Niğde Ömer Halisdemir Üniversitesi Mühendislik Bilimleri Dergisi
Niğde Ömer Halisdemir University Journal of Engineering Sciences

Araştırma makalesi / Research article

www.dergipark.org.tr/tr/pub/ngumuh / www.dergipark.org.tr/en/pub/ngumuh

# Enhancing the performance of machine learning models through Meta-Heuristic Hyperparameter tuning: A comparison for breast cancer prediction

## Makine öğrenmesi modellerinin performansının Meta-Sezgisel Hiperparametre ayarlaması yoluyla Geliştirilmesi: Meme kanseri tahmini için bir karşılaştırma

**Harun Akbulut**[1,*]

[1] *Kayseri University, Department of Artificial Intelligence and Machine Learning, 38280, Kayseri, Türkiye*

**Abstract**

Breast cancer is one of the leading causes of mortality and morbidity among women. Early diagnosis and timely intervention significantly improve treatment outcomes and contribute to maintaining patients' quality of life. This study investigates metaheuristic hyperparameter optimization (HPO) techniques to enhance the performance of machine learning models used for breast cancer prediction. Five different machine learning models were evaluated on a dataset consisting of 569 samples and 30 features. In addition, the performance of ten widely used metaheuristic algorithms reported in the literature was comparatively analyzed. The statistical significance of the performance differences among the algorithms was first assessed using the Kruskal–Wallis test. Subsequently, pairwise comparisons were conducted using the Mann–Whitney U test with Bonferroni correction, taking the SFOA algorithm as the reference method. The initial classification accuracy improved by approximately 2.6% following the application of metaheuristic optimization. Furthermore, a comprehensive evaluation was performed using multiple performance metrics, including accuracy, precision, recall, F1-score, confusion matrix, Cohen's kappa, and log loss. The findings demonstrate that metaheuristic methods provide an effective and reliable approach to HPO. Moreover, while the SFOA algorithm achieved statistically significantly higher performance than some competing methods, it attained a level of predictive accuracy comparable to that of other strong-performing algorithms.

**Keywords:** Breast cancer prediction, Machine learning algorithms, Meta-heuristic hyperparameter optimization

**Öz**

Meme kanseri, kadınlar arasında mortalite ve morbiditenin başlıca nedenlerinden biridir. Erken teşhis ve müdahale, tedavi sürecinin başarısını artırmakta ve hastaların yaşam kalitesinin korunmasına önemli katkı sağlamaktadır. Bu çalışma, meme kanseri tahmininde kullanılan makine öğrenmesi modellerinin performansını artırmak amacıyla meta-sezgisel hiperparametre optimizasyon tekniklerini incelemektedir. Beş farklı makine öğrenmesi modeli, 569 örnek ve 30 özellikten oluşan bir veri seti üzerinde değerlendirilmiş; ayrıca literatürde yaygın olarak kullanılan on farklı meta-sezgisel algoritmanın performansları karşılaştırmalı olarak analiz edilmiştir. Algoritmalar arasındaki performans farklılıklarının istatistiksel anlamlılığı, öncelikle Kruskal–Wallis testi ile değerlendirilmiş, ardından SFOA algoritması referans alınarak Mann–Whitney U testi ve Bonferroni düzeltmeli çoklu karşılaştırmalar ile ikili analizler gerçekleştirilmiştir. Başlangıçta elde edilen doğruluk oranları, uygulanan meta-sezgisel optimizasyon süreçleri sayesinde ortalama yaklaşık %2.6 oranında artış göstermiştir. Çalışmada ayrıca doğruluk, kesinlik, duyarlılık, F1 skoru, karmaşıklık matrisi, Cohen's Kappa ve log loss gibi performans ölçütleri kullanılarak çok boyutlu bir değerlendirme yapılmıştır. Elde edilen bulgular, meta-sezgisel yöntemlerin hiperparametre optimizasyonunda etkili ve güvenilir bir yaklaşım sunduğunu, SFOA algoritmasının ise bazı rakip yöntemlere göre istatistiksel olarak anlamlı düzeyde daha yüksek performans sergilerken diğer güçlü yöntemlerle benzer doğruluk düzeyine ulaştığını göstermektedir.

**Anahtar kelimeler:** Meme kanseri tahmini, Makine öğrenmesi algoritmaları, Meta-sezgisel hiperparametre optimizasyonu

## 1 Introduction

Breast cancer (BC) remains the most prevalent malignancy diagnosed among women globally and constitutes one of the leading causes of cancer-related mortality, particularly when diagnosis is delayed. The World Health Organization (WHO) reports millions of new BC diagnoses annually, with a significant proportion of these cases presenting at advanced stages, leading to high mortality rates [1]. Early detection and precise risk prediction are paramount in mitigating mortality, as timely intervention can significantly enhance patient outcomes. However, conventional BC screening modalities, particularly mammography, are encumbered with several limitations, including high false-negative rates, particularly

* Sorumlu yazar / Corresponding author, e-posta / e-mail: harunakbulut@kayseri.edu.tr (H. Akbulut)
Geliş / Received: 20.04.2026 Kabul / Accepted: 24.08.2026 Yayımlanma / Published: 17.09.2026
doi: 10.28948/ngumuh.1934329

in women with dense breast tissue, and reduced sensitivity in detecting small or indeterminate lesions [2].

In recent years, artificial intelligence methodologies have found widespread application in various domains, including BC diagnosis. Consequently, advancements in machine learning (ML) and deep learning have emerged as promising tools for early detection and risk prediction of BC. These computational approaches leverage large datasets to identify intricate patterns and correlations, offering superior accuracy and predictive power compared to traditional epidemiological methods [3]. Imaging-based models, in particular, demonstrate potential not only in detecting BC at earlier stages but also in predicting future risks based on mammographic images and other diagnostic data [4]. Although deep learning models, especially convolutional neural networks (CNNs), have outperformed classical models, their clinical implementation is hindered by challenges related to interpretability, generalizability, and data dependency—issues that remain crucial in medical applications. Nevertheless, the generalizability of these methods across disparate datasets and healthcare settings remains uncertain.

Given the limitations of deep learning models, such as their requirement for large datasets, significant computational resources, and the associated costs, traditional machine learning methods continue to play a critical role in BC diagnosis, particularly in the context of smaller datasets. Algorithms such as Random Forest (RF), Gradient Boosting (GB), and SVM have been successfully applied to BC detection, especially when more complex deep learning models are not warranted due to data constraints [5, 6]. However, as datasets grow larger and more complex, deep learning models generally surpass classical methods, primarily due to their ability to capture nuanced patterns and relationships in large-scale data. Hybrid approaches, which combine the strengths of both classical and deep learning techniques, have demonstrated significant promise in enhancing predictive performance. Several recent studies have applied HPO to the WDBC dataset. Edsa et al. employed Grey Wolf Optimizer with a cost-sensitive neural network and reported accuracy values of up to 98.6% on this dataset. [7].

Although recent studies have explored multi-modal and multi-source approaches to BC prediction, the present work focuses on the widely benchmarked WDBC dataset, which provides a controlled and reproducible basis for comparing metaheuristic HPO strategies. Das et al. applied a multi-stage machine learning pipeline and achieved 97.4% accuracy. The present work extends these efforts by systematically comparing ten distinct metaheuristic algorithms across five different classifiers, providing a broader comparative baseline than previously available in the literature [8].

A significant challenge associated with deep learning methods is their inherent "black-box" nature, which hampers transparency and trust in clinical settings. Healthcare providers are often reluctant to rely on models whose decision-making processes are not easily interpretable. Recent advancements in explainable artificial intelligence seek to address this concern by providing transparency to complex models. The interpretability of these models is essential not only for their practical application but also for understanding their limitations, biases, and potential ethical issues. Beyond interpretability, the real-world clinical value of AI-assisted decision support has also been demonstrated empirically: in a prospective multicenter cohort study within a national BC screening program, AI-assisted computer-aided detection (AI-CAD) significantly improved radiologists' cancer detection rate in a single-read setting without increasing recall rates [9]. This study therefore focuses on evaluating the classification performance of traditional machine learning models on tabular BC data, with the aim of identifying how metaheuristic HPO affects both discriminative accuracy and probability calibration.

In conclusion, machine learning and deep learning algorithms have revolutionized BC diagnosis, offering substantial improvements in predictive accuracy over traditional methods. Deep learning models, particularly CNNs, have demonstrated superior performance across diverse clinical settings. Classical machine learning algorithms such as extreme gradient boosting (XGBoost), AdaBoost, GB, Decision Trees (DT), and K-Nearest Neighbors (KNN) continue to offer competitive performance in structured clinical datasets, particularly when dataset sizes are moderate and computational resources are limited. A critical factor affecting the performance of these models is the selection of appropriate hyperparameters, which are typically set prior to training and directly influence model capacity, regularization, and convergence behavior. Manual tuning of these parameters is both time-consuming and suboptimal; metaheuristic optimization algorithms offer a principled alternative. Despite the growing interest in this area, systematic comparisons of multiple metaheuristic strategies applied to the same machine learning models on the WDBC dataset remain limited in the literature.

