features. Relevance represents the relevance between features and categorical variables reflecting the recognition ability of the selected features. We found two papers that pursued relevance and redundancy. In [54], the authors use these metrics to calculate the objective function and relate them by subtraction, as follows:

$$F(X) = \text{Relevance} - \text{Redundancy} \quad (3)$$

On the other hand, in [55], the authors relate these metrics by means of a division, as shown below:

$$F(X) = \frac{\text{Redundancy}}{\text{Relevance}} \quad (4)$$

(e) **Accuracy and correlation**: In [56], the authors present an objective function that relates the correlation between the selected features without the presence of class labels and the accuracy. This objective function was pursued in [56] and mathematically is defined as follows:

$$F(X) = \frac{A + (1 - M)}{2} \quad (5)$$

where A is the accuracy and M is the computed correlation.

(f) **Shannon entropy**: Measures the amount of information in a distribution. If a distribution has a high entropy value, it contains more information. The authors of [57] used this information as an objective function, defining it mathematically as follows:

$$F(X) = - \sum_{i=1}^n p(x_i) \log_2 p(x_i) \quad (6)$$

where n corresponds to the number of features and $p(x_i)$ is the probability of occurrence of a feature.

(g) **Humming loss**: Defined in detail in Section 4.2.1 and mathematically in Equation (23). This objective function was pursued in [22].

(h) **Jaccard index**: Measures the similarity and overlap between two sets. It is often used in data analysis, information retrieval, and text mining. This objective function was pursued in [58] and mathematically is defined as follows:

$$F(X) = \frac{TP}{TP + FP + FN} \quad (7)$$

where TP (true positive) is the number of positive instances correctly classified; FP (false positive) is the number of negative instances wrongly classified as positive; and FN (false negative) is the number of positive instances wrongly classified as negative.

(i) **Miscellaneous**: Finally, we detected two objective functions which the authors explain in greater detail in the respective papers; see [59,60].

Accuracy is the most studied single-objective function in the collected literature, present in 10% of the research, and maintains a stable behavior in terms of research per year. Figure 10 shows the number of papers by year that have pursued the two most studied single-objective functions in the collected literature.

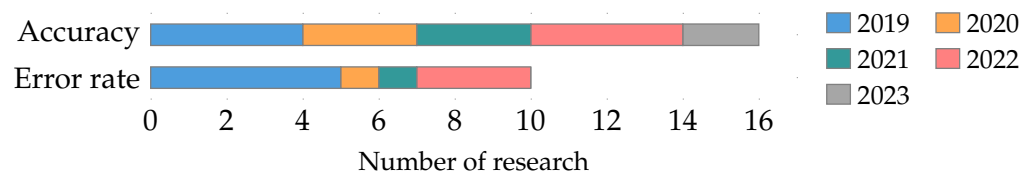


Figure 10. Single-objective functions by year.

4.1.2. Pure Multi-Objective Functions

Pure multi-objective functions are focused on independent optimization of each objective function. Thus, Pareto dominance is used to determine the best solution. Mathematically, multi-objective optimization problems are modeled as follows [61]:

$$\min \text{ or } \max f_1(X), f_2(X), \dots, f_m(X) \quad (8)$$

Subject to

$$\begin{aligned} g_i(X) &< 0 \quad i = 1, 2, \dots, N_{ieq} \\ h_i(X) &= 0 \quad i = 1, 2, \dots, N_{eq} \end{aligned} \quad (9)$$

where $f_1(X), f_2(X), \dots, f_m(X)$ represents the m objective functions to be optimized, X corresponds to the solution vector composed of the decision variables, and $g_i(X)$ and $h_i(X)$ are the inequality and equality constraints, respectively. Within the category of pure multi-objective functions, six different functions were detected:

(a) **Error rate:** Defined in Section 4.2.4 and mathematically in Equation (26). This objective function has been used in [62–73].

(b) **Number of features selected (NFS):** One of the essential aspects when solving the feature selection problem is to increase the performance of the classifiers to the smallest number of features possible. Given this, the number of selected features is an important objective, pursued in [63–74].

(c) **Cost of features:** In [62], the authors incorporate the costs associated with features to the feature selection problem, minimizing the costs associated with the features and the error rate of the classification algorithms.

(d) **Accuracy:** Defined in Section 4.2.3 and mathematically in Equation (22). This objective function was pursued in [74].

(e) **Correlation and complexity of features:** In [63], the authors propose four different metrics to build the objective function, and join to the error rate and the correlation and complexity of the features.

(f) **Miscellaneous:** In [75], the authors use six different metrics to build the objective function, defined in Section 4.2.1 and named as follows:

- $F_1(S)$ = Number of features selected;
- $F_2(S)$ = Accuracy;
- $F_3(S)$ = Relevance;
- $F_4(S)$ = Redundancy;
- $F_5(S)$ = Interclass Distance;
- $F_6(S)$ = Intraclass Distance.

Thus, the objective function is defined as follows:

$$\min F(X) = F_1(S), -F_2(S), -F_3(S), F_4(S), -F_5(S), F_6(S) \quad (10)$$

Error rate and NFS are the most studied pure multi-objective functions in the collected literature, each one is studied in 7% of the research and shows a considerable increase in 2021. Figure 11 shows the number of papers by year that have pursued the two most studied pure multi-objective functions in the collected literature.



Figure 11. Pure multi-objective functions by year.

4.1.3. Weighted Multi-Objective Functions

In general, metaheuristics are designed to solve single-objective optimization problems, and adapting them to multi-objective optimization problems is very costly both computationally and in development time. In [76], the authors present a way to translate a multi-objective optimization problem into a single-objective optimization problem. This procedure is a weighted sum of all the objective functions, and mathematically it is defined as follows:

$$\min \text{ or } \max f(X) = w_1f_1 + w_2f_2 + \dots + w_mf_m \quad (11)$$

Subject to

$$\begin{aligned} w_i &\geq 0 \quad \forall i = 1, 2, \dots, m \\ w_1 + w_2 + \dots + w_m &= 1 \end{aligned} \quad (12)$$

where w_1, w_2, \dots, w_m are non-negative weights for m objective functions. Within the category of weighted multi-objective functions, five different ones were detected:

(a) Error rate and number of features selected (error rate and NFS): Within the feature selection problem, it is essential to improve the performance of the classifier and reduce the number of features. Given this, a weighted multi-objective function that relates these two terms was proposed as follows:

$$F(X) = \alpha \cdot \text{Error Rate} + \beta \cdot \frac{S}{F} \quad (13)$$

where S and F correspond to the number of features selected and the total number of features of the dataset, and α and β assign the importance of the error rate and the number of features selected. α and $\beta \in [0, 1]$, and there is no consensus on the values of these parameters. This objective function was pursued in [77–161].

(b,c) Accuracy and number of features selected (accuracy and NFS): Objective function similar to error rate and NFS. The difference is that the accuracy is the metric of the classification technique. In the literature, two objective functions that associate accuracy with the number of features selected were detected.

The first version found is defined as follows:

$$F(X) = \text{Accuracy} + \alpha \cdot \left(1 - \frac{S}{F}\right) \quad (14)$$

where S and F correspond to the number of features selected and the total number of features of the dataset, and α assigns the importance of the number of features selected. $\alpha \in [0, 1]$ and there is no consensus on the values of this parameter. This objective function was pursued in [130,162–165].

The second version found is defined as follows:

$$F(X) = \alpha \cdot \text{Accuracy} + \beta \cdot \left(\frac{F - S}{F}\right) \quad (15)$$

where S and F correspond to the number of features selected and the total number of features of the dataset, and α and β assign the importance of the accuracy and the number of features selected. α and $\beta \in [0, 1]$, and there is no consensus on the values of these parameters. This objective function was pursued in [166–174].

(d) F-score and number of features selected (F-score and NFS): This objective function has only been studied in [175] and mathematically is defined as follows:

$$\min F(X) = w_1 z_1 + w_2 z_2 + w_3 z_3 \quad (16)$$

where $w_1 + w_2 + w_3 = 1$ and the authors determine that $w_1 = 0.5$, $w_2 = 0.25$, and $w_3 = 0.25$. z_1 , z_2 , and z_3 are defined as follows:

$$\min z_1 = 1 - \text{F-score} \quad (17)$$

$$\min z_2 = \frac{|S|}{T} \quad (18)$$

$$\min z_3 = \frac{\max\{t | x_t \in S\}}{T} \quad (19)$$

where z_1 aims to maximize the F-score, z_2 seeks to minimize the number of features selected per unit of time, and z_3 pursues the minimize the last feature to be selected.

(e) Accuracy, mutual information, and number of features selected: This objective function has only been studied in [176] and mathematically, is defined as follows:

$$F(X) = \alpha \cdot \text{Accuracy} + \beta \cdot \left(\frac{|F - S|}{F}\right) + \gamma \cdot \text{Mean}(I(X_k; Y)) \quad (20)$$

where $|S|$ is the number of selected features, $\alpha \cdot \text{Accuracy}$, $\beta \left(\frac{|F-S|}{F} \right)$ and $\delta \cdot \text{Mean}(I(X_k; Y))$ are considered for increasing the classification accuracy, reducing the number of selected features, and increasing the mean of the mutual information, respectively. α , β , and $\delta \in [0, 1]$ and the sum equals 1.

(f) **Dependence of rough set theory and number of features selected:** This objective function has only been studied in [177] and mathematically is defined as follows:

$$F(X) = \alpha \cdot \text{dep}(X) + \beta \cdot \frac{1}{S} \quad (21)$$

where X is the feature subset found. Fitness is calculated based on the dependency measure of rough set theory $\text{dep}(X)$, and S is the length of the feature subset size(x_i). $\alpha \in [0, 1]$ controls the relative weight of the dependency value and feature subset length, and β is $(1 - \alpha)$.

The objective functions found in this classification combine the number of features selected. In this sense, the most studied combination in the literature is error rate and NFS, present in 53% of the research. Figure 12 shows the number of papers per year that have pursued the three most studied weighted multi-objective functions in the collected literature.

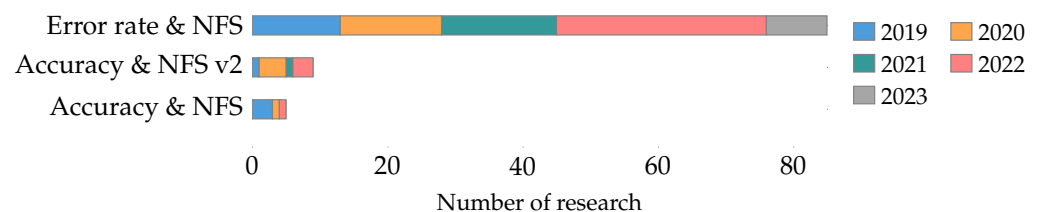


Figure 12. Weighted multi-objective functions by year.

4.2. What Metrics Are Used to Analyze the Performance of the Feature Selection Problem?

To facilitate the understanding of the different metrics found in the literature, the collected papers were grouped into four categories according to the metrics used. Figure 13 shows the proposed classification for the metrics found.

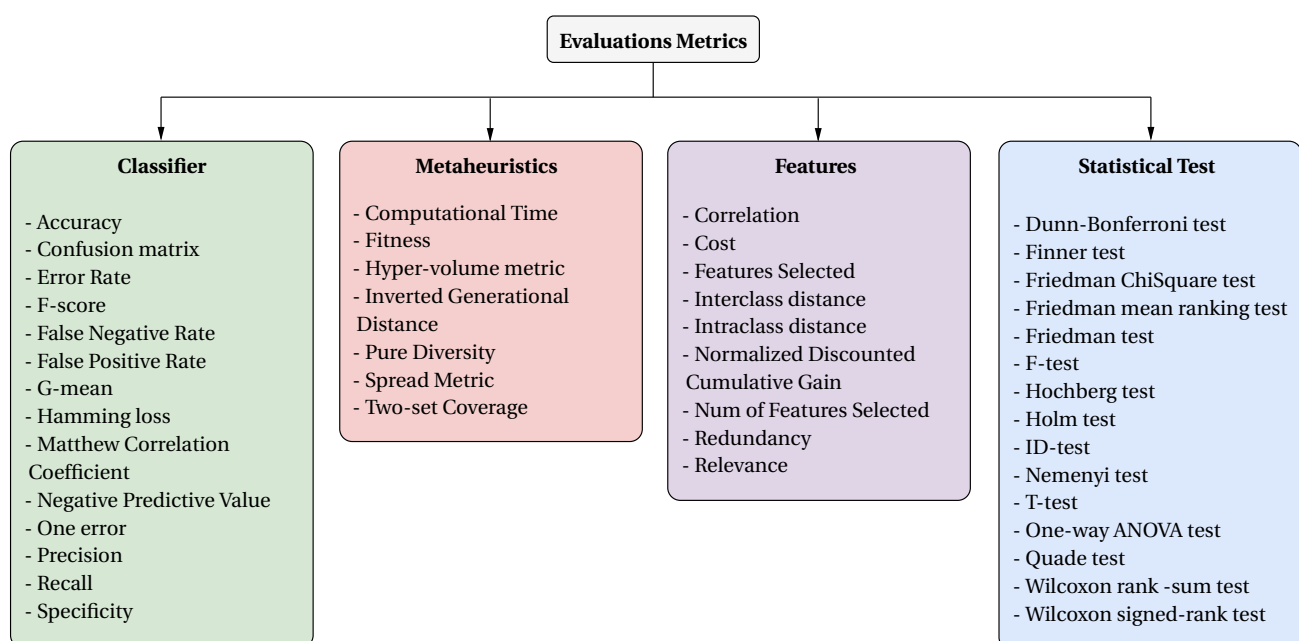


Figure 13. Taxonomy of evaluation metrics.

Classifiers are the most used metric in the collected literature, followed by metaheuristic metrics, with statistical tests being the least used. Figure 14 shows the number of papers that have used the different metric classifications in general (left) and their trends over the years (right).

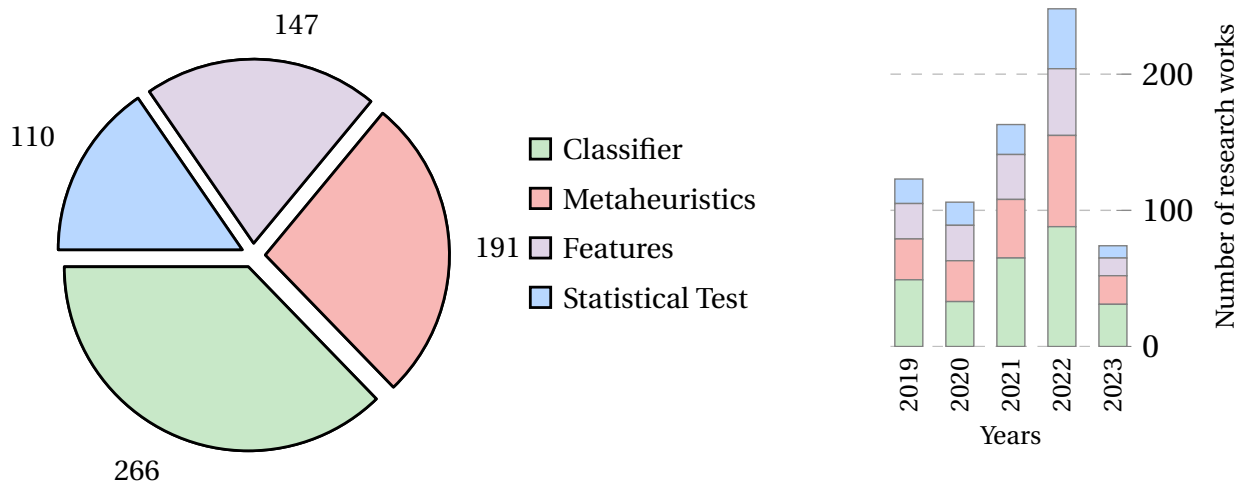


Figure 14. Evaluation metrics by category.

4.2.1. Classifier Metrics

The classifier metrics measure the performance of the machine learning technique using the subset of features obtained from solving the feature selection problem. In the literature, we detected 14 metrics of the classifier.

(a) **Accuracy:** Measures how well a classification algorithm correctly predicts the classes of the datasets. Accuracy is calculated as the ratio of correctly predicted cases to the total number of cases in the dataset. This metric has been used in [21,22,27–48,50,51,53–60,65,69,70,72,74,75,77,78,80–92,94–99,101–119,121,124–133,135–139,141–154,156,158–162,164–174,176,178–186] and mathematically is defined as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (22)$$

where TP (true positive) is the number of positive instances that are correctly classified, FP (false positive) is the number of negative instances that are wrongly classified as positive, TN (true negative) is the number of negative instances that are correctly classified, and FN (false negative) is the number of positive instances that are wrongly classified as negative.

(b) **F-score or f1-score or f-measure:** Used when dealing with imbalanced datasets. It combines precision and recall into a single value and is particularly useful for balancing these metrics. This metric has been used in [22,33,35,39,45,48,51,55,59,60,83,92,98,101,117,132,141,149,155,162,164,167,171,176–178,182,184,185] and mathematically is defined as follows:

$$f - score = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall} \quad (23)$$

where recall and precision are defined in Equation (24) and Equation (25), respectively.

(c) **Recall or sensitivity or true positive rate:** Ratio of true positive predictions to the total number of actual positive instances in the dataset. Measures the model's ability to correctly identify all positive instances. This metric has been used in [22,33,35,55,56,60,70,85,96,98,101,102,106,117,129,132,133,141,149,162,164,171,176–178,185] and mathematically is defined as follows:

$$Recall = \frac{TP}{TP + FN} \quad (24)$$

(d) **Precision or positive predictive value:** Ratio of true positive predictions to the total number of positive predictions made by the model. Measures the accuracy of the

model when it predicts a positive class. This metric has been used in [22,33,35,45,48,51,55,59,74,83,98,101,106,117,129,132,141,149,164,171,176–178,185] and mathematically is defined as follows:

$$Precision = \frac{TP}{TP + FP} \quad (25)$$

(e) **Error rate**: Sometimes referred to as the misclassification rate, it is an essential complement to accuracy. Measures the overall accuracy of a model in terms of the proportion of misclassified instances in a dataset. This metric has been used in [46,49,59,62,64–68,70,71,93,120,122,131,133,134,140,150,179] and mathematically is defined as follows:

$$Error\ rate = \frac{FP + FN}{FP + FN + TP + TN} \text{ or } 1 - Accuracy \quad (26)$$

(f) **Specificity or true negative rate**: Evaluates the ability of the model to correctly identify negative instances (true negatives) out of all the actual negative instances in a dataset. It is essential when the cost of misclassifying a negative instance as positive is high or when seeking to emphasize the ability of the model to classify negative cases correctly. This metric has been used in [33,44,70,85,96,102,129,133,162,176,185] and mathematically is defined as follows:

$$Specificity = \frac{TN}{TN + FP} \quad (27)$$

(g) **Matthew correlation coefficient (MCC)**: Assesses the quality of binary and multi-class classifications, especially when dealing with imbalanced datasets. MCC provides a balanced measure of the ability of the model to discriminate between positive and negative instances. This metric has been used in [45,51,181] and mathematically is defined as follows:

$$MCC = \frac{TN \cdot TP - FN \cdot FP}{\sqrt{(TP + FP) \cdot (TP + FN) \cdot (TN + FP) \cdot (TN + FN)}} \quad (28)$$

MCC ranges from -1 to 1 , where an MCC = 1 indicates a perfect classification, an MCC = 0 suggests random classification, and an MCC = -1 indicates a complete disagreement between predictions and actual values.

(h) **False positive rate (FPR)**: Measures the rate at which the model incorrectly classifies negative instances as positive. It is an important metric where the cost of false positives is significant. This metric has been used in [33,58,98] and mathematically is defined as follows:

$$FPR = \frac{FP}{FP + TN} \quad (29)$$

(i) **False negative rate (FNR)**: Measures the rate at which the model incorrectly classifies positive instances as negative. It is essential when failing to detect positive instances has significant consequences. This metric has been used in [33,58] and mathematically is defined as follows:

$$FNR = \frac{FN}{TP + FN} \quad (30)$$

(j) **Hamming loss**: Evaluates the performance of multi-label classification problems. In multi-label classification, each instance can be associated with multiple class labels, and the goal is to predict all the correct labels for each instance. The Hamming loss quantifies how well the model performs in terms of correctly predicting all the labels for each instance. This metric has been used in [41,63] and mathematically is defined as follows:

$$Hamming\ Loss = \frac{1}{p} \sum_{i=1}^p \frac{1}{q} |h(x_i) \triangle Y_i| \quad (31)$$

where p and q indicate the number of samples and labels. $h(x_i)$ represents the result of classification for the i -th sample and Y_i shows the actual labels of the corresponding

sample. $h(x_i)$ and Y_i are binary vectors in which 1's indicate the class labels to which the instances belong. Δ represents the Hamming distance between $h(x_i)$ and Y_i .

(k) One error: Evaluates how often the model makes exactly one error when predicting the set of labels for an instance. This metric has been used in [41,63] and mathematically this metric is defined as follows:

$$One\ Error = \frac{1}{|T|} \sum_{i=1}^{|T|} [\arg \max_{l_k \in L} g_k(w_i) \notin \lambda_i] \quad (32)$$

Let $T = \{(w_i, \lambda_i) | 1 \leq i \leq |T|\}$ be a given test set, where $\lambda_i \subseteq L$ is a correct label subset that is associated with a pattern w_i . Here, L represents the set of all possible labels in the multi-label classification task. Given a test pattern w_i and a multi-label classifier estimate a predicted label set $Y_i \subseteq L$. Specifically, a series of functions $\{g_1, g_2, \dots, g_{|L|}\}$ is induced from the training patterns. Next, each function g_k determines the class membership of l_k with respect to each pattern, i.e., $Y_i = \{l_k | g_k(w_i) > \theta, 1 \leq k \leq |L|\}$, where θ is a predetermined threshold, such as 0.5.