In a study conducted by Zhang et al. [10], the performance of models such as SVM, Logistic Regression, RF, and KNN was compared. The study observed that SVM achieved the highest accuracy with a rate of 97.95%. On the other hand, Senthilkumar et al. [11] employed ensemble learning methods, including AdaBoost, GB, Extra Trees, Bagging, and RF, with AdaBoost attaining the highest accuracy of 95.45%. This study aims to experimentally compare the discriminative performance of established machine learning models under meta-heuristic HPO, while also considering their transparency and interpretability. As the experiments are restricted to a single, well-established benchmark dataset (WDBC) without external validation, claims regarding real-world clinical applicability are intentionally left outside the scope of this study and are discussed as a direction for future work. The contributions of this study to the literature are as follows:

- The performance of XGBoost, AdaBoost, GB, DT, and KNN algorithms used in BC prediction is presented through a comparative analysis.
- Experiments were conducted on a well-established, publicly available clinical dataset derived from real patient samples (WDBC), enabling direct comparability with prior studies.

- The application of meta-heuristic-based HPO techniques in BC prediction is discussed.
- The HPO process has been improved using meta-heuristic algorithms that have gained prominence in the literature over the past five years and basic meta-heuristic algorithms.
- The performance of the algorithms is examined in detail using various performance metrics.
- In previous studies, only a single meta-heuristic algorithm was used in HPO, whereas this study compares multiple meta-heuristic algorithms, analyzing the results in a more comprehensive manner.
- This study demonstrated the effectiveness of meta-heuristic HPO techniques in BC prediction on the WDBC benchmark dataset; the methodological framework may in principle be applicable to other structured, tabular disease-prediction problems, although such generalization would require independent empirical validation on the corresponding datasets and is not tested in this work.

The remainder of the paper is structured as follows: Section 2 presents the materials and methods, Section 3 provides an analysis of the experimental results, and the final section discusses the findings and implications of the study.

## 2 Materials and methods

In this study, the performance of various machine learning algorithms for BC prediction has been evaluated. The performance of popular machine learning algorithms such as XGBoost, AdaBoost, GB, DT, and KNN
was compared. The experiments were conducted on the Breast Cancer Wisconsin (Diagnostic) dataset (WDBC), a publicly available benchmark dataset derived from real patient samples, obtained via the Kaggle platform [12]. Metaheuristic HPO techniques were applied to improve the performance of the algorithms. For the comparative analysis, the selected metaheuristic optimization techniques were
those recommended in the past five years, published in Q1 journals, having high citation counts, and involving code sharing. These algorithms include Starfish Optimization Algorithm (SFOA), Puma Optimizer (PO), Fire Hawk Optimizer (FHO), Snake Optimizer (SO), Aquila Optimizer (AO), and Equilibrium Optimizer (EO). Furthermore, performance comparisons were also made with commonly used foundational algorithms such as Genetic Algorithm (GA), Differential Evolution (DE), Artificial Bee Colony Optimization (ABC), and Particle Swarm Optimization (PSO). The performance of the algorithms was comprehensively analyzed using various evaluation metrics including accuracy, precision, recall, F1 score, confusion matrix, Cohen's Kappa, and log loss. The flowchart of the study is presented in Figure 1.

In the methodology illustrated in Figure 1, the initial phase involves preprocessing the dataset. This stage encompasses the identification of missing values, and, if present, their imputation using standard techniques, as well as the transformation of categorical variables into numerical representations. After splitting the dataset into 80% training and 20% testing subsets, 5-fold cross-validation was applied only on the training data during the HPO process. Following the determination of the optimal hyperparameters, the final model was retrained using the entire training set and evaluated on the independent 20% test set. The following step involves the optimization of hyperparameters using ten distinct metaheuristic optimization algorithms. Finally, the performance of each model is evaluated using an array of metrics, including accuracy, precision, and recall, which are employed to provide a comprehensive analysis of the models' efficacy

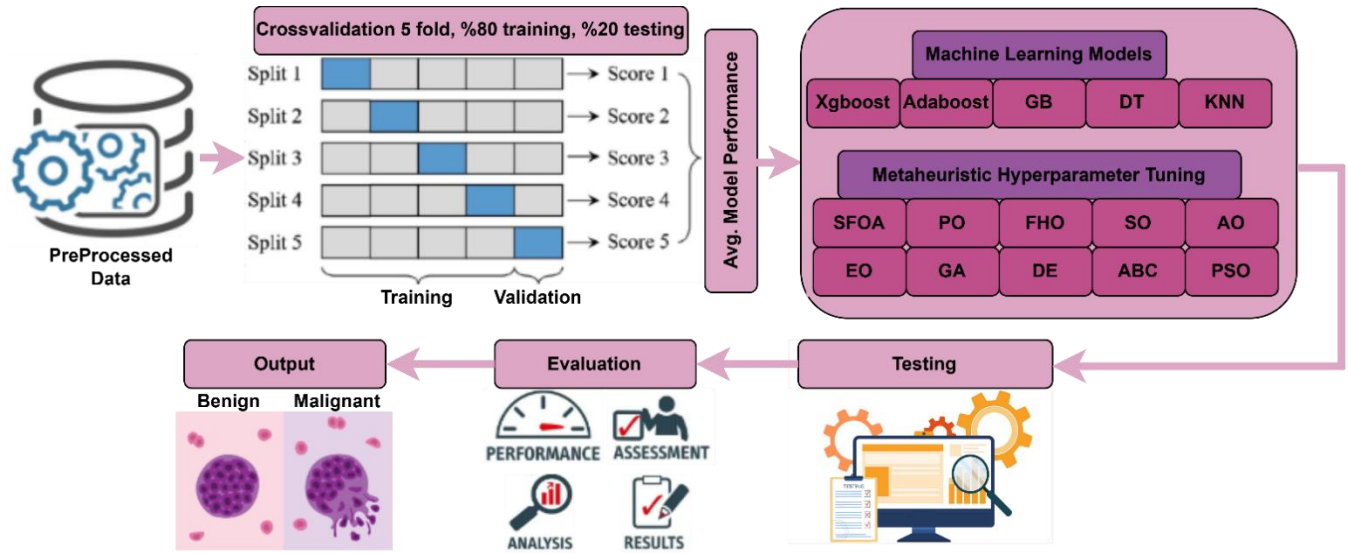


**Figure 1.** A methodological flowchart of the meta-heuristic hyperparameter tuning process for enhancing BC prediction models

### 2.1 Dataset description

The Breast Cancer Wisconsin (Diagnostic) [12] dataset is an important resource used in BC diagnosis, consisting of 569 real patient samples. Each sample represents a tumor and contains 30 numerical features, along with a categorical target variable, 'diagnosis', indicating whether the tumor is benign or malignant. These features are based on geometric and texture measurements of the tumor, such as mean radius (radius_mean), mean texture (texture_mean), perimeter (perimeter_mean), area (area_mean), smoothness (smoothness_mean), compactness (compactness_mean), concavity (concavity_mean), symmetry (symmetry_mean), and fractal dimension (fractal_dimension_mean). Additionally, for each feature, columns representing the standard error (se) and the worst-case scenario (worst) are also included. The dataset contains no missing values. Benign tumors account for 357 samples, while malignant tumors account for 212 samples. This structure plays a crucial role in data analysis and model development. The dataset is a valuable resource with high potential for early detection and classification of BC.

### 2.2 GB model

GB is an advanced machine learning strategy that iteratively learns by utilizing the errors of the previous decision tree structures. The algorithm consists of four main components: base learner, loss function, boosting, and learning rate. In the algorithm, a new shallow tree is added to the tree ensemble in each iteration, with each new tree trained using the error information from its predecessors. The overall prediction of the algorithm is calculated as the weighted average of all the trees [13, 14].

### 2.3 XGBoost model

The XGBoost is a widely utilized and effective machine learning strategy in classification problems. The algorithm constructs an ensemble-based model by combining multiple decision trees in a sequential manner. The residuals predicted by each tree are passed on to the subsequent tree. Each new tree is constructed with the objective of minimizing the residuals of the previous tree. The algorithm consists of a loss function and a regularization process. The former is employed to calculate the difference between the predicted and actual values. The latter serves to prevent overfitting, thus enhancing the model's generalization capabilities [15].

### 2.4 Adaptive boosting (AdaBoost) model

AdaBoost is a machine learning strategy that produces a high-performing tree model from multiple low-performing tree models. In the algorithm, each tree is trained using the error of the previous tree. Thus, in each iteration, weak trees gradually become stronger trees. In the end, the algorithm's prediction performance is determined as the weighted average of the trees [16].

### 2.5 DT model

Decision tree is a machine learning strategy known for its low parameter requirements, operating on labeled data. The algorithm is fundamentally composed of nodes and branches. Nodes vary depending on the data structure, and processing each node results in the outcomes, referred to as leaves. Decision trees are commonly used for classification and regression tasks. In these problems, it is crucial to carefully adjust certain parameters for the algorithm to perform accurately. These parameters play a significant role in error calculations and determining the most appropriate split points in the dataset. Additionally, the depth of the decision tree has a significant impact on the model's performance; while increasing the depth can improve predictions up to a certain point, excessive depth may lead to overfitting the model [17].