(l) Confusion matrix: Provides a comprehensive and detailed model performance summary. It is particularly useful for evaluating the quality of predictions, understanding the types of errors a model makes, and assessing its strengths and weaknesses. This metric has been used in [149,185] and is defined as a square matrix where each row represents the actual class labels, and each column represents the predicted class labels. Table 3 shows the confusion matrix of a binary classification.

Table 3. Confusion matrix.

	Predicted Negative	Predicted Positive
Actual negative	TN	FP
Actual positive	FN	TP

(m) G-mean or geometric mean: Calculated as the geometric mean of sensitivity and specificity. Sensitivity measures the model's ability to correctly identify positive instances, while specificity measures its ability to identify negative instances correctly. This metric has been used in [70] and mathematically is defined as follows:

$$G - mean = \sqrt{Sensitivity \cdot Specificity} \quad (33)$$

where sensitivity and specificity are defined in Equations (24) and (27).

(n) Negative predictive value (NPV): Used mainly in medical and diagnostic applications, it assesses the ability of a model to identify correctly negative instances among those it predicts as negative. This metric has been used in [162] and mathematically is defined as follows:

$$NPV = \frac{TN}{TN + FN} \quad (34)$$

Accuracy is the most used classifier metric in the literature, presented in 85% of the collected research. On the other hand, the rest of the classifier metrics are studied in less than 25% of the research. Figure 15 shows the classifier metrics studied by year.

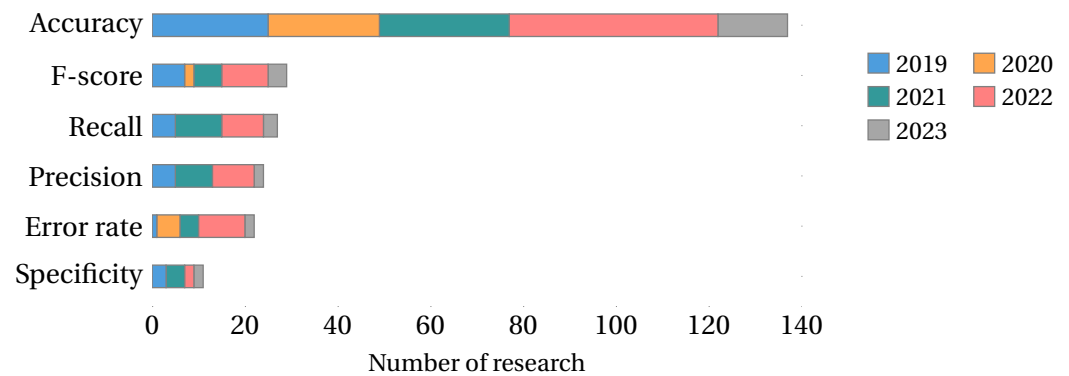


Figure 15. Classifier metrics by year.

4.2.2. Metaheuristic Metrics

The metaheuristic metrics measure the performance of the metaheuristic when solving the feature selection problem. Within this classification, seven different metaheuristic metrics were detected.

(a) **Fitness:** Corresponds to the evaluation of the objective function of the solution obtained. The authors report the best and worst solutions obtained, the average of the different executions performed, and their standard deviation. This metric has been used in [21,31,33,46,48,50,53,54,57,60,78–82,84,85,87–99,101–103,105–108,110,112,114,115,118–140,142–149,151–155,157–161,163–167,172,175,177,179,183,185]

(b) **Computational time (CT):** Corresponds to the computational times used by the algorithms to solve the feature selection problem. It is important to note that the authors report the CPU, operating system, RAM capacity, and programming language used in the experimental environment. The authors report the times in seconds or minutes and present the average obtained in the different executions performed as their standard deviation. This metric has been used in [21,22,34,36,38,39,44,48,49,51,53,55,65,67,69,71,72,75,77,80–87,89,91,98,105–109,113–116,119,125,126,128,132,134,138–140,142–148,157,158,160,163,165–172,176,178,179,181,182,184,186]

(c) **Hyper-volume (HV):** Used for pure multi-objective optimization problems and measures the hyper-volume of a region formed by the solution set and a reference point. The reference point usually is the anti-optimal point or “worst possible” point in the objective space. This metric has been used in [62–65,67–70,73,75,120,179]

(d) **Inverted generational distance (IGD):** Computes the average Euclidean distance from true Pareto fronts to its closest solution in the population. This metric has been used in [64,65,68,73,75] and mathematically is defined as follows:

$$IGD(S) = \frac{\sum_{x^* \in F^*} dist(x^*, S)}{|F^*|} \quad (35)$$

where S is the current Pareto front set, $dist(x^*, S)$ corresponds to the Euclidean distance between a point $x^* \in F^*$ and its nearest solution S .

(e) **Two-set coverage (SC):** Used for pure multi-objective optimization problems and employed to compare the convergence degree of two algorithms. This metric has been used in [62,63,70] and mathematically is defined as follows:

$$SC(A, B) = \frac{|\{b \in B | \exists a \in A : a \text{ dominates } b\}|}{|B|} \quad (36)$$

where A and B are two Pareto fronts obtained by two algorithms, and $SC(A, B)$ is defined as the percentage of solutions in B that are dominated by at least one solution in A .

(f) **Pure diversity (PD)**: Monitor the diversity of solutions during the optimization process. This metric has been used in [63,65] and mathematically is defined as follows:

$$PD(S) = \max_{s_i \in S} \{PD(S - s_i) + d(s_i, S - s_i)\} \quad (37)$$

where S is the current Pareto front, and $d(s_i, S - s_i)$ denotes the dissimilarity d from one solution S_i to the population S .

(g) **Spread**: Measures the extent of spread achieved among the obtained non-dominated solutions. This metric has been used in [65,179] and mathematically is defined as follows:

$$Spread(S) = \frac{d_f + d_l + \sum_{i=1}^{N-1} |d_i - \bar{d}|}{d_f + d_l + (N - 1)\bar{d}} \quad (38)$$

where d_i is the Euclidean distance between neighboring solutions on the obtained non-dominated solutions set and \bar{d} is the mean of all d_i . The parameters d_f and d_l are the Euclidean distances between the extreme and boundary solutions of the obtained non-dominated set. N is the number of non-dominated solutions found so far.

Fitness and computational time are the most used metaheuristic metrics in the literature, present in 58% and 46% of the collected papers. The rest of the metaheuristic metrics are studied in less than 8% of the literature. Figure 16 shows the metaheuristic metrics studies by year.

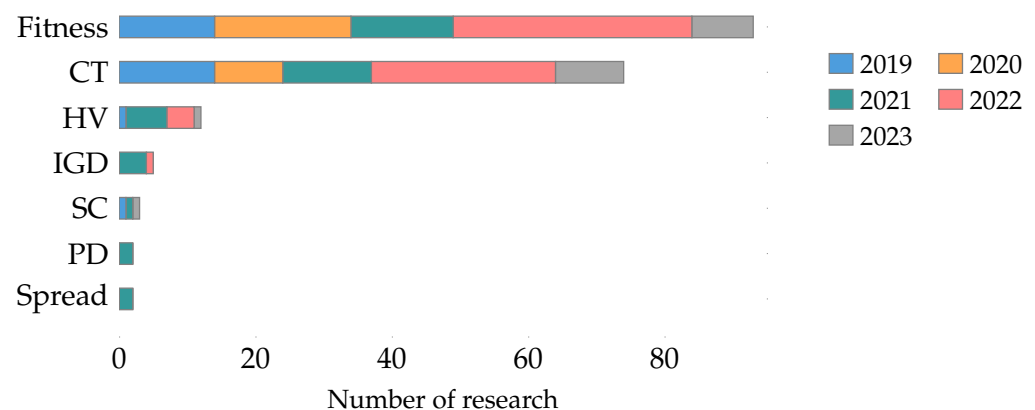


Figure 16. Metaheuristic metrics by year.

4.2.3. Feature Metrics

Feature metrics measure the attributes of the features that comprise the subset of selected features. Within this classification, nine different feature metrics were detected.

(a) **Number of features selected (NFS)**: Corresponds to the number of features that make up the best subset of features. The authors report the average number of features selected in the performed runs and their standard deviation. This metric has been used in [21,27,28,32,34,36–38,40,42,44–51,53–55,57,58,63–72,75,77,78,81–99,101–122,124–140,142–146,149,151–153,155–174,176,177,182–187]

(b) **Feature selected (FS)**: Corresponds to the identification of the selected characteristics. According to what has been detected in the literature, it is not only enough to indicate how many features were selected, but it is also essential to indicate which were selected. This metric has been used in [29,33,53,58,62,81]

(c) **Cost**: Considers the costs associated with the features. This cost can be either the cost of procurement or processing. This metric has been used in [62]

(d) **Normalized discounted cumulative gain (NDCG)**: Measures the effectiveness of a ranking algorithm by considering the relevance of the items ranked and their positions in the list. NDCG is an extension of the discounted cumulative gain metric, normalized to

provide a score between 0 and 1, making comparing different rankings or recommendation systems easier. This metric has been used in [74]

(e) **Correlation**: Used in multi-class classification and seeks to determine the correlation between features and classes. This metric has been used in [63] and mathematically is defined as follows: Given a set of features (F) and a set of labels (L), the correlation r_{fl} between the features $f \in F$ and the label $l \in L$ should be calculated using a predefined measure. The average feature-label correlation is required over all labels and all features. This average is calculated as follows:

$$\overline{r_{FL}} = \frac{1}{F \cdot L} \sum_{f \in F} \sum_{l \in L} r_{fl} \quad (39)$$

(f) **Relevance**: Represents the relevance between features and categorical variables and reflects the recognition ability of the selected features. This metric has been used in [75].

(g) **Redundancy**: Quantifies the level of similarity between selected features. This metric has been used in [75].

(h) **Interclass distance**: Represents the distance between the mean sample of each class and the average of the mean samples of all classes. This metric has been used in [75] and mathematically is calculated as follows:

$$Interclass\ distance = \sum_{i=1}^L \left(m_i - \frac{1}{L} \sum_{i=1}^L m_i \right)^2 \quad (40)$$

where L is the total number of classes and m_i is the average value of all samples with feature S in class i .

(i) **Intraclass distance**: Reflects the cohesion of the same type of samples. It is calculated by the distances between the samples with the selected characteristic and the average of all samples of the same type. This metric has been used in [75] and mathematically is calculated as follows:

$$Intraclass\ distance = \sum_{i=1}^L \sum_{a_{ij} \in L_i} (a_{ij} - m_i)^2 \quad (41)$$

where a_{ij} is the j -th sample in class i .

The number of features selected (NFS) is the most used feature metric in the literature, present in 83% of the collected papers. Features selected (FS) is the other metric used in more than one study. The rest of the feature metrics are studied in only one study. Figure 17 shows the feature metrics studied by year.

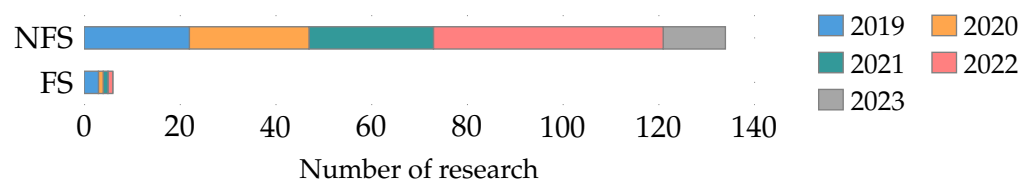


Figure 17. Feature metrics by year.

4.2.4. Statistical Tests

To demonstrate that one proposal is better than another, improvements must be shown in a particular metric, and a statistical test must be applied. Within this classification, 15 different statistical tests were detected.

(a) **Non-parametric statistical test**: Used to make inferences about data when the assumptions of parametric tests are not met. To apply a non-parametric statistical test, (1) the data do not follow a normal distribution; (2) the data are measured on an ordinal

or nominal scale rather than on a continuous scale; (3) the assumption of homogeneity of variances are violated; and (4) the sample size is small, making it difficult to rely on the central limit theorem to approximate a normal distribution. In this systematic review, six non-parametric statistical tests were detected. (i) **Wilcoxon signed-rank test or Wilcoxon test** [29,36,48,50,59,78,79,85,88,90,91,95,99,108,110,115,119,121,124–126,130,131,136–138,142,148,151,152,159,165,169,170], (ii) **Friedman test** [39,57,59,75,77,94,96,107,108,120,128,132,136,140,142,154,161,179,184], (iii) **Wilcoxon rank-sum test or Mann–Whitney U test** [30,38,49,72,82,86,94,97,98,117,129,139,145,157,160,163,164,167,173,176], (iv) **Friedman mean ranking test** [112,133,146,179,180], (v) **Friedman chi-square test** [128,137] and (vi) **Iman–Davenport test** [27].

(b) **Parametric statistical test** : Used to make specific assumptions about the underlying probability distribution of the analyzed data. To apply a parametric statistical test, (i) assume that the data follow a specific probability distribution, often the normal distribution; (ii) the data under analysis are continuous; and (iii) homogeneity of variances. In this systematic review, five parametric statistical tests were detected. (i) **T-test** [28,39,41,62,70,80,83,105,111,113,127,172,177], (ii) **F-test** [27,138], (iii) **one-way ANOVA test** [101], (iv) **Quade test** [179], and (v) **Finner test** [39].

(c) **Post hoc analysis**: Statistical procedure after an initial statistical analysis. The term “post hoc” is Latin for “after this.” In the context of statistics, it refers to conducting additional tests or comparisons after the primary analysis to investigate and understand the results further. In this systematic review, four post hoc analyses were detected: (i) **Nemenyi test** [27,111,128], (ii) **Hochberg test** [57,107,161], (iii) **Holm test** [57,107], and (iv) **Bonferroni–Dunn test** [41,59].

The Wilcoxon test is the most used in the literature, presented in 21% of the collected papers, followed by the Friedman test and Wilcoxon rank-sum test, each studied in 12% of the research. The rest of the statistical tests have been used in less than 10% of the literature. Figure 18 shows the statistical tests studied by year for the tests used in at least three investigations.

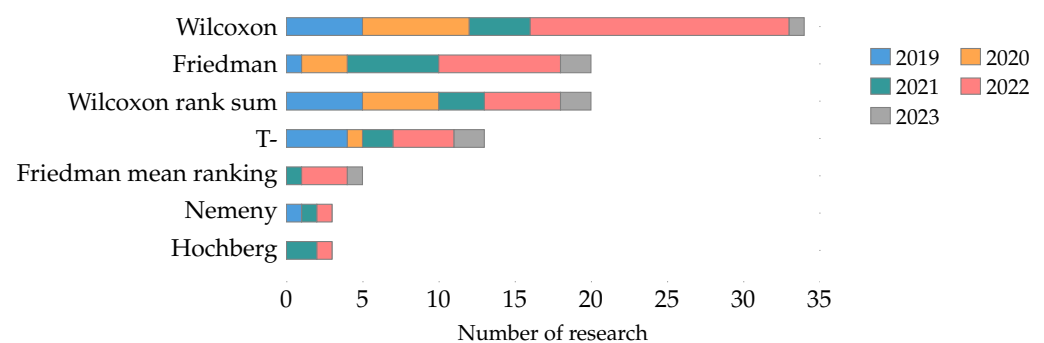


Figure 18. Most used statistical tests by year.

4.3. What Machine Learning Techniques Have Been Used to Calculate Fitness in the Feature Selection Problem?

In machine learning, classifiers are relevant for data analysis, pattern recognition, and decision-making [188–192]. In order to determine which classifiers are employed in this context, the collected literature was analyzed, allowing us to improve the understanding of the evolution of classifier usage, particularly in evaluating the effectiveness of selected features in various models. The review of these articles uncovers trends, distributions, and characteristics of classifiers as evaluative tools in the feature selection process. Comprehending these aspects is essential to grasp how classifiers enhance the efficiency and accuracy of machine learning models. By examining the prevalence and variations in classifier deployment, we gain insights into how researchers innovatively use these algorithms in complex problems, explicitly selecting and validating features across diverse datasets. The

following sections provide a detailed analysis of these trends, elucidating usage patterns and classifier categories.

4.3.1. Classifier Trends over Time

Researchers have increasingly deployed classifiers over the past five years in feature selection and optimization tasks. This trend is evident in the rising number of articles published on the topic and the average number of classifiers used per article, as shown in Figure 19. This figure presents two sets of data: the number of articles published per year (represented as a bar chart) and the average number of classifiers used per article (depicted as a line graph). It is important to note that these two data series are represented on different scales to ensure clarity and legibility. While the bar chart reflects the total number of articles analyzed (152 out of 161, excluding 9 articles where the classifier was not specified), the line graph for the average number of classifiers is plotted on a separate scale. This approach was adopted to prevent the average number of classifiers from appearing too close to the zero line, thereby preserving its visibility and explanatory power in the figure. The dual-scale representation, while creating a visual impression of disproportionality, is essential for an accurate and clear depiction of the trends.

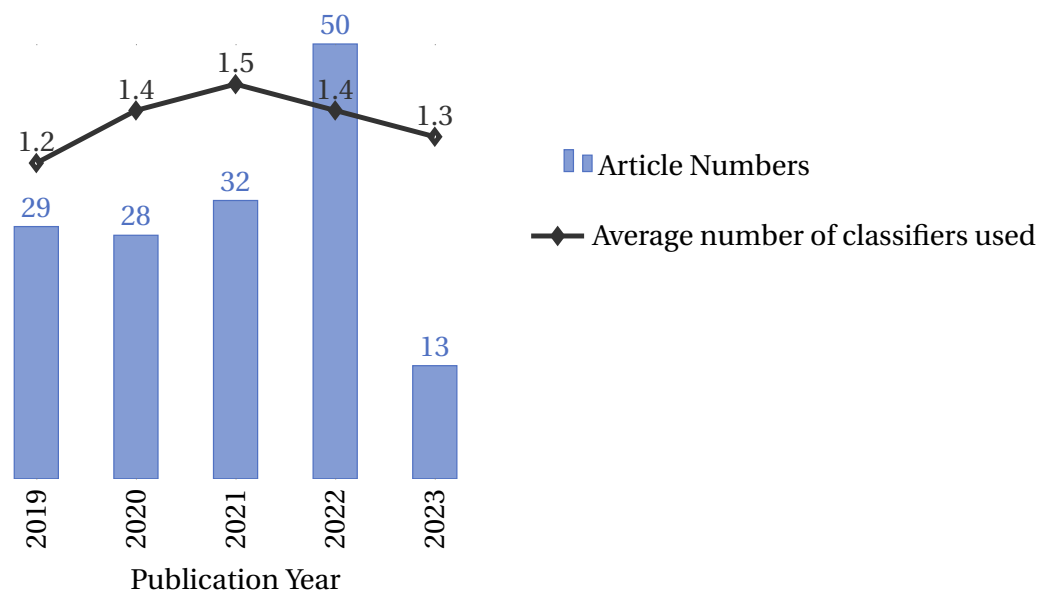


Figure 19. Trend in the average number of classifiers used per article.

This trend is likely due to the increasing popularity of machine learning and the growing availability of data. Classifiers are a powerful tool for extracting insights from data and can be used to improve the performance of feature selection and optimization algorithms. The trend towards using more classifiers per article is also noteworthy. This suggests that researchers are increasingly experimenting with different classifier types and combinations to achieve better results.

Examining these trends, we discern several patterns. Over the years, the increasing number of articles suggests the growing significance of feature selection and optimization in the research landscape. Notably, the slight fluctuations in the average number of classifiers per article indicate the adaptability of the field, where researchers balance the quest for precision with practical considerations.

In this context, there are a particular set of highlight articles that, although not explicitly detailing the classifiers used, made substantial contributions to the field, directing their focus toward innovative methodologies and applications. For instance, Chaudhuri et al. [110] made strides in feature selection using the binary crow search algorithm with time-varying flight length. Long et al. [180] investigated numerical optimization and feature selection through a butterfly-balanced optimization algorithm. In similar veins,

the studies in Takieldeem et al. [122] and Kalra et al. [123] introduced the dipper-throated optimization algorithm and a novel binary emperor penguin optimizer, respectively, both serving feature selection tasks. Further contributions came from Tubishat et al. [50], who delved into dynamic generalized normal distribution optimization for feature selection, and Li et al. [187] designed a two-stage hybrid feature selection algorithm with applications in Chinese medicine. Oyelade et al. [144] explored evolutionary binary feature selection using an adaptive Ebola optimization search algorithm tailored for high-dimensional datasets. Meanwhile, a hybrid global optimization algorithm for feature selection was meticulously examined in [186], and a dynamic butterfly optimization algorithm for feature selection was showcased in [151].

While the classifiers in these studies may not be distinctly outlined, the sheer breadth of approaches and applications in these works underscores the diversity and innovation pulsating through feature selection and optimization, offering a rich tapestry of knowledge and avenues for future exploration.

4.3.2. Classifier Usage by Year

Examining the evolution of classifier usage across different years can provide valuable insights into the dynamic landscape of feature selection and optimization. Figure 20 shows the annual distribution of papers based on the number of classifiers employed. This chart provides a comprehensive overview of classifier usage across the years, categorized by the number of classifiers employed in each paper.