### 2.6 KNN model

The KNN algorithm is an instance-based, non-parametric machine learning method commonly used for classification and regression tasks involving labeled data. The algorithm predicts the label of a new data point based on the majority vote or average of its 'k' nearest neighbors, with proximity measured using distance metrics such as Euclidean, Manhattan, or Minkowski. Unlike model-based algorithms, KNN does not require a training phase, which makes it both easy to implement and flexible. In classification problems, KNN assigns the class label of the majority of the nearest neighbors to new observations. In the regression problems, it estimates a continuous output using the average or weighted average of the 'k' nearest neighbors' values. The choice of 'k' and the distance metric play a crucial role in determining the algorithm's accuracy and generalizability. Smaller values of 'k' tend to make the model more sensitive to noise in the data, while larger values of 'k' typically result in smoother decision boundaries and enhance the model's robustness [18].

### 2.7 Performance metrics

Relying on a single performance metric for model evaluation can be misleading. Therefore, assessing the model's performance using multiple metrics provides a more realistic approach. In this study, metrics such as accuracy, F1-score, recall, and precision have been utilized [19].

Accuracy: This metric reflects the overall correctness of the model by showing the fraction of accurate predictions relative to the total number of predictions made.

Precision: This metric evaluates how many of the instances identified as "positive" by the model are genuinely positive, focusing on the accuracy of positive predictions.

Recall: This metric measures how well the model identifies all actual positive instances, indicating the model's ability to correctly detect positives.

F1-Score: This metric combines both precision and recall, providing a harmonic mean that balances the two, especially useful when there is an uneven class distribution.

### 2.8 Confusion matrix

The confusion matrix is a crucial tool for evaluating a model's performance by visually presenting the relationship between the actual classes and the predicted ones. This matrix assesses the model's performance using four key components: TP (True Positive), FP (False Positive), TN

(True Negative), and FN (False Negative). These components indicate in which classes the model made correct predictions and where errors occurred [20].

### 2.9 Correlation matrix

The correlation matrix is a metric that quantifies the relationship between features. Each cell in the matrix is represented by a value within the range of [-1, 1]. A value of -1 indicates a perfect negative relationship, 0 signifies no relationship, and 1 represents a perfect positive relationship between the features. The correlation matrix, particularly in multivariate analysis, facilitates the quick visualization of relationships between variables, thereby aiding in the understanding of the relationships within the dataset [21].

### 2.10 Cohen's Kappa

Cohen's Kappa measures the agreement between two models' classifications while eliminating the influence of random chance. The Kappa value ranges from -1 to 1; 1 indicates perfect agreement, 0 represents random agreement, and negative values indicate disagreement in the opposite direction. Cohen's Kappa provides a more reliable assessment, particularly in situations where class imbalance exists and accuracy might be misleading [22].

### 2.11 Log loss

This metric evaluates the accuracy of the probabilities predicted by the model and calculates the difference between the actual labels and the predicted probabilities. Low log loss values indicate that the model's predictions are accurate, whereas higher values signal incorrect predictions. Log loss provides a more detailed assessment, especially in multi-class classification problems, by taking into account the probabilities for each class. To improve the model's performance, efforts are made to minimize the log loss value [23].

### 2.12 Meta-Heuristic algorithms

Metaheuristic optimization algorithms are widely used and effective tools for solving complex and nonlinear problems. These algorithms go beyond traditional deterministic methods by employing heuristic and nature-inspired approaches. Among the most commonly used metaheuristic algorithms are GA, DE, PSO, and ABC algorithms. These algorithms typically explore a given solution space either randomly or through a specific structure, followed by evaluating the quality of the solution using a fitness function. The primary advantage of metaheuristic algorithms is their ability to avoid local optima and find solutions that are close to the global optimum. However, these methods generally do not provide guarantees regarding the precision of the solution. The success of these algorithms is often contingent on the nature of the solution space, the tuning of the algorithm's parameters, and the strategies employed. Consequently, metaheuristic optimization algorithms are effectively applied to a wide range of complex problems [24]. The algorithms selected in this study have been published in the past five years and have garnered more citations than the algorithms proposed within the same year. Additionally, foundational and widely-utilized algorithms, such as GA, DE, ABC, and PSO, have also been incorporated alongside these algorithms. In this study, the metaheuristic algorithms employed are presented in Table 1.

**Table 1.** Meta-heuristic optimization algorithms used for hyperparameter tuning

| Algorithm | References |
|---|---|
| SFOA | [24] |
| PO | [25] |
| FHO | [26] |
| SO | [27] |
| AO | [28] |
| EO | [29] |
| ABC | [30] |
| GA | [31] |
| DE | [32] |
| PSO | [33] |

## 3 Experimental results

In this study, the performance of various machine learning algorithms for BC prediction and the HPO performance of metaheuristic optimization algorithms are evaluated. For this purpose, the dataset available on the Kaggle platform was employed. The experimental studies were conducted on a computer equipped with a 12th Gen Intel® Core™ i5-12400 2.50 GHz processor, 16 GB RAM, and Intel® UHD Graphics 730, running on a Windows operating system. All modeling, data preprocessing, HPO, and performance evaluation processes were implemented using the Python programming language. Necessary Python libraries, including NumPy, Pandas, Scikit-learn, XGBoost, and related optimization libraries, were utilized throughout the study.

Preprocessing steps applied to the dataset included transforming categorical variables into numerical values, scaling the data, and handling missing values. The dataset was partitioned into 80% training and 20% testing subsets. After splitting the dataset into 80% training and 20% testing subsets, 5-fold cross-validation was applied only on the training data during the HPO process. Following the determination of the optimal hyperparameters, the final model was retrained using the entire training set and evaluated on the independent 20% test set.

Metaheuristic methods were utilized for the HPO of the algorithms. The algorithms were selected in two main groups. For the first group, selection was based on the criteria of being published in the last five years, receiving the highest citation count in their respective publication year, and the availability of open-source code. The second group consisted of fundamental and widely-known optimization algorithms, such as GA, DE, ABC, and PSO. Thus, a comparative performance analysis of both newly proposed and classical optimization algorithms was conducted. Each algorithm was executed independently 30 times, and the average and standard deviation values were computed. The recommended parameter values for each algorithm were selected based on the existing literature. The parameter settings of the metaheuristic optimization algorithms used in

the experiments are presented in detail in Table 2. This table includes the control parameters employed during the execution of each algorithm as well as the standard values recommended in the related literature. In this way, consistency and reproducibility of the comparative analyses are ensured. Preliminary experiments determined the population size to be 20 and the maximum number of iterations to be 100. No early stopping mechanism was employed throughout the experimental process; all model training runs were terminated after a fixed 100 iterations. The optimized hyperparameters for each machine learning algorithm, along with their lower and upper bounds, are presented in Table 3. The metaheuristic HPO employed the accuracy value of the models as the objective function to maximize. Consequently, the performance of the models was further enhanced.

**Table 2.** Algorithm-specific parameter settings of metaheuristic optimization methods

| Algorithm | Parameters and Values |
|---|---|
| SFOA | Phase transition parameter (Gp) = 0.5 |
| PO | PI mechanism switching condition (Nc): adaptive/internal; Random factor (Rand): internally generated stochastic control variable |
| FHO | Number of fire hawks (NFH) ≈ adaptive (problem-dependent, internally determined); Number of prey (NP) ≈ adaptive (NFH complementary structure) |
| SO | Exploration coefficient (c1) = 0.5; Position update coefficient (c2) = 0.05; Exploitation coefficient (c3) = 2; Food/temperature thresholds (Q1, Q2) = 0.25 and 0.6 |
| AO | Fine-tuning constants (α, δ) = 0.1; Spiral control constants (U, r) = 0.00565 and 10 |
| EO | Equilibrium pool size (C_eq) = 4; Generation probability (GP) = 0.5; Control constants (a1, a2) = 2 and 1 |
| ABC | Limit parameter = SN × D (standard definition from Karaboga's formulation) |
| GA | Crossover probability (Pc) = 0.6–0.9; Mutation probability (Pm) = 0.001–0.05; Selection strategy = tournament/roulette wheel; Elitism rate = 0.05–0.10 |
| DE | Mutation factor (F) = 0.5–0.9; Crossover rate (CR) = 0.1 or 0.9; Strategy = DE/rand/1/bin |
| PSO | Constriction factor χ = 0.729; Cognitive coefficient c1 = 1.49445; Social coefficient c2 = 1.49445 |

**Table 3.** Optimized hyperparameters of machine learning algorithms and their corresponding lower and upper bounds

| Algorithms | Optimized Hyperparameters | Lower bound | Upper bound |
|---|---|---|---|
| XgBoost | learning_rate | 0.01 | 0.3 |
| | n_estimators | 50 | 500 |
| | max_depth | 3 | 15 |
| | min_child_weight | 1 | 10 |
| | subsample | 0.5 | 1 |
| | colsample_bytree | 0.3 | 1 |
| AdaBoost | n_estimators | 10 | 200 |
| | learning_rate | 0.01 | 1 |
| | max_depth | 1 | 20 |
| | min_samples_split | 2 | 10 |
| | min_samples_leaf | 1 | 10 |
| GB | learning_rate | 0.01 | 0.3 |
| | n_estimators | 50 | 500 |
| | max_depth | 3 | 15 |
| | min_samples_split | 2 | 10 |
| | min_samples_leaf | 1 | 10 |
| | subsample | 0.5 | 1 |
| DT | max_depth | 1 | 20 |
| | min_samples_split | 2 | 10 |
| | min_samples_leaf | 1 | 10 |
| KNN | n_neighbors | 1 | 50 |

Table 4 presents the best numerical results obtained from the machine learning algorithms before hyperparameter optimization and after hyperparameter optimization performed using SFOA. The first column of the table lists the machine learning models, while the first row displays the evaluation metrics, with the numerical results obtained from the conducted experiments. The reported results were obtained from the independent test set, while 5-fold cross-validation was employed during the hyperparameter optimization and model selection stages, with the performance of each model assessed both prior to (N) and following (HPO) hyperparameter optimization. For the sake of reproducibility and fair comparison among the machine learning models, the train-test split and all machine learning procedures were performed using a fixed random seed of 42.

**Table 4.** Numerical results obtained for BC diagnosis prediction using machine learning algorithms and SFOA-based metaheuristic HPO

| Algorithm | Method | Accuracy | Precision | Recall | F1 Score | Cohen's Kappa | Log Loss |
|---|---|---|---|---|---|---|---|
| XGBoost | N | 0.9561 | 0.9524 | 0.9302 | 0.9412 | 0.9063 | 0.1024 |
| | HPO | 0.9912 | 1.0000 | 0.9767 | 0.9882 | 0.9812 | 0.0505 |
| AdaBoost | N | 0.9474 | 0.9302 | 0.9302 | 0.9302 | 0.8880 | 0.4007 |
| | HPO | 0.9737 | 0.9762 | 0.9535 | 0.9647 | 0.9437 | 0.2395 |
| GB | N | 0.9474 | 0.9302 | 0.9302 | 0.9302 | 0.8880 | 0.2100 |
| | HPO | 0.9649 | 0.9756 | 0.9302 | 0.9524 | 0.9245 | 0.3425 |
| DT | N | 0.9386 | 0.9091 | 0.9302 | 0.9195 | 0.8700 | 0.3000 |
| | HPO | 0.9649 | 1.0000 | 0.9070 | 0.9512 | 0.9239 | 0.8974 |
| KNN | N | 0.7544 | 0.7419 | 0.5349 | 0.6216 | 0.4466 | 0.3213 |
| | HPO | 0.7807 | 0.7250 | 0.6744 | 0.6988 | 0.5267 | 0.5744 |

N: Normal, HPO: Hyperparameter optimization

However, due to the stochastic nature of the metaheuristic optimization algorithms, each optimization algorithm was executed independently for 30 runs with different random initializations, and the reported mean and standard deviation values were calculated from these independent runs. To avoid potential data leakage, the dataset was initially partitioned into training (80%) and test (20%) subsets. Within the standardization process, the StandardScaler parameters were learned only from the training data and then applied to the test set using the same transformation statistics. In addition, all cross-validation procedures were performed exclusively on the training subset. In the context of quality metrics, a smaller Log Loss value indicates superior performance, whereas higher values in all other metrics are indicative of better model performance and thus more favorable evaluation outcomes.

Table 4 summarizes the numerical performance results obtained in the BC diagnosis prediction study based on machine learning algorithms and metaheuristic HPO methods. The results obtained using the default parameter settings of the models (N: Normal) are compared with the best results achieved after applying SFOA-based HPO. Upon scrutinizing Table 4, it is apparent that HPO has further refined the performance of the XGBoost model. The accuracy has surged to 99.12%, with both precision and recall approaching near-perfect levels of 100%. The F1 score has ascended to 98.82%, and Cohen's Kappa has elevated to 98.12%, thereby augmenting the model's overall reliability. Notably, the GB model's log loss increased from 0.2100 to 0.3425 after HPO—a 1.63-fold deterioration—suggesting that while accuracy improved marginally (0.9474 → 0.9649), the model's confidence in its probabilistic outputs substantially degraded. This pattern is consistent with overfitting to the training fold accuracy metric, underscoring the need for calibration-aware optimization strategies.

With HPO, the AdaBoost algorithm has achieved an accuracy of 97.37%. Both precision (97.62%) and recall (95.35%) have improved, while the F1 score has risen to 96.47%. Cohen's Kappa has advanced to 94.37%. Notably, the log loss has diminished to 0.2395, suggesting that the model now makes more precise predictions with reduced uncertainty.

Following optimization, the DT algorithm has achieved an accuracy of 96.49%, with precision increasing from 90.91% to 100%; recall, however, decreased slightly from 93.02% to 90.70%, indicating that the model became more conservative in its positive predictions while correctly identifying a marginally smaller proportion of malignant cases. The F1 score has increased to 95.12%, and Cohen's Kappa has escalated to 92.39%. However, the log loss value has surged to 0.8974, indicating that post-optimization, the model experiences greater uncertainty in its predictions.

For the KNN algorithm, HPO has led to an accuracy of 78.07%. Recall improved considerably, from 53.49% to 67.44%, whereas precision decreased slightly, from 74.19% to 72.50%; the F1 score nonetheless rose to 69.88%, indicating that the net gain in classification performance was driven primarily by the improvement in recall rather than precision. Cohen's Kappa has increased to 52.67%, reflecting improved performance post-optimization. However, the log loss value increased from 0.3213 to 0.5744 following HPO—a 1.79-fold deterioration—indicating that while discriminative performance improved, the model's probability outputs became less well-calibrated. This finding suggests that optimizing solely for accuracy can compromise the probabilistic integrity of the model to a measurable degree. In clinical decision support contexts, even moderate miscalibration is noteworthy, as risk stratification systems rely on well-calibrated probability estimates rather than binary predictions alone. These results reinforce the importance of incorporating calibration-aware metrics such as log loss or Brier score as co-objectives in future HPO frameworks.

HPO generally improved discriminative accuracy across most models; however, a notable finding is the deterioration of log loss values in three of five models following HPO. This pattern suggests that optimizing exclusively for accuracy as the fitness function can induce probability miscalibration: the models assign less well-calibrated confidence scores even when discrete classification improves. The most pronounced calibration deterioration was observed in the DT model (log loss: 0.3000 → 0.8974), followed by GB (0.2100 → 0.3425) and KNN (0.3213 → 0.5744). Both XGBoost and AdaBoost showed improvement in both accuracy and log loss simultaneously. These findings caution against interpreting accuracy gains as holistic performance improvements. In clinical decision support applications, probability calibration is as critical as classification accuracy, since clinicians rely on risk scores—not merely binary predictions—to guide treatment decisions. Future work should consider incorporating log loss or Brier score directly into the fitness function, or applying post-hoc calibration methods (e.g., Platt scaling, isotonic regression) to correct for this miscalibration.

In Figure 2, the confusion matrices of the XGBoost algorithm are compared before and after HPO Figure 2a illustrates the confusion matrix of the XGBoost model without HPO. Here, the correctly classified positive (malignant) samples are 40, and the correctly classified negative (benign) samples are 69. The false positive and false negative values are observed to be 2 and 3, respectively. This result indicates that the model performs well in general, yet some misclassifications persist. In contrast, Figure 2b presents the confusion matrix of the XGBoost model after the application of HPO. In this case, the correctly classified positive (malignant) and negative (benign) samples are 42 and 71, respectively, while false positives are 0 and false negatives are reduced to only 1. These results suggest that the improvements brought about by HPO have enhanced the model's accuracy. The HPO notably reduced the misclassification rates and augmented the overall classification performance. In summary, the comparison in Figure 2 reveals that HPO has led to a substantial enhancement in the classification performance of the XGBoost algorithm. Post-HPO, the model's accuracy

increased, while the misclassification rate significantly decreased.

Figure 3 compares the confusion matrices of the Adaboost algorithm before and after HPO. Figure 3a presents the confusion matrix of the Adaboost model without HPO, where the correctly classified positive (malignant) samples are 40, and the correctly classified negative (benign) samples are 68, with false positives and false negatives observed as 3 and 3, respectively. These results indicate that while the model exhibits a generally competent performance, certain misclassifications persist, suggesting the necessity for further refinement. Conversely, Figure 3b presents the confusion matrix of the AdaBoost model after HPO. The model correctly classified 41 positive (malignant) and 70 negative (benign) samples, with only 1 false positive and 2 false negatives, demonstrating improved classification performance. This marked reduction in misclassification rates and the increase in correct classifications underscores the substantial improvement in model accuracy attributable to HPO. In summary, the comparison in Figure 3 demonstrates the positive impact of HPO on the classification performance of the Adaboost algorithm.