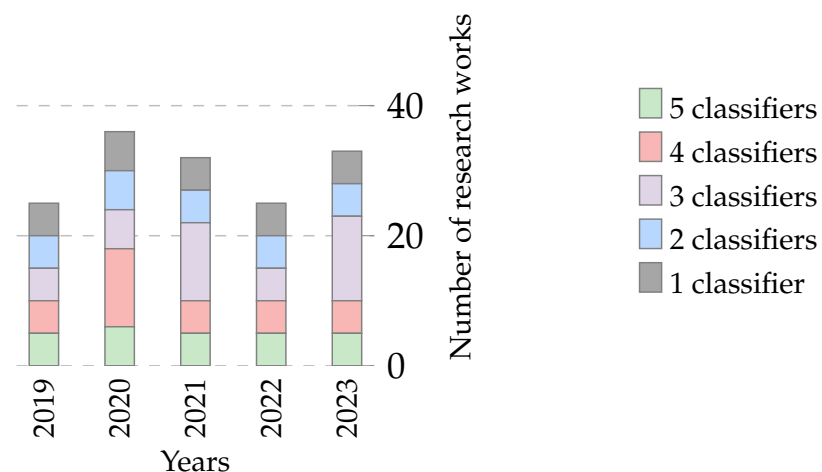


Figure 20. Number of classifiers by year.

In this context we notice the following patterns and shifts in classifier usage over the years:

- **2019:** The majority of articles (86.2%) employed a single classifier [27–29,42,44,45,51,54,62,77–88,155,163,164,166], setting the tone for a strong emphasis on foundational methodologies.
- **2020:** The trend continues, with the prominence of single classifiers remaining steady at 78.6% [23,31,41,46,53,71,72,89,90,92,93,95,152–154,156–159,165,168,173]. A slight increase in articles employing two classifiers suggests a nascent exploration of combinations [91,167,175].
- **2021:** The year sees an expanded adoption of multiple classifiers, with a noticeable uptick in papers employing three (12.5%) [35,56,107,176] and four (3.1%) classifiers [106]. This potentially signifies a growing confidence in ensemble methodologies.
- **2022:** A significant leap is observed in the total number of articles, accompanied by a proportional increase in the use of diverse classifiers. The rise in articles employing multiple classifiers, including five classifiers [117], underscores a dynamic approach to optimization challenges.

- **2023:** The number of articles decreases, and the distribution reverts to a focus on single classifiers [40,70,143,145,146,148–150,160,184,185], while a minimal presence of two and four classifiers persists.

The trends in classifier usage suggest that researchers are moving from foundational exploration to embracing more complex and multifaceted deployments. The increasing adoption of ensemble methodologies aligns with the field’s maturation and the need to comprehensively address complex optimization objectives.

4.3.3. Classifier Descriptions

To make informed decisions about feature selection and optimization, it is essential to understand the nuances of different classifiers. This subsection summarizes the purposes and fundamental characteristics of each classifier. Table 4 presents a concise delineation of each classifier’s intent and attributes.

Table 4. Classifier descriptions.

Classifier	Description
Adaptive Boosting (ADABOOST)	Ensemble technique adjusting weights on misclassified instances for improved accuracy.
Artificial Neural Network (ANN)	Model inspired by the brain, with interconnected neurons for data processing.
Decision Tree (DT)	Divides data into branches by evaluating feature values, arriving at decisions at each internal node, and assigning class labels to leaf nodes.
Decision Tree C4.5 (DT C4.5)	Refined algorithm dividing data based on features, selecting attributes via info gain, handling varied types, missing values, and pruning.
Decision Tree J48 (DT J48)	Improved C4.5 in Weka, selects attributes with info gain, handles varied attributes, missing data, and pruning.
Discriminant Analysis (DA)	Technique finding linear combinations of features for class separation and dimensionality reduction.
Extreme Gradient Boosting (XGBOOST)	Boosting algorithm that builds strong learners by focusing on instances with poor previous learner performance.
Extreme Learning Machine (ELM)	Single-hidden-layer neural network that randomly assigns weights and determines output weights analytically.
Fuzzy Classifier (FC)	Classifier using fuzzy logic to handle uncertainty in data.
Fuzzy Min–Max Neuronal Network (FMM)	Fuzzy system for classification, handling uncertainty using membership functions.
Gaussian Naive Bayes (GNB)	Naive Bayes variation assuming Gaussian distribution of feature values.
Growing Hierarchical Self-Organizing Map (GHSOM)	Neural-network-based algorithm for clustering and visualization of high-dimensional data.
K-Nearest Neighbor (k-NN)	Assigns labels based on the majority class of k nearest neighbors.
Kernel Extreme Learning Machine (KELM)	ELM variant using kernel methods for nonlinear classification in high-dimensional space.
Kstar Classifier (KSTAR)	Lazy learning algorithm classifying new instances based on closest neighbors.
Latent Dirichlet Allocation (LDA)	Generative model used for topic modeling in text data, revealing hidden topic structures.
Light Gradient Boosting (LightGBM)	Gradient boosting with histogram-based training for efficiency and accuracy.
Logistic Model Tree (LMT)	Decision tree with leaf nodes containing logistic regression models.
Logistic Regression (LR)	Linear model estimating the probability of binary classification.
Multi-Label KNN (ML-KNN)	Extends k-NN for multi-label classification, allowing instances to have multiple labels.
Multi-Label Naive Bayes (MLNB)	Naive Bayes extension for multi-label classification problems.

Table 4. Cont.

Classifier	Description
Multilayer Perceptron (MLP)	Neural network with multiple layers for complex nonlinear mappings.
Naive Bayes (NB)	Probabilistic classifier based on Bayes’ theorem, assuming feature independence.
Oblique Random Forest Heterogeneous (OblRF(H))	Variant of random forest using oblique splits for decision trees.
Optimum-Path Forest (OPF)	Pattern recognition algorithm constructing decision boundaries through graph-based approach.
Random Forest (RF)	Ensemble classifier that combines multiple decision trees to improve accuracy.
Standard Voting Classifier (SVC)	Ensemble technique combining classifier predictions through majority voting.
Support Vector Machine (SVM)	Finds a hyperplane to separate classes, maximizing the margin between them.

From the robust support vector machine (SVM) that finds optimal separating hyperplanes, to the intricate multilayer perceptron (MLP) that captures complex nonlinear relationships, each classifier serves a unique role. Ensembles like random forest (RF) and extreme gradient boosting (XGBOOST) showcase the power of collective learning, while naive Bayes (NB) relies on probabilistic reasoning for classification. Decision trees, represented by Decision Tree C4.5 (DT C4.5) and its variants, offer interpretability.

These classifier descriptions provide a quick reference guide for comprehending the diverse methodologies related to feature selection and optimization.

4.3.4. Most Common Classifiers

A diverse range of classifiers are used in feature selection and optimization to tackle complex challenges. Figure 21 shows the prevalence of specific classifiers in the reviewed articles.

At the forefront of the classifier ensemble, the k-nearest neighbor (k-NN) emerges as the most frequently employed technique, featured in a substantial 77% of the papers collected [27–30,32,35–39,42,43,45–49,51,53–57,60,62,64–70,77–93,95–97,99–109,112–121,124–143,145–149,152–161,163,165–168,172–174,177–179,181]. This exemplifies its role as a foundational and versatile approach in addressing complex optimization tasks. Notably, the support vector machine (SVM) follows, with a notable presence in 17.4% of papers [21,32,35,39,40,43,55–57,59,60,72,74,91,106,107,117,147,164,169,171,174–178,181,182]. The presence of naive Bayes (NB) [33,35,60,94,106,172,176–178,181] and various decision tree classifiers, including Decision Tree Classifier C4.5 (DT C4.5)[30,94] and random forest (RF) [22,34,147,162,167,176], underscores the ongoing significance of interpretable and ensemble-based methodologies. Meanwhile, emerging techniques like Extreme Gradient Boosting (XGBOOST) [22,33,56,117] and Artificial Neural Network (ANN) [31,71,115,175] reflect the integration of modern learning paradigms to address complex optimization endeavors.

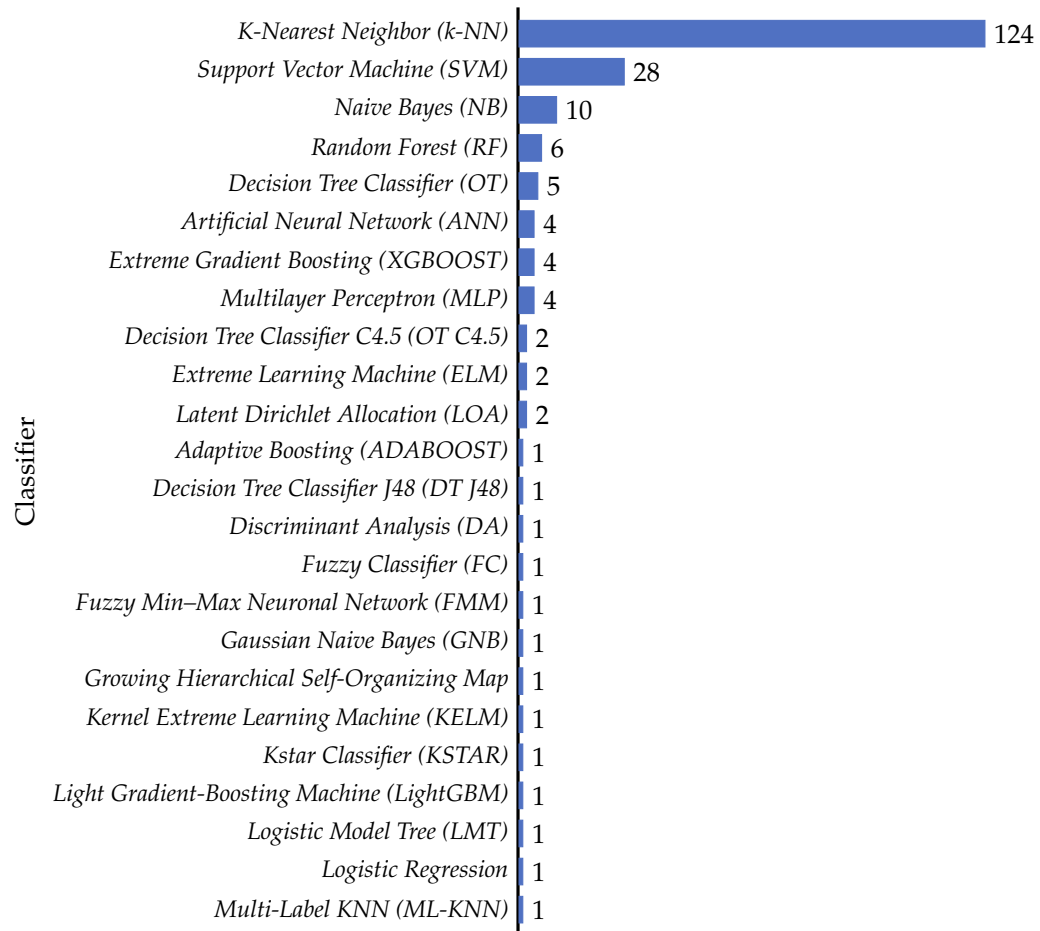


Figure 21. Classifier distribution analysis: count of articles per classifier.

4.3.5. Classifier Categories

This research employed various methodologies, highlighting the range of approaches within the field. Figure 22 enriches our comprehension of classifier diversity and highlights key focus areas.

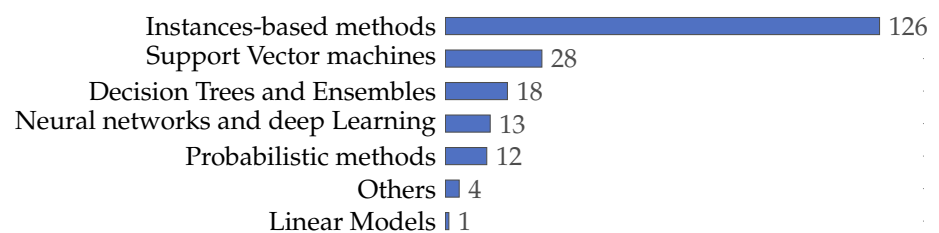


Figure 22. Research by classifier.

Dominating the field are instance-based methods [23,27–30,32,35–39,42,43,45–49,51,53–57,60,62–70,77–93,95–97,99–109,112–121,124–143,145–149,152–163,165–168,172–174,177–179,181–184] employing algorithms such as k-nearest neighbor (k-NN) and its multi-label variant (ML-kNN). This category stands out for its intuitive logic, classifying new instances by a majority vote among the nearest neighbors, thus encapsulating a local approximation of the target function.

The support vector machines category [21,32,35,39,40,43,55–57,59,60,72,74,91,106,107,117,147,164,169,171,174–178,181,182] is characterized by its foundational SVM algorithm. This method seeks the optimal separating hyperplane in a transformed feature space. It

is recognized for its prowess in high-dimensional settings, making it a powerful tool for binary and multi-class classification problems.

Decision trees and ensembles [22,30,33,34,56,59,65,94,107,111,117,138,147,162,167,176,181,182] represent a collective of methodologies like the decision tree classifier (including variations such as J48 and C4.5), random forest, and adaptive boosting (AdaBoost). These models are particularly noted for their interpretability and the ensemble strategies that aggregate the predictions of multiple trees to enhance performance and mitigate overfitting.

The neural networks and deep learning approaches [31,44,58,71,75,106,115,117,150,170,174,175,185] encompass models such as artificial neural networks (ANNs), multilayer perceptrons (MLPs), and more advanced configurations like deep learning architectures. These methods simulate the complex interconnections of a biological brain and excel in capturing nonlinear relationships within large datasets.

The probabilistic methods [33,35,41,60,94,98,106,172,176–178,181] include naive Bayes (NB), Gaussian naive Bayes (GNB), and multi-label naive Bayes (MLNB). These algorithms are based on applying Bayes' theorem and are valued for their ability to handle uncertainty and deliver probabilistic predictions.

A category named “other classifiers/algorithms” includes methods not typically aligned with standard classifier frameworks previously described. Discriminant analysis (DA), employed to classify observations into predefined classes based on their features, is discussed in one study [32]. Fuzzy classifiers (FCs), which apply fuzzy logic to handle ambiguous class memberships, are utilized in another work [47]. Additionally, latent Dirichlet allocation (LDA) is used to model latent topics within text corpora, as demonstrated in two articles [73,174].

Lastly, linear models were mentioned in one article [94], with logistic regression being a primary example of this approach. It models the probability of a binary outcome and is often favored for scenarios where relationships between the input variables and the output are presumed to be linear.

4.4. What Metaheuristics Have Been Used to Solve the Feature Selection Problem?

Researchers have employed a diverse range of metaheuristics to address the feature selection problem. This section identifies and discusses the various metaheuristics featured in the reviewed articles. Key aspects covered include the frequency of metaheuristic usage, observed binarization approaches, hybridization of metaheuristics, techniques employed to enhance performance, the application of multi-objective approaches in metaheuristics, and the interplay between objective function formulation and metaheuristics.

4.4.1. Frequency of Source Metaheuristics Utilization

Metaheuristics are general-purpose algorithms that, with few modifications, can solve different optimization problems [193]. We refer to “**base**” or “**source**” metaheuristics as the main metaheuristics used by researchers that are later adapted. Figure 23 shows the metaheuristics that have emerged as the most frequently utilized for solving the feature selection problem.

Particle swarm optimization is a population metaheuristic based on the swarming behavior of animals such as birds or fish [194]. This metaheuristic is the most used by the authors in [22,27,28,30,32,35,36,38,39,45,54,65,69,77,79,80,83,87,100,127,138,153,155,161,168,174,186]. The second most used metaheuristic is the grey wolf optimizer, a population metaheuristic based on the hunting behavior of grey wolves [195] and used in [51,71,84,87,88,95,101,105,128,134,137,157,160,170,173]. The other metaheuristic used in more than ten studies is the genetic algorithm, a population metaheuristic inspired by Darwin's laws of evolution [196] and used in [31,32,41,56,58,63,72,73,79,98,100,135,175].

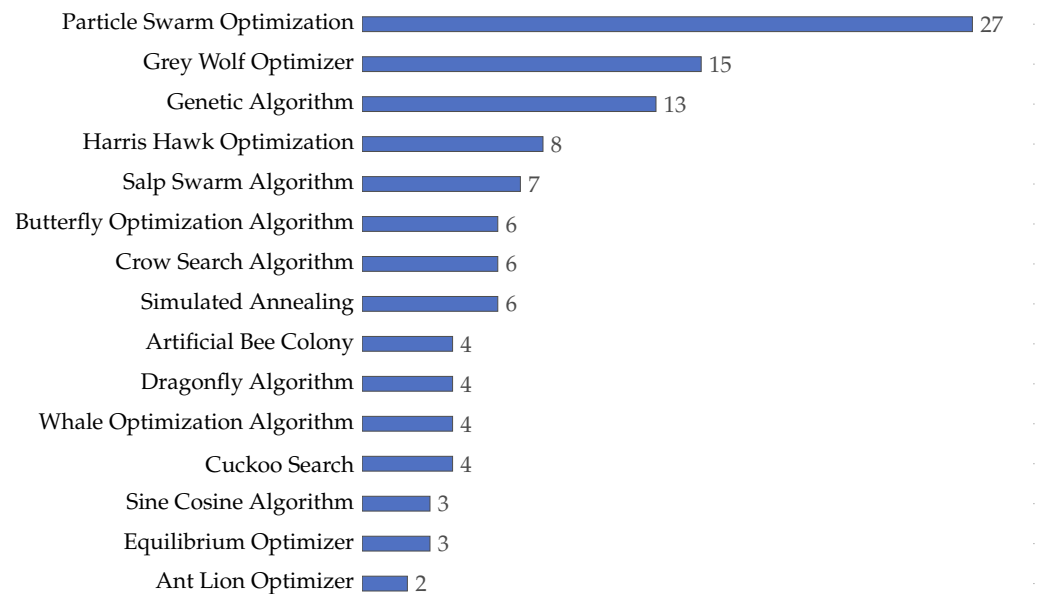


Figure 23. Research by metaheuristic.

It is noteworthy that while some metaheuristics like PSO, GWO, and GA have been extensively used, a vast array of other algorithms have been explored less frequently. This diversity suggests that the field of feature selection is rich and continuously evolving, with researchers experimenting with different algorithms to find the most suitable solution for specific problems.

The prominence of these metaheuristics also underscores the importance of continuous improvement and adaptation. The no free lunch theorem [18,19] inspires researchers to innovate. Given this, newer variants or hybrid versions of these algorithms are likely to emerge, further expanding the boundaries of feature selection research.

4.4.2. Binarization Approaches in Metaheuristics

Binarization in metaheuristics refers to converting continuous solutions into binary solutions, which is essential for problems like feature selection where the solution space is binary [20,197]. There are two primary approaches observed in the reviewed documents related to binarization: Binarization in metaheuristics is transforming continuous solutions into binary solutions [20,197], a necessary step for problems like feature selection where the solution space is binary. The reviewed literature indicates two primary binarization approaches:

- **Direct binarization:** This approach involves straightforward methods where the binarization process is direct and does not involve extensive testing or evaluation of different techniques. It is often used for its simplicity and efficiency. Cases of this approach are the papers [21,35,36,40,48,63,67,73,80,82,87,94,95,98,101,104,105,108,112,115,118,119,123,133,134,136,149,150,152,153,158,159,161,169,177].
- **Binarization with various approaches:** This approach involves a comprehensive study and evaluation of multiple binarization techniques to determine the most effective one for a given problem. It is more exhaustive and aims to find the optimal binarization method for specific scenarios. Cases of this approach are the articles [47,71,78,89,91,92,96,106,107,110,114,124,129,131,132,138,142,144,146,154,165,172,176,179].

Figure 24 shows the comparative trends over the past five years for these binarization approaches.

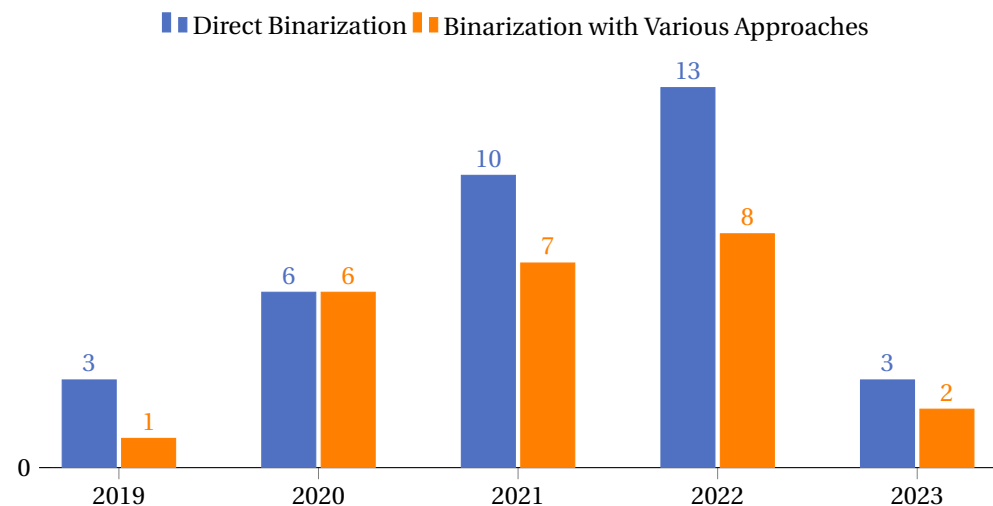


Figure 24. Direct Binarization vs. Binarization with various approaches by year.

Analyzing Figure 24, it is evident that direct binarization has significantly increased from 2019 to 2022. This could be attributed to its straightforward nature, making it a preferred choice for researchers who prioritize efficiency. On the other hand, binarization with various approaches has remained relatively consistent over the years, with a slight increase from 2019 to 2022. This indicates a steady interest in exploring and evaluating different binarization techniques to find the most effective one.

4.4.3. Hybridization in Metaheuristics

Hybridization in the context of metaheuristics refers to combining two or more metaheuristic algorithms to create a new, often more efficient, method. This subsection specifically focuses on hybridization, which involves merging one metaheuristic with another, excluding combinations with non-metaheuristic techniques. The primary objective behind such hybridization is to capitalize on the strengths of the individual metaheuristics while mitigating their weaknesses. By integrating the best features of multiple metaheuristics, these hybrid methods often achieve superior performance, faster convergence, and more robust solutions, especially in complex optimization problems like feature selection. Figure 25 shows the trend in metaheuristic hybridization over the past five years.