Figure 4 juxtaposes the confusion matrices of the GB algorithm before and after HPO. Figure 4a depicts the confusion matrix for the GB model without HPO, where 40 positive (malignant) and 68 negative (benign) samples are correctly classified, with false positives and false negatives recorded as 3 and 3, respectively. These results suggest that while the model exhibits an overall satisfactory performance, it still commits certain misclassifications. In contrast, Figure 4b presents the confusion matrix for the GB model post-HPO, showing 40 positive (malignant) and 70 negative (benign) samples correctly classified, with false positives reduced to 1 and false negatives remaining at 3. Following HPO, the model demonstrates an increase in correct classifications and a slight reduction in misclassifications, particularly highlighted by the decrease in false positives from 3 to 1, indicating a marked improvement in the model's predictive accuracy.

Figure 5 presents a comparative analysis of the confusion matrices for the DT algorithm, both with and without the application of HPO. In panel a, the DT model without HPO demonstrates a correct classification of 40 positive (malignant) and 67 negative (benign) instances, accompanied by 4 false positives and 3 false negatives. This suggests a relatively competent performance, albeit with an inclination towards false positives, reflecting a potential area of improvement in the model's precision. In contrast, panel b illustrates the results of the DT model after HPO. Notably, the correct classification of negative (benign) instances has increased to 71, while positive (malignant) instances correctly classified decreased slightly to 39, and the number of false positives has been reduced to zero, which signifies a substantial enhancement in the model's specificity. However, the false negatives increased from 3 to 4, indicating that, although the model became more precise in labeling benign cases, it now misses a marginally larger number of true malignant instances. Overall, the results underscore the efficacy of HPO in optimizing the performance of the DT model, particularly in terms of reducing false positives, though some limitations in minimizing false negatives persist.

Figure 6 presents a comparative analysis of the confusion matrices for the KNN algorithm, both with and without the application of HPO. Figure 6a depicts the confusion matrix for the KNN model without HPO, wherein 23 positive (malignant) and 63 negative (benign) instances are correctly classified, with 8 false positives and 20 false negatives. This indicates a significant propensity for misclassification, particularly among positive instances, revealing the model's susceptibility to false negatives. In contrast, Figure 6b illustrates the confusion matrix for the KNN model after HPO, where the number of correctly classified negative (benign) instances decreases to 60, while the correct classification of positive (malignant) instances increases to 29. Nevertheless, false positives rise to 11, while false negatives decrease from 20 to 14. These findings suggest

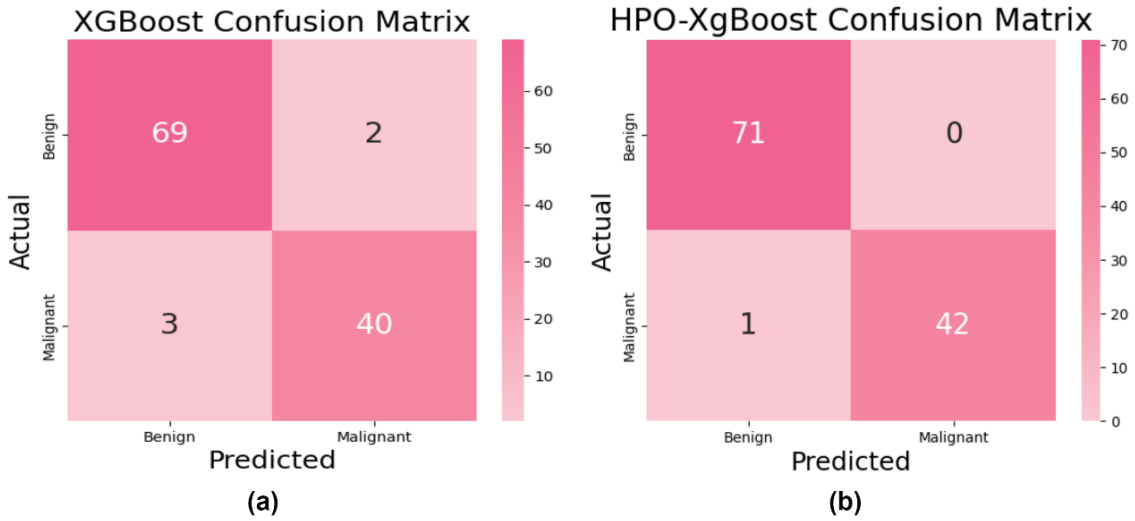


**Figure 2.** Confusion matrix comparison, a) XGBoost confusion matrix, b) HPO-XGBoost confusion matrix

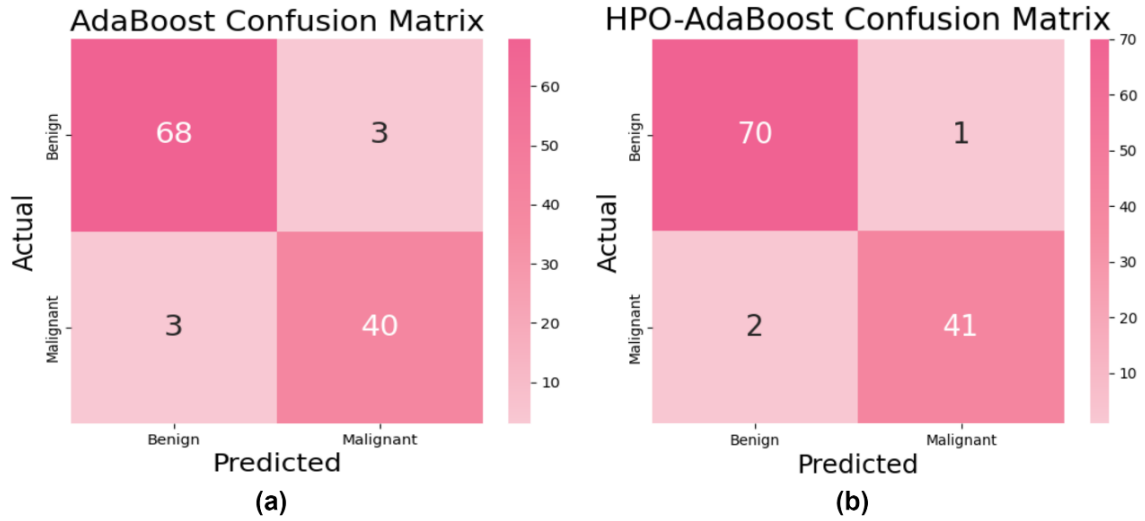


**Figure 3.** Confusion matrix comparison, a) Adaboost confusion matrix, b) HPO-AdaBoost confusion matrix

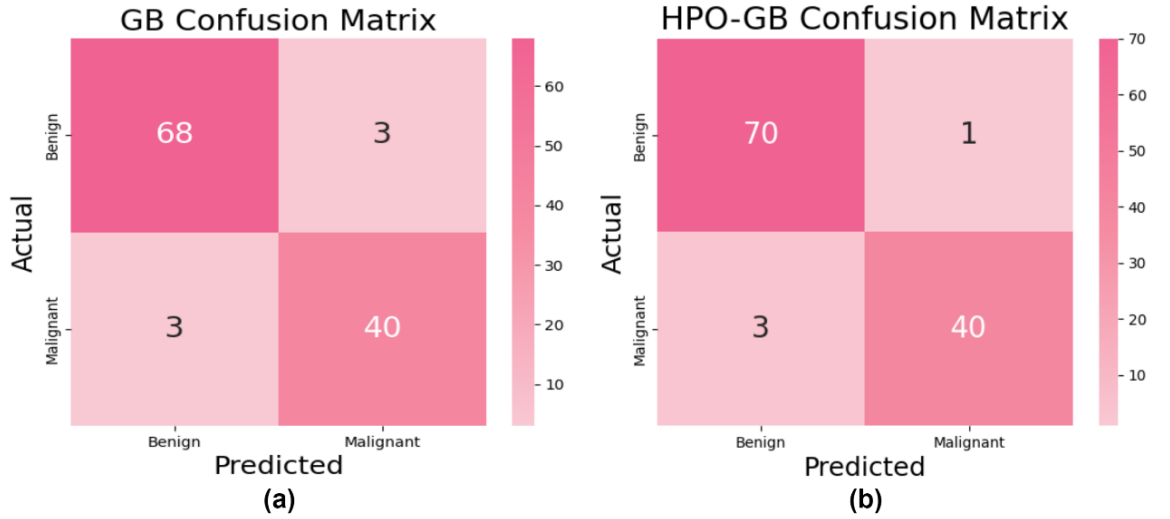


**Figure 4.** Confusion matrix comparison, a) GB confusion matrix, b) HPO-GB confusion matrix

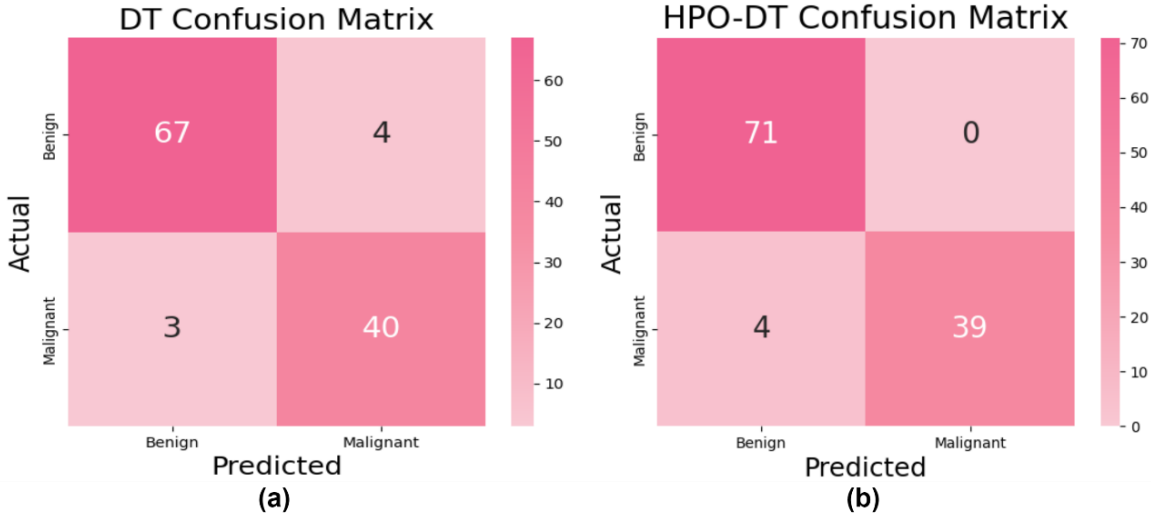


**Figure 5.** Confusion matrix comparison, a) DT confusion matrix, b) HPO-DT confusion matrix

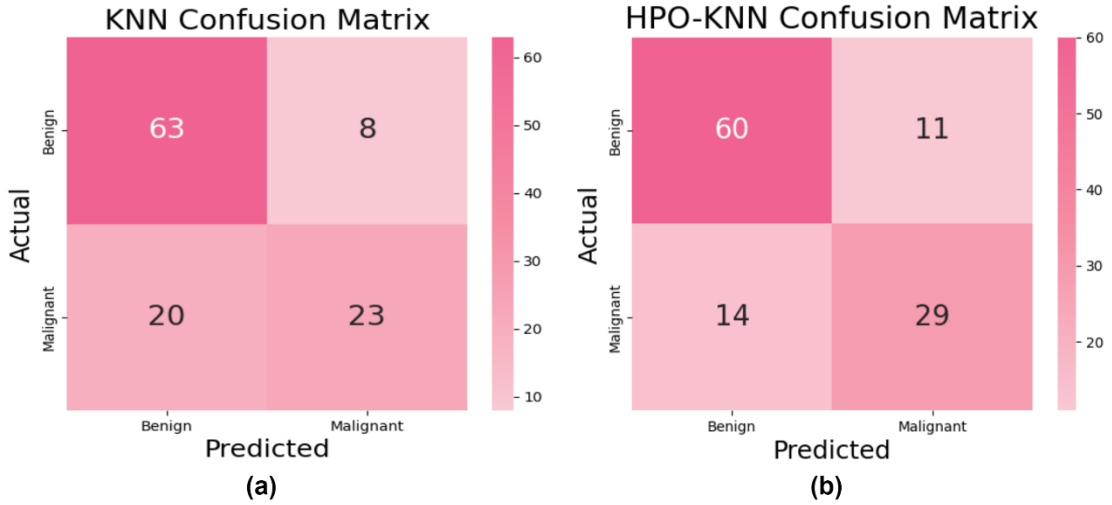


**Figure 6.** Confusion matrix comparison, a) KNN confusion matrix, b) HPO-KNN confusion matrix

that, despite an improvement in the classification of negative instances, HPO leads to an increase in false positives and the persistence of misclassified positive instances. Overall, while HPO has enhanced certain facets of the KNN model's performance, it underscores the need for a more refined balance between precision and recall to optimize classification efficacy.

Figure 7 illustrates the correlation matrix obtained for BC prediction using the HPO-XGBoost methodology. Upon examining Figure 7, a notably high positive correlation between "radius_mean" and "perimeter_mean" (0.9979) and between "radius_mean" and "area_mean" (0.9874) is particularly striking. This suggests a robust interrelationship among these three features, likely indicative of factors that influence each other. Similarly, the substantial correlation between "perimeter_mean" and "area_mean" (0.9865) reveals a closely synchronized variation between these two parameters, which may reflect analogous biological or structural attributes. Additionally, correlations between "compactness_mean" and "concavity_mean" (0.8831), as well as between "compactness_mean" and "concave points_mean" (0.8311), are also notably high, underscoring that these features offer critical insights into the geometric architecture of the cells. Another salient observation from the correlation matrix is the strikingly high correlation (0.9651) between "radius_worst" and "perimeter_worst," suggesting that these parameters may play a pivotal role in BC detection. Similarly, the elevated correlation (0.9592) between "area_worst" and "perimeter_worst" signifies that these parameters likely become more pronounced in advanced stages of cancer. On the other hand, features with relatively lower correlations, such as "fractal_dimension_mean," though less conspicuous, may still contribute meaningfully to the early detection of cancer. These parameters, despite exhibiting lower correlation, may provide subtle yet valuable clues for the diagnosis and staging of BC, serving as potential harbingers of underlying malignancy not immediately apparent through more prominent correlations.

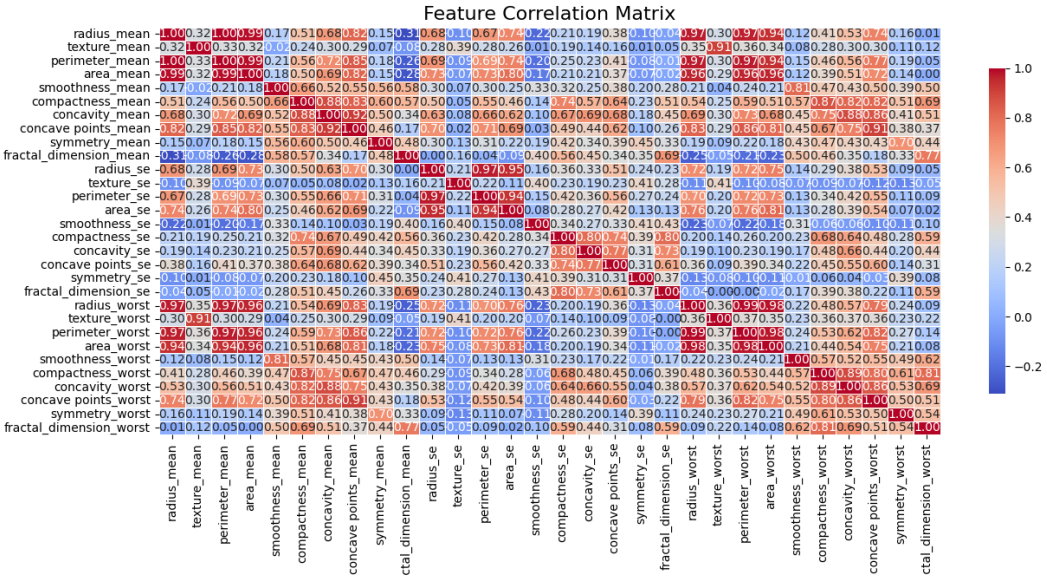


**Figure 7.** HPO-XGBoost correlation matrix

Figure 8 presents a multivariate analysis that explores the interrelationships among specific features used in BC diagnosis. The pairplot visualization encompasses scatter plots and bivariate relationships, distinguishing between Benign (blue) and Malignant (orange) tumors. Notably, a strong linear correlation has been identified between the variables Radius Mean, Perimeter Mean, and Area Mean. These variables reflect the geometric dimensions of the tumors, and their high correlation raises concerns of multicollinearity, potentially impairing model performance. It has been observed that Malignant tumors typically exhibit larger radii, perimeters, and areas, underscoring their critical role in classification tasks. In contrast, the weak correlation between Smoothness Mean and Compactness Mean suggests that these variables may be independent and should be analyzed separately. In machine learning modeling, the elimination or combination of highly correlated variables is recommended, while features like Compactness and Smoothness, despite their independent characteristics, could also prove influential in enhancing classification accuracy.

Table 5 presents the descriptive statistics of the accuracy results obtained from ten different metaheuristic algorithms used for HPO of the XGBoost model, along with the non-parametric pairwise comparisons conducted using SFOA as the reference method. For each algorithm, the mean ± standard deviation (Mean ± SD) values obtained from 30 independent runs are reported. The differences between SFOA and the other nine algorithms were evaluated using the Mann–Whitney U test, which is appropriate for independent samples. The test statistic (U), raw p-value, and statistical significance status after Bonferroni correction ($\alpha = 0.05/9 = 0.0056$) are presented in separate columns.

As shown in Table 5, the SFOA algorithm achieved the highest mean accuracy value among the evaluated methods ($0.9888 \pm 0.0105$). The PO algorithm exhibited a very similar mean accuracy ($0.9886 \pm 0.0108$), and the difference between PO and SFOA was not statistically significant ($U = 473.0$, $p = 0.7389$). Similarly, the FHO ($0.9834 \pm 0.0050$, $p = 0.0079$), SO ($0.9855 \pm 0.0092$, $p = 0.1449$), AO ($0.9863 \pm 0.0097$, $p = 0.2768$), and ABC ($0.9861 \pm 0.0012$, $p = 0.0376$) algorithms did not exhibit statistically significant differences from SFOA after Bonferroni correction, indicating that these five algorithms achieved comparable performance levels to SFOA.