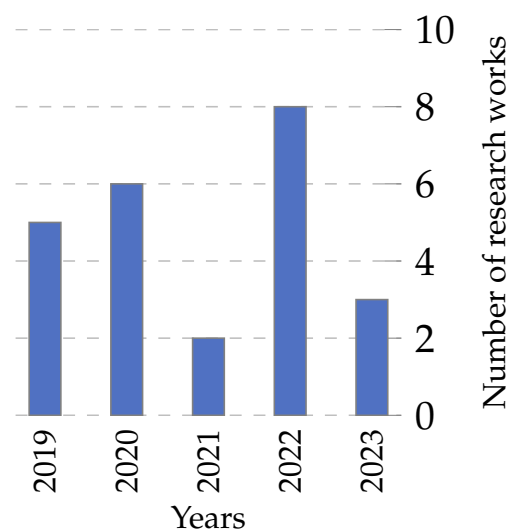


Figure 25. Metaheuristic hybridization.

The data show a noticeable fluctuation in the interest in metaheuristic hybridization over the years. While there was a steady increase from 2019 [85–88] to 2020 [72,92,154,157,159,173], there was a significant drop in 2021 [97,161]. However, 2022 [69,112,113,121,135–137,140] saw a resurgence in the number of articles focusing on hybridization, indicating a renewed interest or perhaps the emergence of new hybrid techniques that garnered attention. The decline in 2023 [70,144,145], similar to other trends, might be attributed to the data only covering up to April, and it is possible that the numbers might increase as the year progresses.

The fluctuating trend suggests that while hybridization remains a topic of interest, its application and exploration might be influenced by various factors, including the emergence of new standalone algorithms, the complexity of hybrid methods, or shifts in research focus. Nonetheless, the consistent presence of hybridization articles underscores its importance and potential in enhancing feature selection methodologies.

In the hybridization process involving two metaheuristics, one typically serves as the foundational or base algorithm, while the other acts as an enhancement. This enhancement specifically targets and strengthens aspects of the base metaheuristic that may be perceived as weaker than other metaheuristics. From 2019 to April 2023, a review of 24 articles revealed various metaheuristics employed as foundational or base algorithms in the hybridization process. These metaheuristics serve as the backbone upon which enhancements are made using other algorithms to address specific weaknesses or to leverage unique strengths. Figure 26 shows the various metaheuristics employed as foundational algorithms in the hybridization process from 2019 to April 2023.

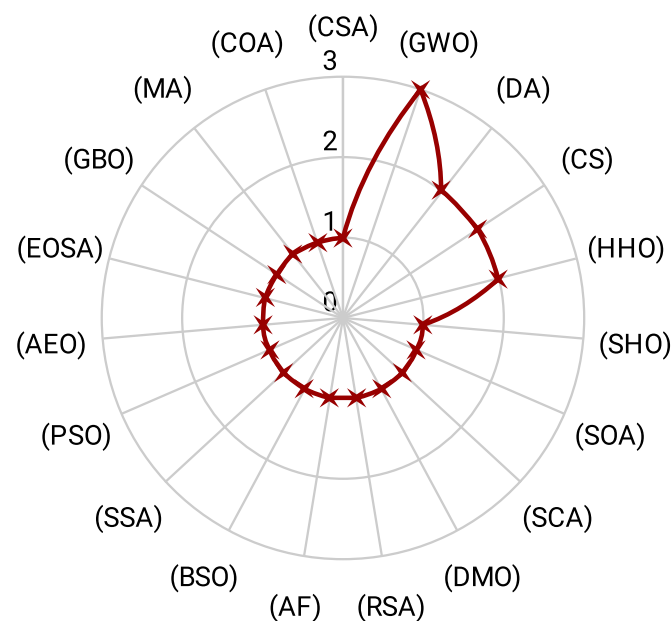


Figure 26. Frequency of foundational metaheuristics.

Analyzing Figure 26 we detect the following:

- The grey wolf optimizer (GWO) [87,88,157] stands out as the most frequently used foundational metaheuristic, having been employed as a base in three different studies. This suggests its prominence and potential adaptability in hybrid models.
- The dragonfly algorithm (DA) [97,155], cuckoo search (CS) [113,173], and harris hawk optimization (HHO) [140,159] have each been utilized twice. Their repeated use indicates their significance and robustness as foundational techniques in the hybridization process.
- Most of the metaheuristics, including but not limited to the spotted hyena optimization algorithm (SHO) [85], seagull optimization algorithm (SOA) [86], sine cosine algorithm (SCA) [92], and dwarf mongoose optimization (DMO) [112], have been used once as

foundational algorithms. This showcases the diversity of metaheuristics explored by researchers in the hybridization process.

- The wide range of foundational metaheuristics, even those used just once, underscores the richness of the field. It indicates that researchers continuously experiment with different base algorithms to find the most suitable combinations for specific problems.

Figure 27 visually represents the various metaheuristics employed to enhance or improve base algorithms. These enhancers are specifically used to address specific weaknesses in the base metaheuristics or to capitalize on unique strengths.

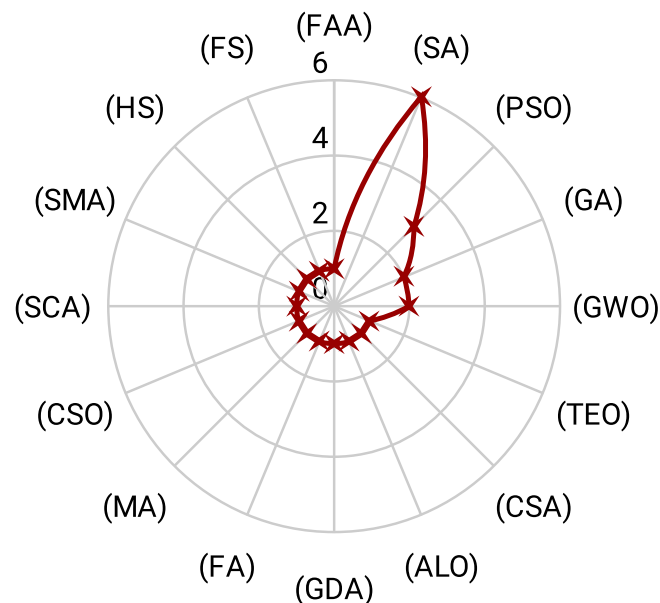


Figure 27. Frequency of metaheuristics enhancers.

Analyzing Figure 27, the following insights can be derived:

- Simulated annealing (SA) [85,97,112,121,144,159] emerges as the most frequently used metaheuristic for enhancement, with a count of six. Its recurrent use suggests it offers versatile capabilities in refining and optimizing base metaheuristics.
- Particle swarm optimization (PSO) [87,155,161] has been employed three times as an enhancer, indicating its adaptability and effectiveness in improving various foundational algorithms.
- The genetic algorithm (GA) [72,135] and Grey Wolf Optimizer (GWO) [137,173] have both been utilized twice as enhancing metaheuristics. Their repeated use underscores their potential in augmenting the performance of base algorithms.
- A wide array of metaheuristics, including the firefly algorithm (FAA) [144], thermal exchange optimization (TEO) [86], the cuckoo search algorithm (CSA) [88], and harmony search (HS) [154], among others, have been used once. This diversity reflects the rich experimentation in the field, with researchers exploring various combinations to achieve optimal results.

4.4.4. Techniques to Enhance Metaheuristics

Researchers have developed various techniques to enhance the performance of metaheuristics for feature selection. The most commonly used techniques include chaotic maps, local search, and fuzzy learning. These techniques are used to improve the exploration and exploitation capabilities of metaheuristics, which can lead to better solutions. It is important to note that these are just a few of the many techniques researchers have used to optimize metaheuristics. As the field of feature selection continues to evolve, researchers

are likely to develop even more innovative and effective techniques. Figure 28 illustrates the trend of techniques utilized over the analyzed time period.

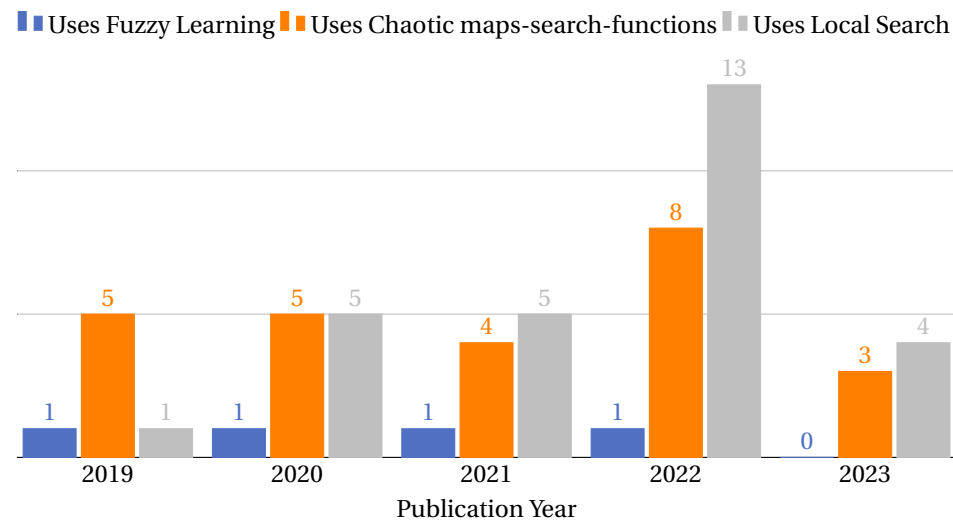


Figure 28. Enhancement techniques trend.

From Figure 28, several observations can be made:

- **Chaotic maps search function:** With a total of 25 instances across the years [34,40,42,53,81,90,93,103,106,114,117,120,121,126,129,134,143,147,156,159,161–164,183] this technique has seen consistent use, with a noticeable peak in 2022. Its application suggests that researchers find value in its chaotic dynamics to enhance the exploration capabilities of metaheuristics.
- **Local search:** This technique has been the most frequently employed, with a total of 28 instances [36,40,46,50,69,77,93,99,103,107,111–113,120,126–130,134,143,145,147,151,153,161,174,183]. Particularly in 2022, there was a significant surge in its application, indicating its effectiveness in refining solutions and improving convergence rates.
- **Fuzzy learning:** While this has been used less frequently, with only four instances over the years [36,44,47,53], it offers a unique approach to handling uncertainties and improving adaptability in metaheuristics.

In conclusion, while chaotic maps, local search, and fuzzy learning are among the more common techniques to enhance metaheuristics, their varied application over the years underscores the dynamic nature of research in this field. Researchers continuously experiment with different techniques, seeking the most effective combinations to address complex optimization challenges.

4.4.5. Multi-Objective Approaches in Metaheuristics

Multi-objective metaheuristics are specifically designed to tackle problems with multiple objectives. This is important because many real-world problems have multiple conflicting objectives, and single-objective metaheuristics cannot find optimal solutions for these problems [198].

Multi-objective metaheuristics aim to find solutions that balance and optimize all of the objectives simultaneously [198]. This is a challenging task, but it is essential in many real-world applications. Figure 29 shows the evolution of multi-objective metaheuristics proposals over the past five years. This trend suggests a growing interest in multi-objective metaheuristics, likely due to the increasing complexity of real-world problems.

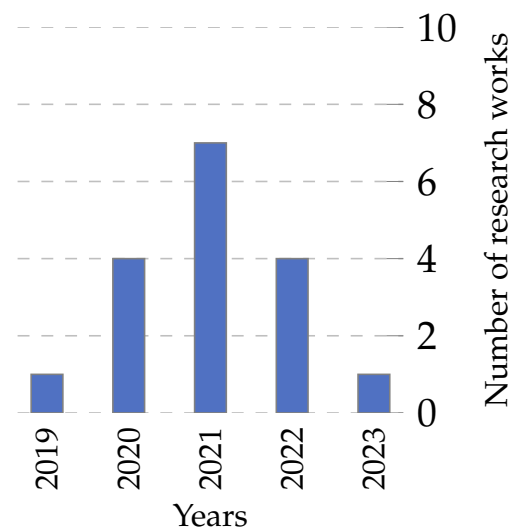


Figure 29. Multi-objective metaheuristics.

From the data, several observations can be made:

- There was a noticeable increase in multi-objective metaheuristic proposals from 2019 to 2021, peaking in 2021 with seven proposals [63–66,73,74,179]. This suggests a growing recognition of the importance of multi-objective approaches during this period.
- The numbers in 2022 and 2023 (up to April) show a decline, which could be attributed to various factors, including shifts in research focus or the maturation of multi-objective techniques developed in previous years.
- In the context of feature selection, multi-objective metaheuristics are invaluable. Feature selection often involves balancing reducing dimensionality (and thus computational cost) and retaining the most informative features for accurate prediction or classification. Multi-objective approaches provide a framework to navigate these conflicting objectives, ensuring robust and efficient models.

Table 5 presents an overview of the above-mentioned algorithms, where we highlight their primary areas of application, fundamental innovation, and the results of their respective evaluation processes.

Table 5. Overview of the multi-objective algorithms.

Year	Ref.	Algorithm	Focus	Innovation	Validation
2023	[70]	MOAEOSCA	Botnet detection in IoT opposition-based learning	Hybridization of AEO and SCA; bitwise operations	Achieved acceptable accuracy in Botnet detection in IoT
2022	[68]	CMODE	Multi-objective optimization and crowding distance	Rank based on non-dominated sorting optimization algorithms	Outperformed six state-of-the-art multi-objective algorithms
2022	[69]	PSOMMFS	High-dimensional feature selection adaptive local search	Information entropy-based initialization	Improved quality of Pareto front
2022	[120]	MOHHOAC	Feature selection using HHO chaotic local search	Associative learning; grey wolf optimization	Effective feature selection on sixteen UCI datasets
2022	[172]	BChOA	Biomedical data classification operator for enhanced exploration	Two binary variants of ChOA; crossover	Effective feature selection on biomedical datasets
2021	[63]	NSGA-III	Multi-label data feature selection maximizing feature-label correlation	Incorporation of additional objectives	Outperformed other algorithms on eight multi-label datasets
2021	[64]	DAEA	Bi-objective feature selection in classification diversity-based selection method	Duplication analysis method	Superior performance on 20 classification datasets
2021	[65]	MOPSO-ASFS	High-dimensional feature selection particle selection mechanism	Adaptive penalty mechanism; adaptive leading	Enhanced performance on high-dimensional datasets
2021	[66]	MOBIFS	Multi-objective feature selection roulette wheel mechanism	Bacterial foraging optimization algorithm	Effective removal of redundant features
2021	[73]	MOBGA-AOS	Feature selection as a pre-processing technique five crossover operators	Adaptive operator selection mechanism	Outperformed other evolutionary multi-objective algorithms
2021	[74]	MOIA/D-FSRank	Feature selection in L2R clonal selection and mutation operators	Tchebycheff decomposition; elite selection strategy	Significant improvements on public LETOR datasets
2021	[179]	OBCOOA	Wrapper-based feature selection; opposition-based learning mechanism	Time-varying V-shape transfer function	Applied to 27 benchmark datasets
2020	[23]	MOSCA_FS	Hyperspectral imagery feature selection Jeffries–Matusita distance and mutual information	Novel discrete SCA framework; ratio between	Tested on diverse datasets
2020	[71]	BMOGW	Feature selection	Multi-objective grey wolf optimizer	Effective feature selection with reduced classification error rates
2020	[72]	BCNSG3 & BCNSG2	Multi-objective feature selection	Cuckoo optimization algorithm	Achieved non-dominated solutions with reduced error rates
2020	[175]	EGA	Early time-series classification mathematical model targeting classification performance	Emphasis on the starting time of classification	Outperformed a general genetic algorithm
2019	[62]	TMABC-FS	Cost-sensitive feature selection diversity-guiding searches; dual-archive system	Introduction of convergence and	Demonstrated robustness on UCI datasets

4.4.6. Relationship between Objective Function Formulation and Metaheuristics

The relationship between the formulation of the objective function and the chosen metaheuristic offers a lens into the evolving research preferences and trends in feature selection. Figure 30 elucidates this relationship, detailing the distribution of articles based on the objective function and the metaheuristic employed.

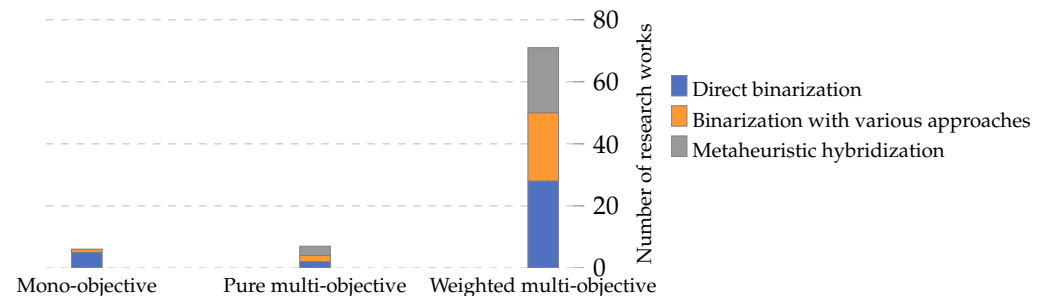


Figure 30. Objective function formulation and metaheuristic (hybridization and binarization).

The majority of articles with a “weighted multi-objective” formulation predominantly employ both “direct binarization” [73,80,82,87,94,95,98,101,104,105,108,112,115,118,119,123,133,134,136,149,150,152,153,158,159,161,169,177] and “binarization with various approaches” [78,89,91,92,96,106,107,110,114,124,129,131,132,138,142,144,146,154,165,172,176,179], with a similar inclination towards “metaheuristic hybridization” [85–88,92,97,112,113,121,135–137,140,144,145,154,155,157,159,161,173].

Interestingly, “pure multi-objective” formulations, while having a limited presence in “direct binarization” [63,67] and “binarization with various approaches” [71,179], exhibit just a slight edge towards “metaheuristic hybridization” [69,70,72].

Conversely, “mono-objective” formulations show a strong favor for “direct binarization” [21,35,36,40,48], with minimal exploration of other metaheuristics. This distribution underscores a discernible trend: researchers increasingly lean towards weighted multi-objective formulations when delving into diverse metaheuristics, likely due to the adaptability and robustness these formulations provide in tackling intricate feature selection challenges.

4.5. Which Datasets Are Commonly Used as Benchmarks, and Which Are Derived from Real-World Applications?

This section offers a detailed analysis of dataset usage across the reviewed articles. It includes an overview of the commonality of benchmark datasets, their real-world applications, and instances where they are combined. The selection of datasets and their sources is pivotal in ensuring research efficacy and relevance, especially in feature selection and metaheuristics. We thoroughly examine the most frequently used datasets, elaborating on their origins, characteristics, and distinct features. Further, this section shows the repositories from which these datasets are sourced, equipping readers with a comprehensive understanding of the data landscape integral to the studies under review.

4.5.1. Overall Trend in Dataset Usage

Figure 31 shows the trend in dataset usage across articles focused on feature selection and metaheuristics over five years, from 2019 to 2023. The chart is segmented by the year of publication, indicating the number of articles produced each year, the total datasets utilized, and the average number of datasets employed per article. From the data, in 2019, 29 articles made use of a combined total of 395 datasets, averaging 13.62 datasets per article. The following year, 2020, also witnessed 29 articles. However, there was an increase in dataset usage, with a cumulative count of 433, translating to an average of 14.93 datasets for each article. 2021 experienced a slight increase in the number of articles to 33, utilizing 457 datasets in total. The mean datasets per article stood at 13.85. A significant surge

was noted in 2022, with 55 articles being published. These articles utilized 840 datasets, averaging 15.27 datasets per article. As of the current year, 2023, data from 15 articles has been analyzed. These articles have used 231 datasets, leading to an average of 15.40 datasets per article, slightly higher than the previous year.

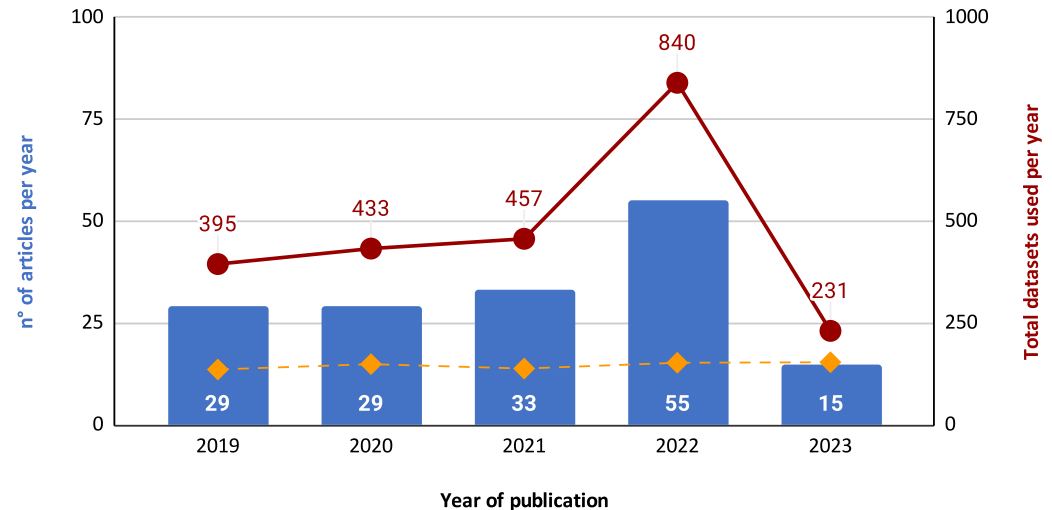


Figure 31. Dataset utilization and article publications trends over time.

Building upon the dataset usage trends, Figure 32 offers a deeper exploration into the nature of these datasets, categorizing them as ‘benchmark only’, ‘real-world application’, or a combination of ‘both benchmark and real-world applications’. The authors consider “benchmark” those datasets that were described as such in the articles reviewed or that are typically described as such in the literature. On the other hand, “real-world application” datasets were constructed specifically for the article in question or from dataset repositories whose purpose is to provide datasets for researchers to conduct investigations on these datasets.

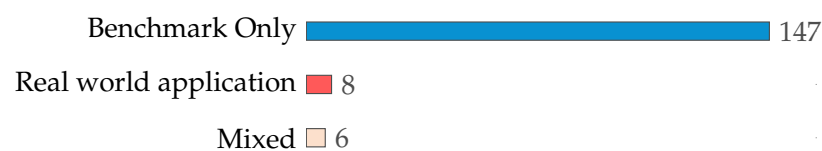


Figure 32. Distribution of benchmark vs. real-world applications in the articles.

From the data presented in Figure 32, a predominant trend emerges related to the utilization of benchmark datasets [27–34,36–43,46–50,53–60,62–75,77–82,84–104,106–140,142–146,148–163,165–177,179–186]. This underscores the preference for controlled, standardized datasets that allow comparative analysis across different feature selections and metaheuristics. In contrast, a smaller subset of eight articles focused solely on real-world application datasets [21,35,45,51,83,105,164,178]. These articles prioritized practical, real-world implications and the challenges that come with them. A nuanced approach was seen in six articles, which employed both benchmark and real-world datasets [22,23,44,141,147,187]. This suggests a comprehensive methodology that balances the theoretical robustness of benchmark datasets with the practical relevance of real-world data.

This distribution highlights the prevailing inclination towards benchmark datasets in the domain. However, the existence of papers using real-world datasets or combining both suggests a budding recognition of the importance of grounding research in real-world scenarios and challenges.

4.5.2. Real-World Application Datasets and Their Characteristics

Following the analysis presented in Figure 32, this subsection delves into the specifics of datasets employed in real-world application studies. These studies, though fewer in number compared to those using benchmark datasets, provide crucial insights into the application of machine learning in practical settings.

- The authors in [178] utilized a dataset constructed from the Twitter API focusing on cancer and drugs, enabling sentiment analysis and text classification.
- For industrial maintenance, The authors in [44] employed a dataset designed for motor fault detection.
- The authors in [164] involved a dataset of 553 drugs bio-transformed in the liver, annotated with toxic effects such as irritant, mutagenic, reproductive, and tumorigenic, each represented by chemical descriptors.
- The authors in [45,51,83] made significant use of the NinaPro database. This database provided EMG signals from healthy subjects and amputees, covering various hand and wrist motions essential for prosthetics and rehabilitation research.
- The authors in [23] used hyperspectral image datasets and spectral data of typical surface features, indicating the application of machine learning in environmental monitoring.
- The authors in [35], a dataset of 500 Arabic email messages from computer science students was analyzed, showing machine learning's application in language processing and cybersecurity.
- The authors in [21] examined data from Iraqi cancer patients, offering a comprehensive dataset for healthcare research across multiple cancer types.
- The authors in [22] focused on constructing a dataset from Zeek network-based intrusion detection logs, underscoring machine learning's role in network security.
- The authors in [187] presented a dataset related to medical treatment for cardiogenic shock, highlighting the intersection of machine learning and medical research.
- The authors in [141,147] demonstrated the versatility of machine learning in biological and medical research using datasets from NCBI and brain imaging datasets for disease classification.

These datasets represent a shift towards employing machine learning in diverse and practical scenarios, extending beyond the controlled conditions of benchmark datasets. The use of real-world data challenges the algorithms' robustness in less predictable environments and ensures the relevance and applicability of machine learning solutions in addressing real-world problems. In reviewing the documentation of these datasets, it becomes evident that clarity in their descriptions is paramount, especially for replicability and application of the research. Benchmark datasets often benefit from well-established documentation practices, clearly outlining their structure in terms of instances, features, and labels. Conversely, some real-world application datasets may have less clear descriptions, particularly those that are custom-constructed or adapted. This is especially challenging in datasets related to text classification, where the complexity of textual data can lead to an extensive range of features. While these documentation discrepancies do not diminish the value of the research, they highlight an opportunity for enhancing reporting standards. Improved clarity and completeness in dataset descriptions would greatly benefit the field by fostering transparency, facilitating study replication, and enhancing the applicability of research findings in real-world scenarios.

4.5.3. Prevalent Datasets and Their Defining Characteristics

Specific datasets have emerged as particularly influential, often serving as foundational benchmarks for multiple studies. By their comprehensive nature or unique characteristics, these datasets have become cornerstones for researchers, enabling rigorous testing and validation of methodologies. Next, we focus on spotlighting the datasets that have been instrumental in feature selection and optimization research. We will outline the most frequently used ones, detailing their source name, subject area, number of instances/samples,

features/characteristics, classes/labels, and the repositories or platforms where they can be accessed.

Figure 33 enumerates the twenty most referenced datasets in the articles under review. The “ionosphere” dataset emerges at the top with 82 mentions [27,29,37,39,40,42–44,46,47,50,54,55,60,62,64–66,68,70,71,77–80,82,84–92,94,95,97,100,102,104,106–109,111–113,115,118–120,124,125,127,130,132,134–136,138,139,142,143,145,146,151–155,157,158,160,161,166,167,169,179–181,185], closely trailed by the “Breast Cancer Wisconsin (Diagnostic)” dataset at 76 citations [27–29,37,43,47,53,55,60,64,65,71,73,77,79,81,82,84,86–89,91,94–97,99,101,103,106–109,111–115,118,119,121,123,127–130,132–136,138,139,142–145,147,149,152–155,158–161,163,169,173,179–181,183,186]. Datasets like “Sonar” [23,27,29,34,37,39,40,42–44,46,47,50,54,60,62,65,68,70,71,77,79,81,82,84,86–89,91,92,94,95,97,98,100,102,104,107–115,118,119,127,130,132,135,136,138,139,142–145,147,150–155,158,160,162,166,180,181,185], “Wine” [23,29,37,39,42,47,54,57,64–66,68,71,77–79,82,84,86,87,89–95,97,102–104,106–112,118–120,123–125,127,129,130,132,134–139,142–145,148,149,152,154,155,158,160,161,166,179,180,183,185], and Zoo [23,37,43,59,62,65,66,71,75,78,79,81,82,84,85,87–92,94,95,97,100,102–104,106–112,115,118–120,123–125,129,130,132,134–139,142–145,148,150,152–155,157,158,160,163,166,167,180,183,185] also secured noteworthy positions, with each being cited at least in 70 articles. The figure shows the most influential and recurrent datasets in feature selection and optimization research, offering insights into the datasets’ prevalence and importance in the academic discourse.

Looking deeper, out of the top ten datasets enumerated in Figure 33, seven pertain to the medical or biological domains. This dominance underscores the significant role of medical and biological data in feature selection and optimization research, possibly due to healthcare data’s complexity, relevance, and critical nature. Datasets such as “Ionosphere” and “Sonar” suggest a diverse application of feature selection techniques across varied fields. However, the prevalence of health-related datasets in the top ranks highlights the growing importance and challenges associated with medical data analytics.

Table 6 provides a detailed breakdown of the 20 most commonly utilized datasets. Each dataset is systematically categorized by:

- Source name: The standardized name or label of the dataset.
- Subject area: The domain or field from which the dataset originates, which reveals a significant leaning towards the medical and biological areas but also showcases diversity, with datasets from physical science, politics, games, and synthetic sources.
- Instances/samples: The number of individual data points or samples in each dataset.
- Features/characteristics: The number of attributes or characteristics each sample in the dataset has.
- Classes: The number of unique labels or outcomes into which the samples can be categorized.
- Reference: Based on DOI, a digital object identifier that provides a persistent link to a dataset.
- Repository: The platform or database from which the dataset can be accessed.

From a glance, it is evident that while datasets like ‘Ionosphere’ and ‘Sonar’ have fewer instances but a moderate number of features, datasets like ‘Colon’ stand out, with many features. The predominance of the UCI repository highlights its role as a primary hub for academic datasets. Moreover, the diversity in subject areas—from ‘physical science’ to ‘game’—underscores the wide applicability and versatility of feature selection and metaheuristics across various fields. It is important to note that within Table 6 three datasets originate from the UCI or ASU repositories, yet they could not be located within those repositories or others. This discrepancy is indicated by ‘n.a.’ (not available) or ‘n.d.’ (not determined) appearing in some table cells, suggesting that access to the information from the source repository is either restricted or unavailable.

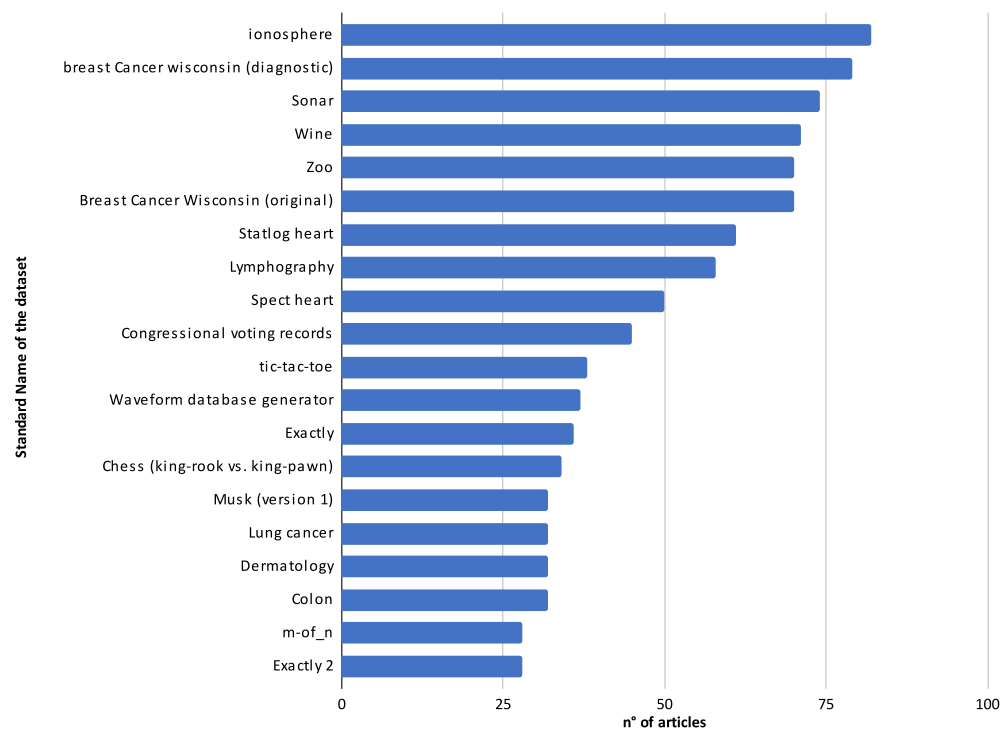


Figure 33. Frequency of dataset usage as a benchmark: top 20.

Table 6. This is a wide table.

Year	Ref.	Description	Field	Instances	Features	Classes	Repository
2018	[199]	Colon	Medical	62	2000	2	ASU
2001	[200]	SPECT Heart	Medical	267	22	2	UCI
1998	[201]	Dermatology	Medical	366	34	6	UCI
1994	[202]	Musk (Version 1)	Chemistry	476	166	2	UCI
1994	[203]	Breast Cancer Wisconsin (Diagnostic)	Medical	569	30	2	UCI
1992	[204]	Breast Cancer Wisconsin (Original)	Medical	699	9	2	UCI
1992	[205]	Lung Cancer	Medical	32	56	3	UCI
1991	[206]	Wine	Biology/Chemistry	178	13	3	UCI
1991	[207]	Tic-Tac-Toe Endgame	Game	958	9	2	UCI
1990	[208]	Zoo	Biology	101	16	7	UCI
1989	[209]	Ionosphere	Physical Science	351	34	2	UCI
1989	[210]	Chess (King-Rook vs. King-Pawn)	Game	3196	36	2	UCI
1988	[211]	Waveform Database Generator (Version 2)	Synthetic	5000	40	3	UCI
1988	[212]	Lymphography	Medical	148	18	4	UCI
1987	[213]	Congressional Voting Records	Politics	435	16	2	UCI
-	[214]	Sonar	Physical Science	208	60	2	UCI
-	[215]	Statlog heart	Medical	270	13	2	UCI
-	n.a.	Exactly	n.d.	1000	13	2	UCI
-	n.a.	Exactly2	n.d.	1000	13	2	UCI
-	n.a.	m-of-n	Biological	1000	13	2	UCI

4.5.4. A Glimpse into Data Sources

Specific repositories consistently emerged as favored choices for researchers in feature selection and optimization. Figure 34 presents a distribution of these articles across the most popular dataset repositories.

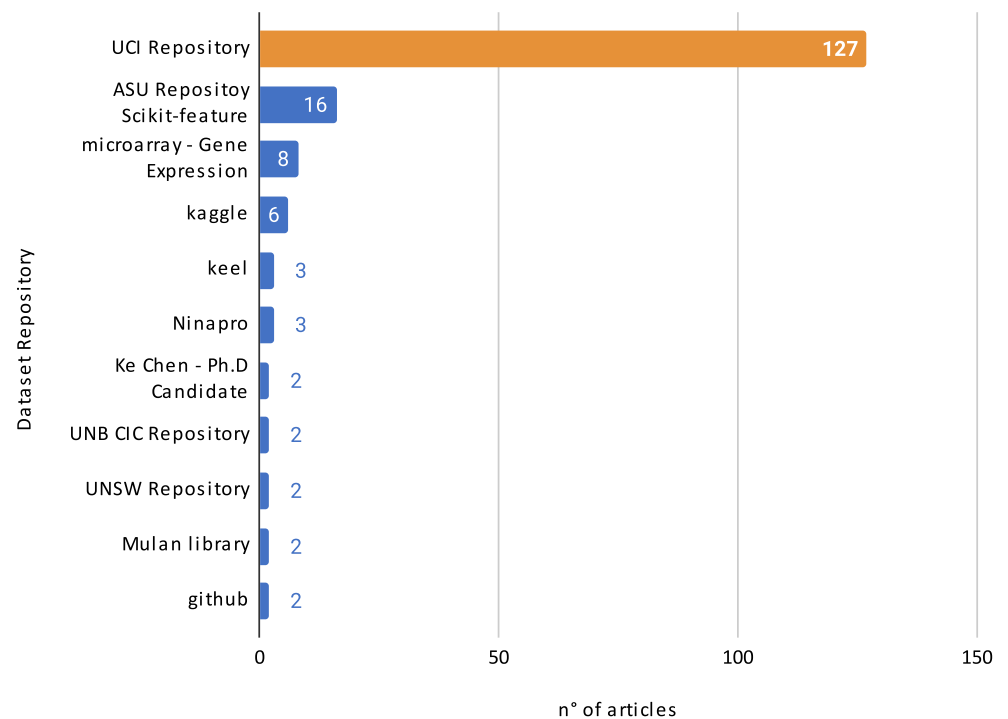


Figure 34. Distribution of articles across popular dataset repositories.

1. UCI Repository: Standing as a stalwart in the academic community, the UCI Repository was referenced by a substantial 127 articles [21,23,27–29,31,34,36–38,40–42,44–46,48,49,51,53,54,58,62–64,66–75,77–90,92,93,95–105,108,109,111–114,116,118–121,124–137,139–143,145,146,148–150,152–161,163–166,168–171,173,179,183–185]. A testament to its vast collection and diverse range of datasets, UCI has proven to be an indispensable resource.
2. ASU and Scikit-feature Repositories: Grouped together, these repositories were mentioned in 16 articles [30,36,38,48,49,55,60,67,104,112,116,125,126,156,165,184]. Recognized for specific types of datasets, these platforms offer specialized data that cater to specific niche research areas.
3. Microarray-Gene Expression Datasets: This repository, with a focus on gene expression data, was mentioned in eight articles [65,77,96,116,117,124,154]. It underscores the increasing interest in genomic data and their importance in feature selection studies.
4. Kaggle: A platform widely known for its machine learning competitions, Kaggle also houses an extensive array of datasets. It was cited in six of the reviewed articles [40,131,133,147,148,187].
5. KEEL: With three mentions [40,47,184], the KEEL repository, which emphasizes evolutionary algorithms and data mining, has a defined user base in our set of articles.
6. Ninapro: Also receiving three mentions [45,51,83], Ninapro, which specializes in hand and finger movements, signals its relevance in biomechanical studies.
7. Miscellaneous repositories: Several repositories mentioned in two articles were Ke Chen—Ph.D. Candidate Datasets Repository [36,38], which caters to specific academic projects; the UNB CIC [34,56] and UNSW Repositories [22,56], known for cybersecurity and network datasets; and the Mulan Library [41,63], emphasizing multi-label learning datasets.

While these top ten repositories encompassed the majority of citations, a range of other repositories were also sourced, albeit less frequently. Although not as dominant, these repositories contribute uniquely to the mosaic of datasets available to researchers. The diversity of repositories indicates the breadth and depth of research in feature selection and optimization, with datasets ranging from political records to intricate genomic data.

To facilitate easy access to the various data repositories referenced, Table 7 compiles a comprehensive list of these resources, complete with functional web links verified at the time of writing this article. The aim is to provide a handy, accessible directory of these repositories, ensuring researchers can efficiently retrieve datasets. In the table, the commonly utilized names for the sources of the datasets are systematically presented based on the information reported in the articles. A webpage is provided alongside each name, derived from the citations or references noted in the articles. The third column of the table details the primary use of each repository. The authors make a clear distinction: ‘benchmarks’ implies that the repository is predominantly used as a source of datasets frequently employed in machine learning benchmarks and related domains. These datasets serve as standard tests or evaluations for algorithms and models, facilitating comparisons across methodologies.

On the other hand, ‘real-world applications’ signify that the repository is rich in datasets primarily derived from authentic, real-world problems, spanning fields such as medicine, biology, cybersecurity, and more. These datasets are provided to the academic and research communities to foster research that can lead to tangible improvements in human life. They present unique challenges and opportunities for innovation, aiming to contribute to developing solutions or new methodologies that can significantly enhance the quality of life and address pressing real-world issues. All links shown in the Source column of Table 7 were last accessed on 20 December 2023.

Table 7. Curated repositories of datasets utilized in feature selection and optimization research.

Repository	Source	Use
AI Studio	https://aistudio.baidu.com/	Real-world application
ASU Repository (Scikit-feature Repository)	http://featureselection.asu.edu/datasets.php	Benchmark
AWID dataset	https://icsdweb.aegean.gr/awid/	Benchmark
BCI Competitions	https://www.bbci.de/competition/	Benchmark
Biopatrec Repository	https://github.com/biopatrec/biopatrec	Real-world application
Causality workbench	https://www.causality.inf.ethz.ch/	Benchmark
Dr. Wang’s Repository dataset	http://infosec.bjtu.edu.cn/wangwei/?page_id=85	Benchmark
Drug bank database	https://go.drugbank.com/	Real-world application
Github	https://github.com/	Benchmark
Kaggle	https://www.kaggle.com/	Benchmark
Keel	https://sci2s.ugr.es/keel/datasets.php	Benchmark
Ke Chen - Ph.D Candidate Datasets Repository	https://ckzixf.github.io/dataset.html	Benchmark
Mulan Library	https://mulan.sourceforge.net/	Benchmark
NCBI National Centre for Biotechnology Information	https://www.ncbi.nlm.nih.gov/	Real-world application
Near East Hospital	https://neareasthospital.com/	Real-world application
Ninapro	https://www.idiap.ch/project/ninapro/	Real-world application
Letor, By Microsoft	https://www.microsoft.com/en-us/research/	Benchmark
Papers with code	https://paperswithcode.com/	Benchmark
Physionet	https://physionet.org/	Real-world application
Quare.ai HeadCT Study	http://headctstudy.quare.ai/	Real-world application
RDRR	https://rdrr.io/r/utlis/data.html	Benchmark
RSNA	https://www.rsna.org/	Real-world application
Time Series Machine Learning Website	https://www.timeseriesclassification.com/	Benchmark
UCI Repository	https://archive.ics.uci.edu/	Benchmark
UNB CIC Repository	https://www.unb.ca/cic/datasets/index.html	Benchmark
UNSW Repository	https://unswworks.unsw.edu.au/home	Benchmark
Yahoo	https://webscope.sandbox.yahoo.com/	Benchmark
Zexuan ZHU Professor Datasets Repository	https://csse.szu.edu.cn/staff/zhux/index.html	Benchmark

4.6. Closing of Discussions

This section synthesizes the findings from the literature review, addressing the five research questions that guided our investigation of the feature selection problem.

- Objective function formulation (RQ1): Our review revealed a diversity of objective functions used in feature selection, generally classified as single-objective or multi-objective functions. We observed that while single-objective functions focus on optimizing a single criterion, multi-objective functions, including pure and weighted types, cater to multiple criteria simultaneously. Weighted multi-objective functions were more prevalent in our dataset, suggesting their broader applicability in complex scenarios.
- Performance metrics (RQ2): We classified the performance metrics used in feature selection research into four main categories: classifier metrics, metaheuristic metrics, feature metrics, and statistical tests. Classifier metrics are the most frequently used, emphasizing the importance of the machine learning technique's performance. The significant use of metaheuristic metrics and feature metrics underscores the complexity of evaluating feature selection methods.
- Used machine learning techniques (RQ3): We investigated machine learning techniques that are improved by feature selection. We found that a variety of classifiers are used, with k-nearest neighbor (k-NN) being the most common. The prevalence of techniques such as SVM, naive Bayes, and decision tree classifiers, including DT C4.5 and random forest, illustrates the wide applicability of feature selection across different learning paradigms.
- Metaheuristics (RQ4): Our study highlights the significant role of metaheuristics in feature selection, particularly particle swarm optimization (PSO), grey wolf optimizer (GWO), and genetic algorithm (GA). Their frequent use points to a preference for adaptive, population-based algorithms adept at handling the complex aspects of feature selection. This observation not only confirms the effectiveness of these methods but also suggests promising directions for future research in enhancing feature selection procedures.
- Practical applications and trends (RQ5): Our analysis of dataset usage trends in feature selection research reveals a slight increase in the number of datasets used per article over time. This shift, along with the dominant use of benchmark datasets and a focus on real-world applications, reflects the escalating complexity and practical significance of feature selection studies. The variety of dataset sources, especially the frequent citation of the UCI Repository, demonstrates the extensive applicability of feature selection in diverse domains.