**Table 5.** Descriptive statistics and non-parametric comparisons of XGBoost accuracy results.

| Algorithm | Mean ± SD | U (vs. SFOA) | p-value | Significant after Bonferroni |
|---|---|---|---|---|
| SFOA | 0.9888 ± 0.0105 | – | – | – |
| PO | 0.9886 ± 0.0108 | 473.0 | 0.7389 | No |
| FHO | 0.9834 ± 0.0050 | 630.0 | 0.0079 | No |
| SO | 0.9855 ± 0.0092 | 549.0 | 0.1449 | No |
| AO | 0.9863 ± 0.0097 | 524.0 | 0.2768 | No |
| EO | 0.9831 ± 0.0050 | 645.0 | 0.0040 | **Yes** |
| ABC | 0.9861 ± 0.0012 | 591.0 | 0.0376 | No |
| GA | 0.9826 ± 0.0023 | 681.0 | 0.00065 | **Yes** |
| DE | 0.9789 ± 0.0087 | 687.0 | 0.00047 | **Yes** |
| PSO | 0.9827 ± 0.0014 | 686.0 | 0.00049 | **Yes** |

Kruskal-Wallis omnibus test (10 groups, n = 30 each): $H = 48.0071$, $p = 2.546 \times 10^{-7}$. Pairwise comparisons were performed using the Mann-Whitney U test (independent samples, non-paired). Bonferroni correction was applied for 9 pairwise comparisons ($\alpha = 0.05/9 = 0.0056$)

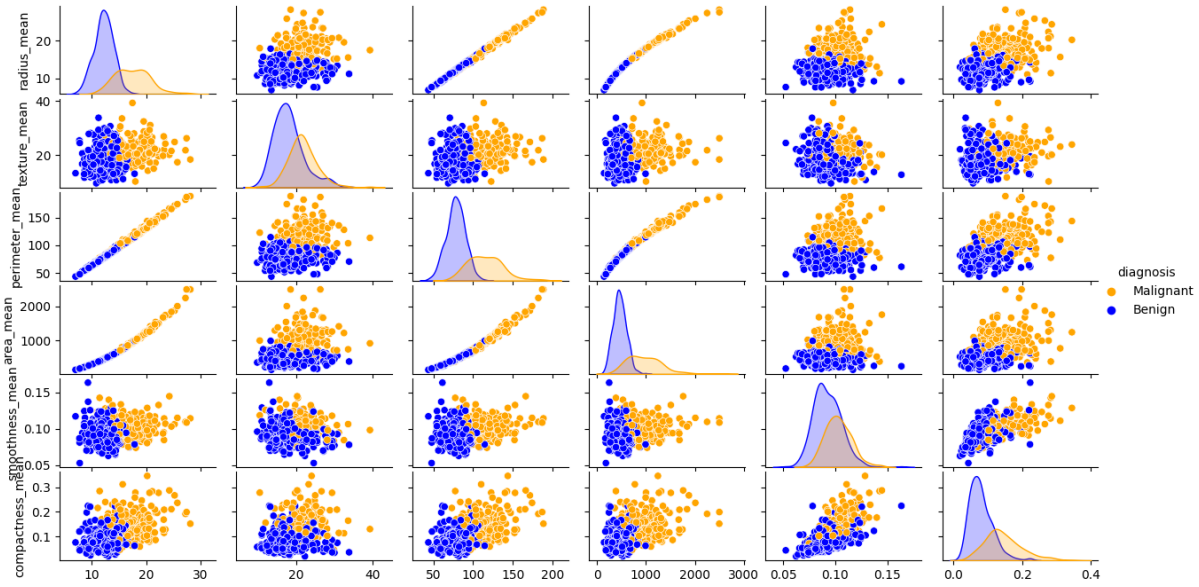


**Figure 8.** Examination of multivariate relationships for BC diagnosis

In contrast, the EO (0.9831 ± 0.0050, p = 0.0040), GA (0.9826 ± 0.0023, p = 0.00065), DE (0.9789 ± 0.0087, p = 0.00047), and PSO (0.9827 ± 0.0014, p = 0.00049) algorithms demonstrated significantly lower accuracy performance than SFOA, with p-values remaining statistically significant even after Bonferroni correction. Regarding variability, ABC exhibited the lowest standard deviation (SD = 0.0012), indicating the most stable performance among the evaluated algorithms, whereas DE (SD = 0.0087) and SFOA (SD = 0.0105) showed relatively higher variability across independent runs.

Overall, these findings demonstrate that SFOA achieved statistically significant superiority over four of the evaluated algorithms (EO, GA, DE, and PSO). However, this superiority could not be statistically confirmed against five algorithms (PO, FHO, SO, AO, and ABC). Therefore, SFOA can be considered a competitive optimization approach that provides strong performance but does not exhibit absolute dominance over all competing methods.

### *3.1 Calibration trade-off between accuracy and log loss*

A central finding of this study is that HPO driven solely by an accuracy-maximizing fitness function does not guarantee improved probabilistic calibration. As summarized in Table 4, log loss deteriorated in three of the five models following HPO: DT (0.3000 → 0.8974, a 2.99-fold increase), KNN (0.3213 → 0.5744, 1.79-fold) and GB (0.2100 → 0.3425, 1.63-fold), while XGBoost and AdaBoost were the only two models that improved simultaneously in both accuracy and log loss. This indicates that several models became more confident without becoming proportionally more correct in their predicted probabilities — a hallmark of miscalibration. In clinical decision-support contexts this distinction is material: a clinician relying on a predicted malignancy probability, rather than the binary label alone, would receive a less trustworthy risk estimate from the HPO-tuned DT or KNN models despite their higher point-accuracy. We therefore treat this trade-off, rather than the raw accuracy gains, as one of the most clinically relevant results of this study.

## 4 Conclusion

This study investigates the efficacy of meta-heuristic HPO techniques in enhancing the performance of machine learning models employed for BC prediction. The performance of algorithms, including XGBoost, AdaBoost, GB, DT, and KNN, was evaluated through a comparative analysis before and after the HPO process. The experiments were conducted on the dataset, with the initial accuracy rates of the algorithms recorded as 95.61%, 94.74%, 94.74%, 93.86%, and 75.44%, respectively. Following HPO using meta-heuristic algorithms, significant improvements in accuracy were observed, with the accuracy rates of XGBoost, AdaBoost, GB, and DT reaching 99.12%, 97.37%, 96.49%, and 96.49%, respectively. Conversely, the improvement in the KNN algorithm remained modest, with its accuracy increasing to 78.07%.

The results demonstrate that meta-heuristic HPO leads to substantial enhancements, particularly in more complex machine learning models such as XGBoost and AdaBoost, whereas the performance improvements in simpler models like KNN are more limited. Furthermore, it was found that optimization led to an increase in log loss values for some algorithms, which may be indicative of overfitting or the models becoming more complex. These findings suggest that while HPO is an effective method for boosting model performance, it requires careful management to avoid detrimental effects.

A key innovation of this study lies in its inclusion of a comparative analysis between widely used meta-heuristic algorithms, which have been proposed in the past five years and garnered significant citations, and traditional optimization techniques such as ABC and GA. This comprehensive comparison offers insights into the performance of both modern and conventional algorithms, thereby providing a nuanced understanding of their respective advantages. The success of HPO is contingent on the specific meta-heuristic algorithm employed and the management of the optimization process. In this study, the SFOA algorithm delivered the most competitive results in terms of accuracy. The DE algorithm exhibited the lowest average accuracy. Therefore, it can be concluded that, in this context, the newer algorithms did not uniformly outperform the traditional techniques in terms of accuracy; ABC, a foundational algorithm, achieved a higher mean accuracy (0.9861) than three of the six recently proposed algorithms (SO, FHO, and EO), indicating that algorithm age alone does not determine performance.

Several limitations of this study should be acknowledged. First, all experiments were conducted on a single dataset (WDBC, n = 569); the observed performance trends may not generalize to other BC datasets or to clinical data collected under different acquisition protocols. Second, the relatively small sample size limits the statistical power of between-algorithm comparisons and may contribute to the overfitting signals observed in the log loss results. Third, external validation on an independent cohort was not performed, which precludes claims regarding clinical deployability. Fourth, the fitness function used during HPO was restricted to classification accuracy; optimizing for calibration-aware metrics (e.g., Brier score, log loss) may yield better-calibrated probability outputs. Fifth, a fixed random seed (42) was employed in the machine learning models to ensure data partitioning consistency and experimental reproducibility. In contrast, considering the stochastic nature of metaheuristic optimization algorithms, each algorithm was evaluated through 30 independent runs. The variability arising from different initialization conditions was quantified by reporting the mean and standard deviation values obtained across these independent runs. These limitations define clear directions for future research, including multi-dataset validation, calibration-aware optimization, and prospective clinical evaluation.

In conclusion, this study demonstrates that metaheuristic HPO can yield meaningful improvements in classification accuracy for BC prediction models trained on the WDBC dataset. The SFOA algorithm provided the most favorable

trade-off between accuracy and computational efficiency among the ten algorithms tested. However, the concurrent deterioration of log loss values in three of the five models following HPO underscores that accuracy-only optimization is insufficient; probability calibration must be considered alongside discriminative performance. These findings are applicable to structured tabular datasets in analogous disease prediction tasks; however, generalization beyond the WDBC dataset requires further investigation.