5. Conclusions

In this work, we were able to evidence of the complex and broad field of research related to feature selection. Metaheuristics are algorithms that play a significant role in different Combinatorial Optimization Problems such as Set Covering [216–220], Knapsack Problem [221,222] and Cell Formation Problem [223]. There is also a high interest in hybridizations and modifying native metaheuristics. This is due to the No Free Lunch Theorem, which allows for continued research into new ways of improving metaheuristics.

In this work, we have found that the problem of feature selection has been constantly changing and challenging to be solved with metaheuristics. Given this, we propose a robust evaluation process tailored to compare the effectiveness of various combinations of methods. This process is based on a standardized framework that encompasses three fundamental components:

- Selection of Objective Function: It is interesting to note that the same optimization problem can be represented through three different types of objective functions, each increasing the complexity of the problem. For researchers who are just starting in the field of feature selection, we recommend starting by solving the problem from a

single-objective perspective, then moving on to weighted multi-objective, and finally to pure multi-objective.

- Selection of Evaluation Metrics: Regarding metrics, we can observe that there are 4 major groups which are classifier metrics, metaheuristic metrics, feature metrics, and statistical tests. For robustness in future research, we recommend incorporating at least one metric from each of the reported categories.
 - For classifier metrics, we recommend using Accuracy, Error Rate, Precision, Recall, and F-score.
 - For the case of metaheuristic metrics, we recommend using the computational time, the fitness in the case of using a mono-objective function or weighted multi-objective function, and the hyper-volume metric in the case of using a pure multi-objective function.
 - In the case of feature metrics, we recommend reporting the number of features selected and which features were selected.
 - For the case of statistical test, we recommend advocating for a balanced application of both non-parametric tests, such as the Wilcoxon and Friedman tests, and parametric tests like the T-test, supplemented by rigorous post hoc analyses for in-depth insights

A metric that, in our opinion, should be included in all research is indicating the solution vector, that is, indicating which features were selected by the metaheuristics.

- Selection of classifier: The choice of classifier will depend closely on the dataset used where the important issues to be considered are the unbalance of the target classes, whether it is multi-class or binary-class, and the number of samples. In this sense, we recommend experimenting with more than one classifier to express robust results and can use the KNN, Random Forest, or Xgboost.
- Selection of Benchmark Dataset: Guided by a curated list of the top 20 datasets, ensuring that experimentation and comparison are grounded in both established and innovative contexts.

This proposed framework aims to standardize and elevate the comparative analysis in feature selection and metaheuristics research, fostering a more consistent, transparent, and replicable approach in future studies.

With all these standardized steps you can innovate, experiment, and focus on proposing new ideas in the field of metaheuristics supported by the No Free Lunch Theorem [17–19]. By implementing this framework, we envision a significant enhancement in the comparability and reliability of findings in this field, thereby contributing to its methodological rigor and practical applicability.

Author Contributions: Conceptualization, J.B.-G., F.C.-C. and B.C.; methodology, J.B.-G. and F.C.-C.; validation, B.C., M.G.S. and R.S.; formal analysis, J.B.-G., F.C.-C. and M.G.S.; investigation, J.B.-G., F.C.-C., B.C., M.G.S. and R.S.; resources, J.B.-G. and F.C.-C.; writing—original draft J.B.-G., F.C.-C., B.C. and M.G.S.; writing—review and editing, R.S.; supervision, B.C. and R.S.; funding acquisition, B.C. and R.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Data Availability Statement: Data are contained within the article.

Acknowledgments: Broderick Crawford and Ricardo Soto are supported by Grant ANID/ FONDECYT/ REGULAR/1210810. Felipe Cisternas-Caneo is supported by National Agency for Research and Development (ANID)/ Scholarship Program/DOCTORADO NACIONAL/2023-21230203. Broderick Crawford, Ricardo Soto and Felipe Cisternas-Caneo are supported by Grant DI Investigación Asociativa Interdisciplinaria/VINCI/PUCV/039.347/2023.

Conflicts of Interest: The authors declare no conflict of interest.

References

- Witten, I.H.; Frank, E. Data mining: Practical machine learning tools and techniques with Java implementations. *ACM Sigmod Rec.* **2002**, *31*, 76–77.
- Guyon, I.; Elisseeff, A. An introduction to variable and feature selection. *J. Mach. Learn. Res.* **2003**, *3*, 1157–1182.
- Agrawal, P.; Abutarboush, H.F.; Ganesh, T.; Mohamed, A.W. Metaheuristic Algorithms on Feature Selection: A Survey of One Decade of Research (2009–2019). *IEEE Access* **2021**, *9*, 26766–26791. <https://doi.org/10.1109/ACCESS.2021.3056407>.
- Nssibi, M.; Manita, G.; Korbaa, O. Advances in nature-inspired metaheuristic optimization for feature selection problem: A comprehensive survey. *Comput. Sci. Rev.* **2023**, *49*, 100559. <https://doi.org/10.1016/j.cosrev.2023.100559>.
- Kurman, S.; Kisan, S. An in-depth and contrasting survey of meta-heuristic approaches with classical feature selection techniques specific to cervical cancer. *Knowl. Inf. Syst.* **2023**, *65*, 1881–1934. <https://doi.org/10.1007/s10115-022-01825-y>.
- Pham, T.H.; Raahemi, B. Bio-Inspired Feature Selection Algorithms with Their Applications: A Systematic Literature Review. *IEEE Access* **2023**, *11*, 43733–43758. <https://doi.org/10.1109/ACCESS.2023.3272556>.
- Sadeghian, Z.; Akbari, E.; Nematzadeh, H.; Motameni, H. A review of feature selection methods based on meta-heuristic algorithms. *J. Exp. Theor. Artif. Intell.* **2023**, *0*, 1–51. <https://doi.org/10.1080/0952813X.2023.2183267>.
- Arun Kumar, R.; Vijay Franklin, J.; Koppula, N. A Comprehensive Survey on Metaheuristic Algorithm for Feature Selection Techniques. *Mater. Today Proc.* **2022**, *64*, 435–441. International Conference on Advanced Materials for Innovation and Sustainability. <https://doi.org/10.1016/j.matpr.2022.04.803>.
- Akinola, O.O.; Ezugwu, A.E.; Agushaka, J.O.; Zitar, R.A.; Abualigah, L. Multiclass feature selection with metaheuristic optimization algorithms: A review. *Neural Comput. Appl.* **2022**, *34*, 19751–19790. <https://doi.org/10.1007/s00521-022-07705-4>.
- Dokeroglu, T.; Deniz, A.; Kiziloğlu, H.E. A comprehensive survey on recent metaheuristics for feature selection. *Neurocomputing* **2022**, *494*, 269–296. <https://doi.org/10.1016/j.neucom.2022.04.083>.
- Abu Khurma, R.; Aljarah, I.; Sharieh, A.; Abd Elaziz, M.; Damaševičius, R.; Krilavičius, T. A Review of the Modification Strategies of the Nature Inspired Algorithms for Feature Selection Problem. *Mathematics* **2022**, *10*, 464. <https://doi.org/10.3390/math10030464>.
- Yab, L.Y.; Wahid, N.; Hamid, R.A. A Meta-Analysis Survey on the Usage of Meta-Heuristic Algorithms for Feature Selection on High-Dimensional Datasets. *IEEE Access* **2022**, *10*, 122832–122856. <https://doi.org/10.1109/ACCESS.2022.3221194>.
- Abiodun, E.O.; Alabdulatif, A.; Abiodun, O.I.; Alawida, M.; Alabdulatif, A.; Alkhalwaldeh, R.S. A systematic review of emerging feature selection optimization methods for optimal text classification: The present state and prospective opportunities. *Neural Comput. Appl.* **2021**, *33*, 15091–15118. <https://doi.org/10.1007/s00521-021-06406-8>.
- Al-Tashi, Q.; Abdulkadir, S.J.; Rais, H.M.; Mirjalili, S.; Alhussian, H. Approaches to Multi-Objective Feature Selection: A Systematic Literature Review. *IEEE Access* **2020**, *8*, 125076–125096. <https://doi.org/10.1109/ACCESS.2020.3007291>.
- Talbi, E.G. *Metaheuristics: From Design to Implementation*; John Wiley & Sons: Hoboken, NJ, USA, 2009.
- Rajwar, K.; Deep, K.; Das, S. An exhaustive review of the metaheuristic algorithms for search and optimization: Taxonomy, applications, and open challenges. *Artif. Intell. Rev.* **2023**, *56*, 13187–13257. <https://doi.org/10.1007/s10462-023-10470-y>.
- Ho, Y.C.; Pepyne, D.L. Simple explanation of the no-free-lunch theorem and its implications. *J. Optim. Theory Appl.* **2002**, *115*, 549–570. <https://doi.org/10.1023/A:1021251113462>.
- Igel, C. No free lunch theorems: Limitations and perspectives of metaheuristics. In *Theory and Principled Methods for the Design of Metaheuristics*; Springer: Berlin/Heidelberg, Germany, 2014; pp. 1–23. https://doi.org/10.1007/978-3-642-33206-7_1.
- Wolpert, D.; Macready, W. No free lunch theorems for optimization. *IEEE Trans. Evol. Comput.* **1997**, *1*, 67–82. <https://doi.org/10.1109/4235.585893>.
- Becerra-Rozas, M.; Lemus-Romani, J.; Cisternas-Caneo, F.; Crawford, B.; Soto, R.; Astorga, G.; Castro, C.; García, J. Continuous metaheuristics for binary optimization problems: An updated systematic literature review. *Mathematics* **2022**, *11*, 129. <https://doi.org/10.3390/math11010129>.
- Ibrahim, H.T.; Mazher, W.J.; Jassim, E.M. Feature Selection: Binary Harris Hawk Optimizer Based Biomedical Datasets. *Intel. Artif.* **2022**, *25*, 33–49.
- Chohra, A.; Shirani, P.; Karbab, E.B.; Debbabi, M. Chameleon: Optimized feature selection using particle swarm optimization and ensemble methods for network anomaly detection. *Comput. Secur.* **2022**, *117*, 102684. <https://doi.org/10.1016/j.cose.2022.102684>.
- Wan, Y.; Ma, A.; Zhong, Y.; Hu, X.; Zhang, L. Multiobjective Hyperspectral Feature Selection Based on Discrete Sine Cosine Algorithm. *IEEE Trans. Geosci. Remote Sens.* **2020**, *58*, 3601–3618. <https://doi.org/10.1109/TGRS.2019.2958812>.
- Kitchenham, B. Procedures for performing systematic reviews. *Keele UK Keele Univ.* **2004**, *33*, 1–26.
- Aria, M.; Cuccurullo, C. bibliometrix: An R-tool for comprehensive science mapping analysis. *J. Inf.* **2017**, *11*, 959–975. <https://doi.org/10.1016/j.joi.2017.08.007>.
- Boyd, S.P.; Vandenberghe, L. *Convex Optimization*; Cambridge University Press: Cambridge, UK, 2004.
- Engelbrecht, A.P.; Grobler, J.; Langeveld, J. Set based particle swarm optimization for the feature selection problem. *Eng. Appl. Artif. Intell.* **2019**, *85*, 324–336. <https://doi.org/10.1016/j.engappai.2019.06.008>.
- Xue, Y.; Xue, B.; Zhang, M. Self-Adaptive Particle Swarm Optimization for Large-Scale Feature Selection in Classification. *ACM Trans. Knowl. Discov. Data* **2019**, *13*, 1–27. <https://doi.org/10.1145/3340848>.
- Huang, Z.; Yang, C.; Zhou, X.; Huang, T. A Hybrid Feature Selection Method Based on Binary State Transition Algorithm and ReliefF. *IEEE Biomed. Health Inform.* **2019**, *23*, 1888–1898. <https://doi.org/10.1109/JBHI.2018.2872811>.

30. Zhang, Y.; Li, H.G.; Wang, Q.; Peng, C. A filter-based bare-bone particle swarm optimization algorithm for unsupervised feature selection. *Appl. Intell.* **2019**, *49*, 2889–2898.
31. Mohammed, T.A.; Bayat, O.; Uçan, O.N.; Alhayali, S. Hybrid efficient genetic algorithm for big data feature selection problems. *Found. Sci.* **2020**, *25*, 1009–1025.
32. Tan, P.; Wang, X.; Wang, Y. Dimensionality reduction in evolutionary algorithms-based feature selection for motor imagery brain-computer interface. *Swarm Evol. Comput.* **2020**, *52*, 100597. <https://doi.org/10.1016/j.swevo.2019.100597>.
33. Alsaleh, A.; Binsaeedan, W. The Influence of Salp Swarm Algorithm-Based Feature Selection on Network Anomaly Intrusion Detection. *IEEE Access* **2021**, *9*, 112466–112477. <https://doi.org/10.1109/ACCESS.2021.3102095>.
34. Wang, L.; Gao, Y.; Li, J.; Wang, X. A feature selection method by using chaotic cuckoo search optimization algorithm with elitist preservation and uniform mutation for data classification. *Discret. Dyn. Nat. Soc.* **2021**, *2021*, 1–19.
35. BinSaeedan, W.; Alramlawi, S. CS-BPSO: Hybrid feature selection based on chi-square and binary PSO algorithm for Arabic email authorship analysis. *Knowl.-Based Syst.* **2021**, *227*, 107224. <https://doi.org/10.1016/j.knosys.2021.107224>.
36. Yang, J.Q.; Chen, C.H.; Li, J.Y.; Liu, D.; Li, T.; Zhan, Z.H. Compressed-Encoding Particle Swarm Optimization with Fuzzy Learning for Large-Scale Feature Selection. *Symmetry* **2022**, *14*, 1142. <https://doi.org/10.3390/sym14061142>.
37. Long, W.; Jiao, J.; Xu, M.; Tang, M.; Wu, T.; Cai, S. Lens-imaging learning Harris hawks optimizer for global optimization and its application to feature selection. *Expert Syst. Appl.* **2022**, *202*, 117255. <https://doi.org/10.1016/j.eswa.2022.117255>.
38. Yang, J.Q.; Yang, Q.T.; Du, K.J.; Chen, C.H.; Wang, H.; Jeon, S.W.; Zhang, J.; Zhan, Z.H. Bi-Directional Feature Fixation-Based Particle Swarm Optimization for Large-Scale Feature Selection. *IEEE Trans. Big Data* **2023**, *9*, 1004–1017. <https://doi.org/10.1109/TBDATA.2022.3232761>.
39. Hu, Y.; Zhang, Y.; Gao, X.; Gong, D.; Song, X.; Guo, Y.; Wang, J. A federated feature selection algorithm based on particle swarm optimization under privacy protection. *Knowl.-Based Syst.* **2023**, *260*, 110122. <https://doi.org/10.1016/j.knosys.2022.110122>.
40. Feizi-Derakhsh, M.R.; Kadhim, E.A. An Improved Binary Cuckoo Search Algorithm For Feature Selection Using Filter Method And Chaotic Map. *J. Appl. Sci. Eng.* **2022**, *26*, 895–901. [https://doi.org/10.6180/jase.202306_26\(6\).0015](https://doi.org/10.6180/jase.202306_26(6).0015).
41. Park, J.; Park, M.W.; Kim, D.W.; Lee, J. Multi-Population Genetic Algorithm for Multilabel Feature Selection Based on Label Complementary Communication. *Entropy* **2020**, *22*, 876. <https://doi.org/10.3390/e22080876>.
42. Du, Z.; Han, D.; Li, K.C. Improving the performance of feature selection and data clustering with novel global search and elite-guided artificial bee colony algorithm. *J. Supercomput.* **2019**, *75*, 5189–5226.
43. Zakeri, A.; Hokmabadi, A. Efficient feature selection method using real-valued grasshopper optimization algorithm. *Expert Syst. Appl.* **2019**, *119*, 61–72. <https://doi.org/10.1016/j.eswa.2018.10.021>.
44. Pourpanah, F.; Lim, C.P.; Wang, X.; Tan, C.J.; Seera, M.; Shi, Y. A hybrid model of fuzzy min-max and brain storm optimization for feature selection and data classification. *Neurocomputing* **2019**, *333*, 440–451. <https://doi.org/10.1016/j.neucom.2019.01.011>.
45. Too, J.; Abdullah, A.R.; Mohd Saad, N.; Tee, W. EMG feature selection and classification using a Pbest-guide binary particle swarm optimization. *Computation* **2019**, *7*, 12.
46. Tubishat, M.; Idris, N.; Shuib, L.; Abushariah, M.A.; Mirjalili, S. Improved Salp Swarm Algorithm based on opposition based learning and novel local search algorithm for feature selection. *Expert Syst. Appl.* **2020**, *145*, 113122. <https://doi.org/10.1016/j.eswa.2019.113122>.
47. Slezkin, A.; Hodashinsky, I.A.; Shelupanov, A.A. Binarization of the Swallow swarm optimization for feature selection. *Program. Comput. Softw.* **2021**, *47*, 374–388.
48. Akinola, O.A.; Agushaka, J.O.; Ezugwu, A.E. Binary dwarf mongoose optimizer for solving high-dimensional feature selection problems. *PLoS ONE* **2022**, *17*, e0274850. <https://doi.org/10.1371/journal.pone.0274850>.
49. Wang, J.; Zhang, Y.; Hong, M.; He, H.; Huang, S. A self-adaptive level-based learning artificial bee colony algorithm for feature selection on high-dimensional classification. *Soft Comput.* **2022**, *26*, 9665–9687. <https://doi.org/10.1007/s00500-022-06826-1>.
50. Tubishat, M.; Rawshdeh, Z.; Jarrah, H.; Elgamal, Z.M.; Elnagar, A.; Alrashdan, M.T. Dynamic generalized normal distribution optimization for feature selection. *Neural Comput. Appl.* **2022**, *34*, 17355–17370. <https://doi.org/10.1007/s00521-022-07398-9>.
51. Too, J.; Abdullah, A.R.; Mohd Saad, N.; Mohd Ali, N.; Tee, W. A New Competitive Binary Grey Wolf Optimizer to Solve the Feature Selection Problem in EMG Signals Classification. *Computers* **2018**, *7*, 58. <https://doi.org/10.3390/computers7040058>.
52. Bezdek, J.C.; Ehrlich, R.; Full, W. FCM: The fuzzy c-means clustering algorithm. *Comput. Geosci.* **1984**, *10*, 191–203. [https://doi.org/10.1016/0098-3004\(84\)90020-7](https://doi.org/10.1016/0098-3004(84)90020-7).
53. Anter, A.M.; Ali, M. Feature selection strategy based on hybrid crow search optimization algorithm integrated with chaos theory and fuzzy c-means algorithm for medical diagnosis problems. *Soft Comput.* **2020**, *24*, 1565–1584.
54. Qiu, C.; Xiang, F. Feature selection using a set based discrete particle swarm optimization and a novel feature subset evaluation criterion. *Intell. Data Anal.* **2019**, *23*, 5–21.
55. Wang, Y.; Li, X.; Wang, J. A neurodynamic optimization approach to supervised feature selection via fractional programming. *Neural Netw.* **2021**, *136*, 194–206. <https://doi.org/10.1016/j.neunet.2021.01.004>.
56. Halim, Z.; Yousaf, M.N.; Waqas, M.; Sulaiman, M.; Abbas, G.; Hussain, M.; Ahmad, I.; Hanif, M. An effective genetic algorithm-based feature selection method for intrusion detection systems. *Comput. Secur.* **2021**, *110*, 102448. <https://doi.org/10.1016/j.cose.2021.102448>.
57. Hanbay, K. A new standard error based artificial bee colony algorithm and its applications in feature selection. *J. King Saud Univ.-Comput. Inf. Sci.* **2022**, *34*, 4554–4567. <https://doi.org/10.1016/j.jksuci.2021.04.010>.

58. Liang, J.; Ma, M. FS-MOEA: A Novel Feature Selection Algorithm for IDSs in Vehicular Networks. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 368–382. <https://doi.org/10.1109/TITS.2020.3011452>.
59. Benkessirat, A.; Benblidia, N. A novel feature selection approach based on constrained eigenvalues optimization. *J. King Saud Univ. -Comput. Inf. Sci.* **2022**, *34*, 4836–4846. <https://doi.org/10.1016/j.jksuci.2021.06.017>.
60. Wang, Y.; Wang, J.; Pal, N.R. Supervised Feature Selection via Collaborative Neurodynamic Optimization. *IEEE Trans. Neural Netw. Learn. Syst.* **2022**, 1–15. <https://doi.org/10.1109/TNNLS.2022.3213167>.
61. Moghaddam, A.A.; Seifi, A.; Niknam, T.; Alizadeh Pahlavani, M.R. Multi-objective operation management of a renewable MG (micro-grid) with back-up micro-turbine/fuel cell/battery hybrid power source. *Energy* **2011**, *36*, 6490–6507. <https://doi.org/10.1016/j.energy.2011.09.017>.
62. Zhang, Y.; Cheng, S.; Shi, Y.; wei Gong, D.; Zhao, X. Cost-sensitive feature selection using two-archive multi-objective artificial bee colony algorithm. *Expert Syst. Appl.* **2019**, *137*, 46–58. <https://doi.org/10.1016/j.eswa.2019.06.044>.
63. Asilian Bidgoli, A.; Ebrahimpour-komleh, H.; Rahnamayan, S. A novel binary many-objective feature selection algorithm for multi-label data classification. *Int. J. Mach. Learn. Cybern.* **2021**, *12*, 2041–2057. <https://doi.org/10.1007/s13042-021-01291-y>.
64. Xu, H.; Xue, B.; Zhang, M. A Duplication Analysis-Based Evolutionary Algorithm for Biobjective Feature Selection. *IEEE Trans. Evol. Comput.* **2021**, *25*, 205–218. <https://doi.org/10.1109/TEVC.2020.3016049>.
65. Han, F.; Chen, W.T.; Ling, Q.H.; Han, H. Multi-objective particle swarm optimization with adaptive strategies for feature selection. *Swarm Evol. Comput.* **2021**, *62*, 100847. <https://doi.org/10.1016/j.swevo.2021.100847>.
66. Niu, B.; Yi, W.; Tan, L.; Geng, S.; Wang, H. A multi-objective feature selection method based on bacterial foraging optimization. *Nat. Comput.* **2021**, *20*, 63–76. <https://doi.org/10.1007/s11047-019-09754-6>.
67. Li, T.; Zhan, Z.H.; Xu, J.C.; Yang, Q.; Ma, Y.Y. A binary individual search strategy-based bi-objective evolutionary algorithm for high-dimensional feature selection. *Inf. Sci.* **2022**, *610*, 651–673. <https://doi.org/10.1016/j.ins.2022.07.183>.
68. Pan, J.S.; Liu, N.; Chu, S.C. A competitive mechanism based multi-objective differential evolution algorithm and its application in feature selection. *Knowl.-Based Syst.* **2022**, *245*, 108582. <https://doi.org/10.1016/j.knosys.2022.108582>.
69. Luo, J.; Zhou, D.; Jiang, L.; Ma, H. A particle swarm optimization based multiobjective memetic algorithm for high-dimensional feature selection. *Memetic Comput.* **2022**, *14*, 77–93. <https://doi.org/10.1007/s12293-022-00354-z>.
70. Hosseini, F.; Gharehchopogh, F.S.; Masdari, M. MOAEOSCA: An enhanced multi-objective hybrid artificial ecosystem-based optimization with sine cosine algorithm for feature selection in botnet detection in IoT. *Multimed. Tools Appl.* **2023**, *82*, 13369–13399. <https://doi.org/10.1007/s11042-022-13836-6>.
71. Al-Tashi, Q.; Abdulkadir, S.J.; Rais, H.M.; Mirjalili, S.; Alhussian, H.; Ragab, M.G.; Alqushaibi, A. Binary Multi-Objective Grey Wolf Optimizer for Feature Selection in Classification. *IEEE Access* **2020**, *8*, 106247–106263. <https://doi.org/10.1109/ACCESS.2020.3000040>.
72. Usman, A.M.; Yusof, U.K.; Naim, S. Filter-Based Multi-Objective Feature Selection Using NSGA III and Cuckoo Optimization Algorithm. *IEEE Access* **2020**, *8*, 76333–76356. <https://doi.org/10.1109/ACCESS.2020.2987057>.
73. Xue, Y.; Zhu, H.; Liang, J.; Słowik, A. Adaptive crossover operator based multi-objective binary genetic algorithm for feature selection in classification. *Knowl.-Based Syst.* **2021**, *227*, 107218. <https://doi.org/10.1016/j.knosys.2021.107218>.
74. Li, W.; Chai, Z.; Tang, Z. A decomposition-based multi-objective immune algorithm for feature selection in learning to rank. *Knowl.-Based Syst.* **2021**, *234*, 107577. <https://doi.org/10.1016/j.knosys.2021.107577>.
75. Li, Y.; Sun, Z.; Liu, X.; Chen, W.T.; Horng, D.J.; Lai, K.K. Feature selection based on a large-scale many-objective evolutionary algorithm. *Comput. Intell. Neurosci.* **2021**, *2021*, 9961727.
76. Ishibuchi, H.; Murata, T. A multi-objective genetic local search algorithm and its application to flowshop scheduling. *IEEE Trans. Syst. Man Cybern. Part (Appl. Rev.)* **1998**, *28*, 392–403. <https://doi.org/10.1109/5326.704576>.
77. Qiu, C. A novel multi-swarm particle swarm optimization for feature selection. *Genet. Program. Evolvable Mach.* **2019**, *20*, 503–529.
78. Too, J.; Abdullah, A.R.; Mohd Saad, N. A New Quadratic Binary Harris Hawk Optimization for Feature Selection. *Electronics* **2019**, *8*, 1130. <https://doi.org/10.3390/electronics8101130>.
79. Hodashinsky, I.A.; Sarin, K.S. Feature selection: Comparative analysis of binary metaheuristics and population based algorithm with adaptive memory. *Program. Comput. Softw.* **2019**, *45*, 221–227.
80. Too, J.; Abdullah, A.R.; Mohd Saad, N. A New Co-Evolution Binary Particle Swarm Optimization with Multiple Inertia Weight Strategy for Feature Selection. *Informatics* **2019**, *6*, 21.
81. Hegazy, A.E.; Makhoulouf, M.; El-Tawel, G.S. Feature selection using chaotic salp swarm algorithm for data classification. *Arab. J. Sci. Eng.* **2019**, *44*, 3801–3816.
82. Arora, S.; Anand, P. Binary butterfly optimization approaches for feature selection. *Expert Syst. Appl.* **2019**, *116*, 147–160. <https://doi.org/10.1016/j.eswa.2018.08.051>.
83. Too, J.; Abdullah, A.R.; Mohd Saad, N. Hybrid binary particle swarm optimization differential evolution-based feature selection for EMG signals classification. *Axioms* **2019**, *8*, 79.
84. Tu, Q.; Chen, X.; Liu, X. Hierarchy Strengthened Grey Wolf Optimizer for Numerical Optimization and Feature Selection. *IEEE Access* **2019**, *7*, 78012–78028. <https://doi.org/10.1109/ACCESS.2019.2921793>.
85. Jia, H.; Li, J.; Song, W.; Peng, X.; Lang, C.; Li, Y. Spotted Hyena Optimization Algorithm With Simulated Annealing for Feature Selection. *IEEE Access* **2019**, *7*, 71943–71962. <https://doi.org/10.1109/ACCESS.2019.2919991>.

86. Jia, H.; Xing, Z.; Song, W. A New Hybrid Seagull Optimization Algorithm for Feature Selection. *IEEE Access* **2019**, *7*, 49614–49631. <https://doi.org/10.1109/ACCESS.2019.2909945>.
87. Al-Tashi, Q.; Abdul Kadir, S.J.; Rais, H.M.; Mirjalili, S.; Alhussian, H. Binary Optimization Using Hybrid Grey Wolf Optimization for Feature Selection. *IEEE Access* **2019**, *7*, 39496–39508. <https://doi.org/10.1109/ACCESS.2019.2906757>.
88. Arora, S.; Singh, H.; Sharma, M.; Sharma, S.; Anand, P. A New Hybrid Algorithm Based on Grey Wolf Optimization and Crow Search Algorithm for Unconstrained Function Optimization and Feature Selection. *IEEE Access* **2019**, *7*, 26343–26361. <https://doi.org/10.1109/ACCESS.2019.2897325>.
89. Awadallah, M.A.; Al-Betar, M.A.; Hammouri, A.I.; Alomari, O.A. Binary JAYA algorithm with adaptive mutation for feature selection. *Arab. J. Sci. Eng.* **2020**, *45*, 10875–10890.
90. Too, J.; Abdullah, A.R. Chaotic atom search optimization for feature selection. *Arab. J. Sci. Eng.* **2020**, *45*, 6063–6079.
91. Baş, E.; Ülker, E. An efficient binary social spider algorithm for feature selection problem. *Expert Syst. Appl.* **2020**, *146*, 113185. <https://doi.org/10.1016/j.eswa.2020.113185>.
92. Hans, R.; Kaur, H. Hybrid binary Sine Cosine Algorithm and Ant Lion Optimization (SCALO) approaches for feature selection problem. *Int. J. Comput. Mater. Sci. Eng.* **2020**, *09*, 1950021. <https://doi.org/10.1142/S2047684119500210>.
93. Zhang, X.; Xu, Y.; Yu, C.; Heidari, A.A.; Li, S.; Chen, H.; Li, C. Gaussian mutational chaotic fruit fly-built optimization and feature selection. *Expert Syst. Appl.* **2020**, *141*, 112976. <https://doi.org/10.1016/j.eswa.2019.112976>.
94. Tawhid, M.A.; Ibrahim, A.M. Feature selection based on rough set approach, wrapper approach, and binary whale optimization algorithm. *Int. J. Mach. Learn. Cybern.* **2020**, *11*, 573–602.
95. Abdel-Basset, M.; El-Shahat, D.; El-henawy, I.; de Albuquerque, V.; Mirjalili, S. A new fusion of grey wolf optimizer algorithm with a two-phase mutation for feature selection. *Expert Syst. Appl.* **2020**, *139*, 112824. <https://doi.org/10.1016/j.eswa.2019.112824>.
96. Nadimi-Shahraki, M.H.; Banaie-Dezfouli, M.; Zamani, H.; Taghian, S.; Mirjalili, S. B-MFO: A binary moth-flame optimization for feature selection from medical datasets. *Computers* **2021**, *10*, 136.
97. Chantar, H.; Tubishat, M.; Essgaer, M.; Mirjalili, S. Hybrid binary dragonfly algorithm with simulated annealing for feature selection. *SN Comput. Sci.* **2021**, *2*, 295.
98. Wang, L.; Gao, Y.; Gao, S.; Yong, X. A new feature selection method based on a self-variant genetic algorithm applied to android malware detection. *Symmetry* **2021**, *13*, 1290.
99. Elgamal, Z.M.; Yasin, N.M.; Sabri, A.Q.M.; Sihwail, R.; Tubishat, M.; Jarrah, H. Improved equilibrium optimization algorithm using elite opposition-based learning and new local search strategy for feature selection in medical datasets. *Computation* **2021**, *9*, 68.
100. Mostert, W.; Malan, K.M.; Engelbrecht, A.P. A feature selection algorithm performance metric for comparative analysis. *Algorithms* **2021**, *14*, 100.
101. Kitonyi, P.M.; Seger, D.R. Hybrid gradient descent grey wolf optimizer for optimal feature selection. *BioMed Res. Int.* **2021**, *2021*, 2555622.
102. Elminaam, D.S.A.; Nabil, A.; Ibraheem, S.A.; Houssein, E.H. An Efficient Marine Predators Algorithm for Feature Selection. *IEEE Access* **2021**, *9*, 60136–60153. <https://doi.org/10.1109/ACCESS.2021.3073261>.
103. Assiri, A.S. On the performance improvement of Butterfly Optimization approaches for global optimization and Feature Selection. *PLoS ONE* **2021**, *16*, e0242612.
104. Liu, W.; Wang, J. Recursive elimination–election algorithms for wrapper feature selection. *Appl. Soft Comput.* **2021**, *113*, 107956. <https://doi.org/10.1016/j.asoc.2021.107956>.
105. Too, J.; Abdullah, A.R. Opposition based competitive grey wolf optimizer for EMG feature selection. *Evol. Intell.* **2021**, *14*, 1691–1705.
106. Mohammadzadeh, H.; Gharehchopogh, F.S. An efficient binary chaotic symbiotic organisms search algorithm approaches for feature selection problems. *J. Supercomput.* **2021**, *77*, 9102–9144.
107. Al-Betar, M.A.; Hammouri, A.I.; Awadallah, M.A.; Abu Doush, I. Binary β -hill climbing optimizer with S-shape transfer function for feature selection. *J. Ambient Intell. Humaniz. Comput.* **2021**, *12*, 7637–7665. <https://doi.org/10.1007/s12652-020-02484-z>.
108. Agrawal, P.; Ganesh, T.; Mohamed, A.W. A novel binary gaining–sharing knowledge-based optimization algorithm for feature selection. *Neural Comput. Appl.* **2021**, *33*, 5989–6008. <https://doi.org/10.1007/s00521-020-05375-8>.
109. Long, W.; Jiao, J.; Liang, X.; Wu, T.; Xu, M.; Cai, S. Pinhole-imaging-based learning butterfly optimization algorithm for global optimization and feature selection. *Appl. Soft Comput.* **2021**, *103*, 107146. <https://doi.org/10.1016/j.asoc.2021.107146>.
110. Chaudhuri, A.; Sahu, T.P. Feature selection using Binary Crow Search Algorithm with time varying flight length. *Expert Syst. Appl.* **2021**, *168*, 114288. <https://doi.org/10.1016/j.eswa.2020.114288>.
111. Jaddi, N.S.; Abdullah, S.; Nazri, M.Z.A. A Recurrence Population-based Great Deluge Algorithm with Independent Quality Estimation for Feature Selection from Academician Data. *Appl. Artif. Intell.* **2021**, *35*, 1081–1105. <https://doi.org/10.1080/08839514.2021.1972253>.
112. Akinola, O.A.; Ezugwu, A.E.; Oyelade, O.N.; Agushaka, J.O. A hybrid binary dwarf mongoose optimization algorithm with simulated annealing for feature selection on high dimensional multi-class datasets. *Sci. Rep.* **2022**, *12*, 14945. <https://doi.org/10.1038/s41598-022-18993-0>.

113. Alsmadi, M.K.; Alzaqebah, M.; Jawarneh, S.; Brini, S.; Al-Marashdeh, I.; Briki, K.; Alrefai, N.; Alghamdi, F.A.; Al-Rashdan, M.T. Cuckoo algorithm with great deluge local-search for feature selection problems. *Int. J. Electr. Comput. Eng. (2088-8708)* **2022**, *12*, 4315–4326.
114. Feng, J.; Kuang, H.; Zhang, L. EBBA: An enhanced binary bat algorithm integrated with chaos theory and lévy flight for feature selection. *Future Internet* **2022**, *14*, 178. <https://doi.org/10.3390/fi14060178>.
115. Hichem, H.; Elkamel, M.; Rafik, M.; Mesaaoud, M.T.; Ouahiba, C. A new binary grasshopper optimization algorithm for feature selection problem. *J. King Saud Univ. -Comput. Inf. Sci.* **2022**, *34*, 316–328. <https://doi.org/10.1016/j.jksuci.2019.11.007>.
116. Zhu, Y.; Li, T.; Li, W. An Efficient Hybrid Feature Selection Method Using the Artificial Immune Algorithm for High-Dimensional Data. *Comput. Intell. Neurosci.* **2022**, *2022*, 1452301. <https://doi.org/10.1155/2022/1452301>.
117. Qin, X.; Zhang, S.; Yin, D.; Chen, D.; Dong, X. Two-stage feature selection for classification of gene expression data based on an improved Salp Swarm Algorithm. *Math. Biosci. Eng.* **2022**, *19*, 13747–13781. <https://doi.org/10.3934/mbe.2022641>.
118. Keleş, M.K.; Kiliç, Ü. Binary Black Widow Optimization Approach for Feature Selection. *IEEE Access* **2022**, *10*, 95936–95948. <https://doi.org/10.1109/ACCESS.2022.3204046>.
119. Singh, H.; Sharma, S.; Khurana, M.; Kaur, M.; Lee, H.N. Binary Drone Squadron Optimization Approaches for Feature Selection. *IEEE Access* **2022**, *10*, 87099–87114. <https://doi.org/10.1109/ACCESS.2022.3192836>.
120. Zhang, Y.; Zhang, Y.; Zhang, C.; Zhou, C. Multiobjective Harris Hawks Optimization With Associative Learning and Chaotic Local Search for Feature Selection. *IEEE Access* **2022**, *10*, 72973–72987. <https://doi.org/10.1109/ACCESS.2022.3189476>.
121. Elgamal, Z.; Sabri, A.Q.M.; Tubishat, M.; Tbaishat, D.; Makhadmeh, S.N.; Alomari, O.A. Improved Reptile Search Optimization Algorithm Using Chaotic Map and Simulated Annealing for Feature Selection in Medical Field. *IEEE Access* **2022**, *10*, 51428–51446. <https://doi.org/10.1109/ACCESS.2022.3174854>.
122. Takieldeem, A.E.; El-kenawy, E.S.M.; Hadwan, M.; Zaki, R.M. Dipper Throated Optimization Algorithm for Unconstrained Function and Feature Selection. *Comput. Mater. Contin.* **2022**, *72*, 1465–1481. <https://doi.org/10.32604/cmc.2022.026026>.
123. Kalra, M.; Kumar, V.; Kaur, M.; Idris, S.A.; Öztürk, Ş.; Alshazly, H. A Novel Binary Emperor Penguin Optimizer for Feature Selection Tasks. *Comput. Mater. Contin.* **2022**, *70*, 6239–6255. <https://doi.org/10.32604/cmc.2022.020682>.
124. Balakrishnan, K.; Dhanalakshmi, R.; Seetharaman, G. S-shaped and V-shaped binary African vulture optimization algorithm for feature selection. *Expert Syst.* **2022**, *39*, e13079. <https://doi.org/10.1111/exsy.13079>.
125. Too, J.; Liang, G.; Chen, H. Memory-based Harris hawk optimization with learning agents: a feature selection approach. *Eng. Comput.* **2022**, *38*, 4457–4478. <https://doi.org/10.1007/s00366-021-01479-4>.
126. Agrawal, U.; Rohatgi, V.; Katarya, R. Normalized Mutual Information-based equilibrium optimizer with chaotic maps for wrapper-filter feature selection. *Expert Syst. Appl.* **2022**, *207*, 118107. <https://doi.org/10.1016/j.eswa.2022.118107>.
127. Isuwa, J.; Abdullahi, M.; Sahabi Ali, Y.; Abdulrahim, A. Hybrid particle swarm optimization with sequential one point flipping algorithm for feature selection. *Concurr. Comput. Pract. Exp.* **2022**, *34*, e7239. <https://doi.org/10.1002/cpe.7239>.
128. Preeti.; Deep, K. A random walk Grey wolf optimizer based on dispersion factor for feature selection on chronic disease prediction. *Expert Syst. Appl.* **2022**, *206*, 117864. <https://doi.org/10.1016/j.eswa.2022.117864>.
129. Khosravi, H.; Amiri, B.; Yazdanjue, N.; Babaiyan, V. An improved group teaching optimization algorithm based on local search and chaotic map for feature selection in high-dimensional data. *Expert Syst. Appl.* **2022**, *204*, 117493. <https://doi.org/10.1016/j.eswa.2022.117493>.
130. Samieiyan, B.; MohammadiNasab, P.; Mollaei, M.A.; Hajizadeh, F.; Kangavari, M. Novel optimized crow search algorithm for feature selection. *Expert Syst. Appl.* **2022**, *204*, 117486. <https://doi.org/10.1016/j.eswa.2022.117486>.
131. Beheshti, Z. BMPA-TVsinV: A Binary Marine Predators Algorithm using time-varying sine and V-shaped transfer functions for wrapper-based feature selection. *Knowl.-Based Syst.* **2022**, *252*, 109446. <https://doi.org/10.1016/j.knosys.2022.109446>.
132. Asghari Varzaneh, Z.; Hossein, S.; Ebrahimi Mood, S.; Javidi, M.M. A new hybrid feature selection based on Improved Equilibrium Optimization. *Chemom. Intell. Lab. Syst.* **2022**, *228*, 104618. <https://doi.org/10.1016/j.chemolab.2022.104618>.
133. Nadimi-Shahraki, M.H.; Zamani, H.; Mirjalili, S. Enhanced whale optimization algorithm for medical feature selection: A COVID-19 case study. *Comput. Biol. Med.* **2022**, *148*, 105858. <https://doi.org/10.1016/j.compbiomed.2022.105858>.
134. Hu, J.; Heidari, A.A.; Zhang, L.; Xue, X.; Gui, W.; Chen, H.; Pan, Z. Chaotic diffusion-limited aggregation enhanced grey wolf optimizer: Insights, analysis, binarization, and feature selection. *Int. J. Intell. Syst.* **2022**, *37*, 4864–4927. <https://doi.org/10.1002/int.22744>.
135. Bacanin, N.; Bezdan, T.; Al-Turjman, F.; Rashid, T.A. Artificial Flora Optimization Algorithm with Genetically Guided Operators for Feature Selection and Neural Network Training. *Int. J. Fuzzy Syst.* **2022**, *24*, 2538–2559. <https://doi.org/10.1007/s40815-021-01191-x>.
136. Bezdan, T.; Zivkovic, M.; Bacanin, N.; Chhabra, A.; Suresh, M. Feature Selection by Hybrid Brain Storm Optimization Algorithm for COVID-19 Classification. *J. Comput. Biol.* **2022**, *29*, 515–529. <https://doi.org/10.1089/cmb.2021.0256>.
137. Qaraad, M.; Amjad, S.; Hussein, N.K.; Elhosseini, M.A. Large scale salp-based grey wolf optimization for feature selection and global optimization. *Neural Comput. Appl.* **2022**, *34*, 8989–9014. <https://doi.org/10.1007/s00521-022-06921-2>.
138. Thaher, T.; Chantar, H.; Too, J.; Mafarja, M.; Turabieh, H.; Houssein, E.H. Boolean Particle Swarm Optimization with various Evolutionary Population Dynamics approaches for feature selection problems. *Expert Syst. Appl.* **2022**, *195*, 116550. <https://doi.org/10.1016/j.eswa.2022.116550>.