## Conflict of interest

The author declares that there is no conflict of interest associated with this work.

## CRediT authorship contribution statement

**Harun Akbulut:** Conceptualization, Methodology, Investigation, Data curation, Formal analysis, Software, Resources, Project administration, Validation, Visualization, Writing – original draft, Writing – review & editing.

## Declaration of generative artificial intelligence (AI) and AI-assisted technologies in the writing process.

During the preparation of this work, the author used GPT-5.6 Luna (OpenAI) for language editing and translation purposes. Following the use of this tool, the author reviewed and edited the content as necessary and takes full responsibility for the content of the publication.

**Similarity rate (iThenticate):** %20

## References

[1] M. Arnold et al., "Current and future burden of breast cancer: Global statistics for 2020 and 2040," *The Breast*, 66, 15–23, 2022. https://doi.org/10.1016/j.breast.2022.08.010

[2] J. Wu and C. Hicks, "Breast cancer type classification using machine learning," *Journal of Personalized Medicine*, 11(2), 61, 2021. https://doi.org/10.3390/jpm11020061

[3] Z. Sabir, A. Nazzal, A. Srour, H. Koyunbakan, and S. Bhatti, "A machine learning radial basis deep neural network procedure for the solutions of the nonlinear breast cancer model," *Advances in Complex Systems*, 16(2), 2025. https://doi/10.1142/S1793962325500357

[4] Y. Kumar, A. Koul, R. Singla, and M. F. Ijaz, "RETRACTED ARTICLE: Artificial intelligence in disease diagnosis: a systematic literature review, synthesizing framework and future research agenda," *Journal of Ambient Intelligence and Humanized Computing*, 14(7), 8459–8486, 2023. https://doi.org/10.1007/s12652-021-03612-z

[5] Y. K. Dwivedi et al., "Artificial Intelligence (AI): Multidisciplinary perspectives on emerging challenges, opportunities, and agenda for research, practice and policy," *International Journal of Information Management*, 57, 101994, 2021. https://doi.org/10.1016/j.ijinfomgt.2019.08.002

[6] N. M. U. Din, R. A. Dar, M. Rasool, and A. Assad, "Breast cancer detection using deep learning: Datasets, methods, and challenges ahead," *Computers in Biology and Medicine*, 149, 106073, 2022. https://doi.org/10.1016/j.compbiomed.2022.106073

[7] S. A. A. Edsa, K. Sunat, and H. Guo, "Cost-sensitive neural network: A grey wolf optimizer-based approach for breast cancer prediction," *Expert Systems With Applications*, 266, 126176, 2025. https://doi.org/10.1016/j.eswa.2024.126176

[8] A. K. Das et al., "Machine Learning based Intelligent System for Breast Cancer Prediction (MLISBCP)," *Expert Systems With Applications*, 242, 122673, 2024. https://doi.org/10.1016/j.eswa.2023.122673

[9] Y. W. Chang et al., "Artificial intelligence for breast cancer screening in mammography (AI-STREAM): preliminary analysis of a prospective multicenter cohort study," *Nature Communications*, 16(1), 2248, 2025. https://doi.org/10.1038/s41467-025-57469-3

[10] Z. Zhang and Z. Li, "Evaluation methods for breast cancer prediction in machine learning field," *SHS Web of Conferences*, 144, pp. 03010, Dali, China, May 13-15, 2022. https://doi.org/10.1051/shsconf/202214403010

[11] B. Senthilkumar et al., "Ensemble modelling for early breast cancer prediction from diet and lifestyle," *IFAC-PapersOnLine*, 55(1), pp. 429–435, 2022. https://doi.org/10.1016/j.ifacol.2022.04.071

[12] W. Wolberg, O. Mangasarian, N. Street, and W. Street, "Breast cancer Wisconsin (Diagnostic) data set," *Kaggle*, [Online]. Available: https://www.kaggle.com/datasets/uciml/breast-cancer-wisconsin-data. [Accessed: Sep. 14, 2026]

[13] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," *The Annals of Statistics*, 29(5), 1189–1232, 2001. https://doi.org/10.1214/aos/1013203451

[14] J. H. Friedman, "Stochastic gradient boosting," *Computational Statistics & Data Analysis*, 38(4), 367–378, 2002. https://doi.org/10.1016/S0167-9473(01)00065-2

[15] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining (KDD '16)*, San Francisco, CA, USA, 2016, 785–794. https://doi.org/10.1145/2939672.2939785

[16] Y. Freund and R. E. Schapire, "A Decision-Theoretic generalization of On-Line learning and an application to boosting," *Journal of Computer and System Sciences*, 55(1), 119–139, 1997. https://doi.org/10.1006/jcss.1997.1504

[17] A. D. Gordon, L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, "Classification and regression trees," *Biometrics*, 40(3), 874, 1984. https://doi.org/10.2307/2530946

[18] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Transactions on Information Theory*, 13(1), 21–27, 1967. https://doi.org/10.1109/TIT.1967.1053964

[19] H. Akbulut, S. Atasever, and E. Sıramkaya, "Machine learning-based orange quality classification: A Hyperparameter optimization approach through Puma

optimizer," *2025 7th International Congress on Human-Computer Interaction, Optimization and Robotic Applications (ICHORA)*, pp. 1–5, Ankara, Türkiye, 2025. https://doi.org/10.1109/ICHORA65333.2025.11017028

[20] M. Sokolova and G. Lapalme, "A systematic analysis of performance measures for classification tasks," *Information Processing & Management*, 45(4), 427–437, 2009. https://doi.org/10.1016/j.ipm.2009.03.002

[21] P. Schober, C. Boer, and L. A. Schwarte, "Correlation Coefficients: appropriate use and interpretation," *Anesthesia & Analgesia*, 126(5), 1763–1768, 2018. https://doi.org/10.1213/ANE.0000000000002864

[22] J. Cohen, "A coefficient of agreement for nominal scales," *Educational and Psychological Measurement*, 20(1), 37–46, 1960. https://doi.org/10.1177/001316446002000104

[23] O. Rainio, J. Teuho, and R. Klén, "Evaluation metrics and statistical tests for machine learning," *Scientific Reports*, 14(1), 6086, 2024. https://doi.org/10.1038/s41598-024-56706-x

[24] C. Zhong, G. Li, Z. Meng, H. Li, A. R. Yildiz, and S. Mirjalili, "Starfish optimization algorithm (SFOA): a bio-inspired metaheuristic algorithm for global optimization compared with 100 optimizers," *Neural Computing and Applications*, 37(5), 3641–3683, 2025. https://doi.org/10.1007/s00521-024-10694-1

[25] B. Abdollahzadeh et al., "Puma optimizer (PO): a novel metaheuristic optimization algorithm and its application in machine learning," *Cluster Computing*, 27(4), 5235–5283, 2024. https://doi.org/10.1007/s10586-023-04221-5

[26] M. Azizi, S. Talatahari, and A. H. Gandomi, "Fire Hawk Optimizer: a novel metaheuristic algorithm," *Artificial Intelligence Review*, 56(1), 287–363, 2023. https://doi.org/10.1007/s10462-022-10173-w

[27] F. A. Hashim and A. G. Hussien, "Snake Optimizer: A novel meta-heuristic optimization algorithm," *Knowledge-Based Systems*, 242, 108320, 2022. https://doi.org/10.1016/j.knosys.2022.108320

[28] L. Abualigah, D. Yousri, M. A. Elaziz, A. A. Ewees, M. A. A. Al-Qaness, and A. H. Gandomi, "Aquila Optimizer: A novel meta-heuristic optimization algorithm," *Computers & Industrial Engineering*, 157, 107250, 2021. https://doi.org/10.1016/j.cie.2021.107250

[29] A. Faramarzi, M. Heidarinejad, B. Stephens, and S. Mirjalili, "Equilibrium optimizer: A novel optimization algorithm," *Knowledge-Based Systems*, 191, 105190, 2020. https://doi.org/10.1016/j.knosys.2019.105190

[30] D. Karaboga, B. Gorkemli, C. Ozturk, and N. Karaboga, "A comprehensive survey: artificial bee colony (ABC) algorithm and applications," *Artificial Intelligence Review*, 42(1), 21–57, 2014. https://doi.org/10.1007/s10462-012-9328-0

[31] J. R. Sampson, "Adaptation in natural and Artificial Systems (John H. Holland)," *SIAM Review*, 18(3), 529–530, 1976. https://doi.org/10.1137/1018105

[32] K. V. Price, R. M. Storn, and J. A. Lampinen, *Differential Evolution: A Practical Approach to Global Optimization*. Berlin, Germany: Springer, 2005. https://doi.org/10.1007/3-540-31306-0

[33] J. Kennedy and R. Eberhart, "Particle swarm optimization," *Proceedings of ICNN'95 - International Conference on Neural Networks*, pp. 1942–1948, Perth, WA, Australia, 1995. https://doi.org/10.1109/ICNN.1995.488968