139. Hu, G.; Du, B.; Wang, X.; Wei, G. An enhanced black widow optimization algorithm for feature selection. *Knowl.-Based Syst.* **2022**, *235*, 107638. <https://doi.org/10.1016/j.knsys.2021.107638>.
140. Wang, X.; Dong, X.; Zhang, Y.; Chen, H. Crisscross Harris Hawks Optimizer for Global Tasks and Feature Selection. *J. Bionic Eng.* **2023**, *20*, 1153–1174. <https://doi.org/10.1007/s42235-022-00298-7>.
141. Balakrishnan, K.; Dhanalakshmi, R.; Akila, M.; Sinha, B.B. Improved equilibrium optimization based on Levy flight approach for feature selection. *Evol. Syst.* **2023**, *14*, 735–746. <https://doi.org/10.1007/s12530-022-09461-1>.
142. Agrawal, P.; Ganesh, T.; Oliva, D.; Mohamed, A.W. S-shaped and V-shaped gaining-sharing knowledge-based algorithm for feature selection. *Appl. Intell.* **2022**, *52*, 81–112. <https://doi.org/10.1007/s10489-021-02233-5>.
143. Chhabra, A.; Hussien, A.G.; Hashim, F.A. Improved bald eagle search algorithm for global optimization and feature selection. *Alex. Eng. J.* **2023**, *68*, 141–180. <https://doi.org/10.1016/j.aej.2022.12.045>.
144. Oyelade, O.N.; Agushaka, J.O.; Ezugwu, A.E. Evolutionary binary feature selection using adaptive ebola optimization search algorithm for high-dimensional datasets. *PLoS ONE* **2023**, *18*, e0282812. <https://doi.org/10.1371/journal.pone.0282812>.
145. Ewees, A.A.; Ismail, F.H.; Sahlol, A.T. Gradient-based optimizer improved by Slime Mould Algorithm for global optimization and feature selection for diverse computation problems. *Expert Syst. Appl.* **2023**, *213*, 118872. <https://doi.org/10.1016/j.eswa.2022.118872>.
146. Devi, R.M.; Premkumar, M.; Kiruthiga, G.; Sowmya, R. IGJO: An Improved Golden Jackel Optimization Algorithm Using Local Escaping Operator for Feature Selection Problems. *Neural Process. Lett.* **2023**, *55*, 6443–6531. <https://doi.org/10.1007/s11063-023-11146-y>.
147. Yu, W.; Kang, H.; Sun, G.; Liang, S.; Li, J. Bio-Inspired Feature Selection in Brain Disease Detection via an Improved Sparrow Search Algorithm. *IEEE Trans. Instrum. Meas.* **2023**, *72*, 1–15. <https://doi.org/10.1109/TIM.2022.3228003>.
148. Yong, X.; Gao, Y.I. Improved firefly algorithm for feature selection with the ReliefF-based initialization and the weighted voting mechanism. *Neural Comput. Appl.* **2023**, *35*, 275–301. <https://doi.org/10.1007/s00521-022-07755-8>.
149. Shaddeli, A.; Gharehchopogh, F.S.; Masdari, M.; Solouk, V. BFRA: A New Binary Hyper-Heuristics Feature Ranks Algorithm for Feature Selection in High-Dimensional Classification Data. *Int. J. Inf. Technol. Decis. Mak.* **2023**, *22*, 471–536. <https://doi.org/10.1142/S0219622022500432>.
150. Tanha, J.; Zarei, Z. The Bombus-terrestris bee optimization algorithm for feature selection. *Appl. Intell.* **2023**, *53*, 470–490. <https://doi.org/10.1007/s10489-022-03478-4>.
151. Tubishat, M.; Alswaitti, M.; Mirjalili, S.; Al-Garadi, M.A.; Alrashdan, M.T.; Rana, T.A. Dynamic Butterfly Optimization Algorithm for Feature Selection. *IEEE Access* **2020**, *8*, 194303–194314. <https://doi.org/10.1109/ACCESS.2020.3033757>.
152. Zhong, C.; Chen, Y.; Peng, J. Feature Selection Based on a Novel Improved Tree Growth Algorithm. *Int. J. Comput. Intell. Syst.* **2020**, *13*, 247–258. <https://doi.org/10.2991/ijcis.d.200219.001>.
153. Ji, B.; Lu, X.; Sun, G.; Zhang, W.; Li, J.; Xiao, Y. Bio-Inspired Feature Selection: An Improved Binary Particle Swarm Optimization Approach. *IEEE Access* **2020**, *8*, 85989–86002. <https://doi.org/10.1109/ACCESS.2020.2992752>.
154. Bhattacharyya, T.; Chatterjee, B.; Singh, P.K.; Yoon, J.H.; Geem, Z.W.; Sarkar, R. Mayfly in Harmony: A New Hybrid Meta-Heuristic Feature Selection Algorithm. *IEEE Access* **2020**, *8*, 195929–195945. <https://doi.org/10.1109/ACCESS.2020.3031718>.
155. Tawhid, M.A.; Dsouza, K.B. Hybrid binary dragonfly enhanced particle swarm optimization algorithm for solving feature selection problems. *Math. Found. Comput.* **2018**, *1*, 181–200. <https://doi.org/10.3934/mfc.2018009>.
156. Pichai, S.; Sunat, K.; Chiewchanwattana, S. An Asymmetric Chaotic Competitive Swarm Optimization Algorithm for Feature Selection in High-Dimensional Data. *Symmetry* **2020**, *12*, 1782. <https://doi.org/10.3390/sym12111782>.
157. El-Kenawy, E.S.M.; Eid, M.M.; Saber, M.; Ibrahim, A. MbGWO-SFS: Modified Binary Grey Wolf Optimizer Based on Stochastic Fractal Search for Feature Selection. *IEEE Access* **2020**, *8*, 107635–107649. <https://doi.org/10.1109/ACCESS.2020.3001151>.
158. Li, J.; Kang, H.; Sun, G.; Feng, T.; Li, W.; Zhang, W.; Ji, B. IBDA: Improved Binary Dragonfly Algorithm With Evolutionary Population Dynamics and Adaptive Crossover for Feature Selection. *IEEE Access* **2020**, *8*, 108032–108051. <https://doi.org/10.1109/ACCESS.2020.3001204>.
159. Elgamal, Z.M.; Yasin, N.B.M.; Tubishat, M.; Alswaitti, M.; Mirjalili, S. An Improved Harris Hawks Optimization Algorithm With Simulated Annealing for Feature Selection in the Medical Field. *IEEE Access* **2020**, *8*, 186638–186652. <https://doi.org/10.1109/ACCESS.2020.3029728>.
160. Wu, H.; Du, S.; Zhang, Y.; Zhang, Q.; Duan, K.; Lin, Y. Threshold Binary Grey Wolf Optimizer Based on Multi-Elite Interaction for Feature Selection. *IEEE Access* **2023**, *11*, 34332–34348. <https://doi.org/10.1109/ACCESS.2023.3263584>.
161. Adamu, A.; Abdullahi, M.; Junaidu, S.B.; Hassan, I.H. An hybrid particle swarm optimization with crow search algorithm for feature selection. *Mach. Learn. Appl.* **2021**, *6*, 100108. <https://doi.org/10.1016/j.mlwa.2021.100108>.
162. Ewees, A.A.; El Aziz, M.A.; Hassanien, A.E. Chaotic multi-verse optimizer-based feature selection. *Neural Comput. Appl.* **2019**, *31*, 991–1006.
163. Sayed, G.I.; Hassanien, A.E.; Azar, A.T. Feature selection via a novel chaotic crow search algorithm. *Neural Comput. Appl.* **2019**, *31*, 171–188. <https://doi.org/10.1007/s00521-017-2988-6>.
164. Sayed, G.I.; Tharwat, A.; Hassanien, A.E. Chaotic dragonfly algorithm: an improved metaheuristic algorithm for feature selection. *Appl. Intell.* **2019**, *49*, 188–205. <https://doi.org/10.1007/s10489-018-1261-8>.
165. Manita, G.; Korbaa, O. Binary Political Optimizer for Feature Selection Using Gene Expression Data. *Comput. Intell. Neurosci.* **2020**, *2020*, 8896570. <https://doi.org/10.1155/2020/8896570>.

166. Zhu, L.; He, S.; Wang, L.; Zeng, W.; Yang, J. Feature Selection Using an Improved Gravitational Search Algorithm. *IEEE Access* **2019**, *7*, 114440–114448. <https://doi.org/10.1109/ACCESS.2019.2935833>.
167. Ibrahim, R.A.; Abd Elaziz, M.; Oliva, D.; Lu, S. An improved runner-root algorithm for solving feature selection problems based on rough sets and neighborhood rough sets. *Appl. Soft Comput.* **2020**, *97*, 105517. <https://doi.org/10.1016/j.asoc.2019.105517>.
168. Ding, Y.; Zhou, K.; Bi, W. Feature selection based on hybridization of genetic algorithm and competitive swarm optimizer. *Soft Comput.* **2020**, *24*, 11663–11672.
169. Mandal, A.K.; Sen, R.; Chakraborty, B. Feature selection in classification using self-adaptive owl search optimization algorithm with elitism and mutation strategies. *J. Intell. Fuzzy Syst.* **2021**, *40*, 535–550. <https://doi.org/10.3233/JIFS-200258>.
170. Shen, C.; Zhang, K. Two-stage improved Grey Wolf optimization algorithm for feature selection on high-dimensional classification. *Complex Intell. Syst.* **2022**, *8*, 2769–2789. <https://doi.org/10.1007/s40747-021-00452-4>.
171. Kundu, R.; Chattopadhyay, S.; Cuevas, E.; Sarkar, R. AltWOA: Altruistic Whale Optimization Algorithm for feature selection on microarray datasets. *Comput. Biol. Med.* **2022**, *144*, 105349.
172. Pashaei, E.; Pashaei, E. An efficient binary chimp optimization algorithm for feature selection in biomedical data classification. *Neural Comput. Appl.* **2022**, *34*, 6427–6451. <https://doi.org/10.1007/s00521-021-06775-0>.
173. Seger, D.; Mbuthia, M.; Nyete, A. An Innovative Excited-ACS-IDGWO Algorithm for Optimal Biomedical Data Feature Selection. *BioMed Res. Int.* **2020**, *2020*, 8506365. <https://doi.org/10.1155/2020/8506365>.
174. Xue, Y.; Tang, T.; Pang, W.; Liu, A.X. Self-adaptive parameter and strategy based particle swarm optimization for large-scale feature selection problems with multiple classifiers. *Appl. Soft Comput.* **2020**, *88*, 106031. <https://doi.org/10.1016/j.asoc.2019.106031>.
175. Ahn, G.; Hur, S. Efficient genetic algorithm for feature selection for early time series classification. *Comput. Ind. Eng.* **2020**, *142*, 106345. <https://doi.org/10.1016/j.cie.2020.106345>.
176. Sadeghian, Z.; Akbari, E.; Nematzadeh, H. A hybrid feature selection method based on information theory and binary butterfly optimization algorithm. *Eng. Appl. Artif. Intell.* **2021**, *97*, 104079. <https://doi.org/10.1016/j.engappai.2020.104079>.
177. Adel, A.; Omar, N.; Abdullah, S.; Al-Shabi, A. Co-Operative Binary Bat Optimizer with Rough Set Reducts for Text Feature Selection. *Appl. Sci.* **2022**, *12*, 11296. <https://doi.org/10.3390/app122111296>.
178. Anuprathibha, T.; Kanimozhiselvi, C. Penguin search optimization based feature selection for automated opinion mining. *Int. J. Recent Technol. Eng* **2019**, *8*, 648–653.
179. Sharafi, Y.; Teshnehlab, M. Opposition-based binary competitive optimization algorithm using time-varying V-shape transfer function for feature selection. *Neural Comput. Appl.* **2021**, *33*, 17497–17533.
180. Long, W.; Jiao, J.; Wu, T.; Xu, M.; Cai, S. A balanced butterfly optimization algorithm for numerical optimization and feature selection. *Soft Comput.* **2022**, *26*, 11505–11523. <https://doi.org/10.1007/s00500-022-07389-x>.
181. Bhadra, T.; Maulik, U. Unsupervised Feature Selection Using Iterative Shrinking and Expansion Algorithm. *IEEE Trans. Emerg. Top. Comput. Intell.* **2022**, *6*, 1453–1462. <https://doi.org/10.1109/TETCI.2022.3199704>.
182. Hashemi, A.; Joodaki, M.; Joodaki, N.Z.; Dowlatshahi, M.B. Ant colony optimization equipped with an ensemble of heuristics through multi-criteria decision making: A case study in ensemble feature selection. *Appl. Soft Comput.* **2022**, *124*, 109046. <https://doi.org/10.1016/j.asoc.2022.109046>.
183. Hussien, A.G.; Amin, M. A self-adaptive Harris Hawks optimization algorithm with opposition-based learning and chaotic local search strategy for global optimization and feature selection. *Int. J. Mach. Learn. Cybern.* **2022**, *13*, 309–336. <https://doi.org/10.1007/s13042-021-01326-4>.
184. Karimi, F.; Dowlatshahi, M.B.; Hashemi, A. SemiACO: A semi-supervised feature selection based on ant colony optimization. *Expert Syst. Appl.* **2023**, *214*, 119130. <https://doi.org/10.1016/j.eswa.2022.119130>.
185. Durgam, R.; Devarakonda, N. A Quasi-Oppositional Based Flamingo Search Algorithm Integrated with Generalized Ring Crossover for Effective Feature Selection. *IETE J. Res.* **2023**, *0*, 1–17. <https://doi.org/10.1080/03772063.2023.2185302>.
186. Azar, A.T.; Khan, Z.I.; Amin, S.U.; Fouad, K.M. Hybrid Global Optimization Algorithm for Feature Selection. *Comput. Mater. Contin.* **2023**, *74*, 2021–2037. <https://doi.org/10.32604/cmc.2023.032183>.
187. Li, Z.; Du, J.; Nie, B.; Xiong, W.; Xu, G.; Luo, J. A new two-stage hybrid feature selection algorithm and its application in Chinese medicine. *Int. J. Mach. Learn. Cybern.* **2022**, *13*, 1243–1264. <https://doi.org/10.1007/s13042-021-01445-y>.
188. Murphy, K.P. *Machine Learning: A Probabilistic Perspective*; MIT Press: Cambridge, MA, USA, 2012.
189. Witten, D.; James, G. *An Introduction to Statistical Learning with Applications in R*; Springer: Berlin/Heidelberg, Germany, 2013.
190. Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*; MIT Press: Cambridge, MA, USA, 2016.
191. Alpaydin, E. *Introduction to Machine Learning*; MIT Press: Cambridge, MA, USA, 2020.
192. Bishop, C.M.; Nasrabadi, N.M. *Pattern Recognition and Machine Learning*; Springer: Berlin/Heidelberg, Germany, 2006.
193. Song, H.; Triguero, I.; Özcan, E. A review on the self and dual interactions between machine learning and optimisation. *Prog. Artif. Intell.* **2019**, *8*, 143–165. <https://doi.org/10.1007/s13748-019-00185-z>.
194. Kennedy, J.; Eberhart, R. Particle swarm optimization. In Proceedings of ICNN'95 - International Conference on Neural Networks, Perth, WA, Australia, 27 November–1 December 1995; Volume 4, pp. 1942–1948. <https://doi.org/10.1109/ICNN.1995.488968>.
195. Mirjalili, S.; Mirjalili, S.M.; Lewis, A. Grey Wolf Optimizer. *Adv. Eng. Softw.* **2014**, *69*, 46–61. <https://doi.org/10.1016/j.advengsoft.2013.12.007>.
196. Holland, J.H. Genetic algorithms. *Sci. Am.* **1992**, *267*, 66–73.

197. Crawford, B.; Soto, R.; Astorga, G.; García, J.; Castro, C.; Paredes, F.; et al. Putting continuous metaheuristics to work in binary search spaces. *Complexity* **2017**, *2017*, 8404231. <https://doi.org/10.1155/2017/8404231>.
198. Coello Coello, C.A. Evolutionary algorithms for solving multi-objective problems. *Genet. Program. Evolvable Mach.* **2007**, *8*, 221–252.
199. Li, J.; Cheng, K.; Wang, S.; Morstatter, F.; Trevino, R.P.; Tang, J.; Liu, H. Feature selection: A data perspective. *ACM Comput. Surv. (CSUR)* **2018**, *50*, 94.
200. Cios, K.; Kurgan, L.; Goodenday, L. SPECT Heart. UCI Machine Learning Repository. 2001. Available online: <https://doi.org/10.24432/C5P304> (accessed on 21 December 2023).
201. Ilter, N.; Guvenir, H. Dermatology. UCI Machine Learning Repository. 1998. Available online: <https://doi.org/10.24432/C5FK5P> (accessed on 21 December 2023).
202. Chapman, D.; Jain, A. Musk (Version 1). UCI Machine Learning Repository. 1994. Available online: <https://doi.org/10.24432/C5ZK5B> (accessed on 21 December 2023).
203. Wolberg, W.; Street, W.N.; Mangasarian, O. *Breast Cancer Diagnosis and Prognosis via Linear Programming*; Technical Report; University of Wisconsin–Madison: Madison, Wisconsin, 1994.
204. Wolberg, W. Breast Cancer Wisconsin (Original). UCI Machine Learning Repository. 1992. Available online: <https://doi.org/10.24432/C5HP4Z> (accessed on 21 December 2023).
205. Hong, Z.; Yang, J. Lung Cancer. UCI Machine Learning Repository. 1992. Available online: <https://doi.org/10.24432/C57596> (accessed on 21 December 2023).
206. Aeberhard, S.; Forina, M. Wine. UCI Machine Learning Repository. 1991. Available online: <https://doi.org/10.24432/C5PC7J> (accessed on 21 December 2023).
207. Aha, D. Tic-Tac-Toe Endgame. UCI Machine Learning Repository. 1991. Available online: <https://doi.org/10.24432/C5688J> (accessed on 21 December 2023).
208. Forsyth, R. Zoo. UCI Machine Learning Repository. 1990. Available online: <https://doi.org/10.24432/C5R59V> (accessed on 21 December 2023).
209. Sigillito, V.G.; Wing, S.P.; Hutton, L.V.; Baker, K.B. Classification of radar returns from the ionosphere using neural networks. *Johns Hopkins APL Tech. Dig.* **1989**, *10*, 262–266.
210. Shapiro, A. Chess (King-Rook vs. King-Pawn). UCI Machine Learning Repository. 1989. Available online: <https://doi.org/10.24432/C5DK5C> (accessed on 21 December 2023).
211. Breiman, L.; Stone, C. Waveform Database Generator (Version 2). UCI Machine Learning Repository. 1988. Available online: <https://doi.org/10.24432/C56014> (accessed on 21 December 2023).
212. Zwitter, M.; Soklic, M. Lymphography. UCI Machine Learning Repository. 1988. Available online: <https://doi.org/10.24432/C54598> (accessed on 21 December 2023).
213. Congressional Voting Records. UCI Machine Learning Repository. 1987. Available online: <https://doi.org/10.24432/C5C01P> (accessed on 21 December 2023).
214. Sejnowski, T.; Gorman, R. Connectionist Bench (Sonar, Mines vs. Rocks). UCI Machine Learning Repository. Available online: <https://doi.org/10.24432/C5T01Q> (accessed on 21 December 2023).
215. Statlog (Heart). UCI Machine Learning Repository. Available online: <https://doi.org/10.24432/C57303> (accessed on 21 December 2023).
216. Lemus-Romani, J.; Becerra-Rozas, M.; Crawford, B.; Soto, R.; Cisternas-Caneo, F.; Vega, E.; Castillo, M.; Tapia, D.; Astorga, G.; Palma, W.; et al. A Novel Learning-Based Binarization Scheme Selector for Swarm Algorithms Solving Combinatorial Problems. *Mathematics* **2021**, *9*, 2887. <https://doi.org/10.3390/math922887>.
217. Crawford, B.; Soto, R.; Lemus-Romani, J.; Becerra-Rozas, M.; Lanza-Gutiérrez, J.M.; Caballé, N.; Castillo, M.; Tapia, D.; Cisternas-Caneo, F.; García, J.; et al. Q-Learnheuristics: Towards Data-Driven Balanced Metaheuristics. *Mathematics* **2021**, *9*, 1839. <https://doi.org/10.3390/math9161839>.
218. Becerra-Rozas, M.; Lemus-Romani, J.; Cisternas-Caneo, F.; Crawford, B.; Soto, R.; García, J. Swarm-Inspired Computing to Solve Binary Optimization Problems: A Backward Q-Learning Binarization Scheme Selector. *Mathematics* **2022**, *10*, 4776. <https://doi.org/10.3390/math10244776>.
219. Becerra-Rozas, M.; Cisternas-Caneo, F.; Crawford, B.; Soto, R.; García, J.; Astorga, G.; Palma, W. Embedded Learning Approaches in the Whale Optimizer to Solve Coverage Combinatorial Problems. *Mathematics* **2022**, *10*, 4529. <https://doi.org/10.3390/math10234529>.
220. Lemus-Romani, J.; Crawford, B.; Cisternas-Caneo, F.; Soto, R.; Becerra-Rozas, M. Binarization of Metaheuristics: Is the Transfer Function Really Important? *Biomimetics* **2023**, *8*, 400. <https://doi.org/10.3390/biomimetics8050400>.
221. García, J.; Moraga, P.; Crawford, B.; Soto, R.; Pinto, H. Binarization Technique Comparisons of Swarm Intelligence Algorithm: An Application to the Multi-Demand Multidimensional Knapsack Problem. *Mathematics* **2022**, *10*, 3183. <https://doi.org/10.3390/math10173183>.
222. García, J.; Leiva-Araos, A.; Crawford, B.; Soto, R.; Pinto, H. Exploring Initialization Strategies for Metaheuristic Optimization: Case Study of the Set-Union Knapsack Problem. *Mathematics* **2023**, *11*, 2695. <https://doi.org/10.3390/math11122695>.

-
223. Figueroa-Torrez, P.; Durán, O.; Crawford, B.; Cisternas-Caneo, F. A Binary Black Widow Optimization Algorithm for Addressing the Cell Formation Problem Involving Alternative Routes and Machine Reliability. *Mathematics* **2023**, *11*, 3475. <https://doi.org/10.3390/math11163475>.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.