Iraqi Journal for Computers and Informatics
Available online: https://ijci.uoitc.edu.iq/index.php/ijci
Volume 52, issue 1. 2026

**Print ISSN: 2313-190X, Online ISSN: 2520-4912**

# A Coherent LPB-Family Optimization Framework for Multilayer Perceptron Training in Heart Disease Prediction

Abbas M. Ahmed [a, b], Tarik A. Rashid [c]

[a] *Sulaimani Polytechnic University, Sulaimani, KR, Iraq.*
[b] *Informatics Institute for Postgraduate Studies, University of Information Technology and Communications (UoITC), Baghdad, Iraq.*
[c] *Computer Science and Engineering; AIIC, University of Kurdistan Hewler, Erbil, KR, Iraq.Corresponding.*

**ARTICLE INFO**

***Keywords**:*

*Machine learning, Multilayer Perceptron (MLP), heart disease prediction, neural network training, LPB-family optimization, mLPB-MLP.*

**ABSTRACT**

Cardiovascular diseases (CVDs) are the leading cause of mortality in the world. This paper constructs models of heart disease classification using a single-hidden-layer Multilayer Perceptron (1HL-MLP) that is trained using a bi-level model that utilizes Bayesian Hyperparameter Optimization (HPO) and three variations of evolutionary strategy-based Learner Performance-Based Behavior (LPB): LPB-MLP, aLPB-MLP and mLPB-MLP. Each of the four heart disease datasets was processed in a common pipeline based on schema alignment, median imputation, one-hot encoding, per-fold Z-score normalization, and Out-of-Fold (OOF) threshold tuning and their performance was checked by stratified K-fold and external testing on an independent dataset. Results have shown that the Modified LPB model (mLPB-MLP) performed better, and it has the highest discrimination and calibration (AUC = 0.9782, AUPRC = 0.9732) and the overall accuracy (93.66%), F1-score (93.53%), recall (94%), specificity (93%), and the lowest BCE loss (0.193). These findings indicate consistent optimization processes, reasonable probability tuning and sensitivity-specificity compromise. In general, the Bayesian HPO in combination with LPB-family evolutionary training using data will lead to a clinically robust, well-calibrated, and reproducible heart disease risk prediction model.

## 1. INTRODUCTION

The new technology has resulted in the fast evolution of the modern electronic and intelligent systems. The key role in this development was played by artificial intelligence (AI), which has revolutionized the most vital spheres of life, medicine, industry, academia, and national security, enabling data-driven decision-making in high-stakes environments [1]. The technologies based on AI can now be used to perform automated analyses, large-scale data processing, real-time monitoring, and statistical planning to enhance efficiency, minimize expenses, and increase the capabilities of machines in decision support. Within this context, machine learning (ML) and deep learning (DL) are central building blocks, as they offer the predictive and analytical capability to modern AI systems [2]. The recent development of machine learning (ML) is improving diagnostic modeling and clinical decision support. Cardiovascular diseases (CVDs) which is the major cause of death (more than 610,000 deaths every year in the U.S. alone) are not easily predicted since there are numerous risk factors such as hypertension and diabetes. Big-data analysis of biomedical data with the help of ML offers the possibility of early diagnoses and better risk stratification. [3] which are able to reduce the burden of disease.[4]. In the last 20 years, a myriad of metaheuristic algorithms has been developed, and each of them is defined by a unique design philosophy and benefit profile. This heterogeneity gives a full range of flexibility to build the classification models; however, at the same time, it makes it hard to decide which algorithm fits best with a particular task,

E-mail address:
*Email: abbas.ahmed@spu.edu.iq.* [a, b]
*Email: tarik.ahmed@ukh.edu.krd* [c]

Corresponding*: *Abbas M. Ahmed*

Received 21 December 2025, Accepted 10 March 2026

*DOI: 10.25195/ijci.v52i1.711*

considering the extent to which the evaluation criteria can differ [5]. Therefore, the combination of evolutionary and metaheuristic techniques in training neural networks is still a research topic that aims at enhancing the model accuracy, stability, and efficiency of convergence.

The key contributions of this paper are:

1. Improved hybrid models: Two better architectures, mLPB-MLP and aLPB-MLP, are designed through combining modified and adaptive LPB optimizers with an MLP with the introduction of adaptive mechanisms that increase convergence stability and exploration exploitation.
2. Bi-level optimization framework: It is proposed to have a unified scheme that consists of Bayesian hyperparameter optimization (outer level) and LPB-family weight optimization (inner level) to ensure more reliable and well-regularized learning.
3. Comparative performance verification: It has been extensively benchmarked with the standard LPB-MLP so that the proposed variants are more accurate, converge faster, and generalize better when it comes to predicting heart diseases.

The rest of this paper will be structured in the following way: Section 2 will review associated literature on the optimization of neural networks with bio-inspired metaheuristics. Section 3 describes the datasets and preprocessing processes. In section 4, the MLP architecture and the training-evaluation pipeline are described. Section 5 outlines the suggested mLPB and aLPB optimizers and the bi-level optimization model. Section 6 covers the research design, measurement criteria, and statistical comparison. Section 7 offers the external testing and credibility validation. Lastly, Section 8 summarizes the research and outlines the future research directions.

## 2. Literature Review

Metaheuristic optimization has been widely studied in the recent past to enhance the training of neural networks more than the traditional gradient-based training. Zhang et al. [6] The swift expansion of machine-learning-driven cardiovascular prediction systems, as well as the significance of novel optimization and hybrid modeling methods in enhancing the diagnostic performance, are also mentioned in recent extensive reviews. [7] proposed a PSO-BP hybrid, which combines the global search of PSO and local refinement of BP with adaptive parameters and a switching strategy; tested on parity, function approximation, and Iris problems, it is superior to single optimizers in terms of the final accuracy, convergence rate, and stability [8]. In the same manner, the training of an MLP has also used the Tree-Seed Algorithm (TSA), which has performed better than PSO and ABC on 18 datasets such as breast cancer and heart disease data and has shown 100 percent accuracy in certain cases. The performance of TSA shows that bio-inspired metaheuristics can be used to significantly increase the performance of classification, which justifies the applicability of this type of tool to the current project. The Butterfly Optimization Algorithm (BOA) has been used in study [9] to enhance the MLP training, and the study suggested a BOA-MLP hybrid that outperformed the MSE and classification accuracy in five benchmark datasets, which reflects the viability of bio-inspired optimization in neural networks. Similarly, [10] proposed RSMGWO, a superior Grey Wolf Optimizer that incorporates hierarchy-reinforcement, altered population dynamics, and random-opposition learning. RSMGWO has been shown to be very robust and easily scalable in biomedical contexts due to its performance surpassing standard GWO and other swarm algorithms over 19 benchmarks and real cancer-classification tasks. These developments are representative of the current trend to come up with better hybrid metaheuristics to solve MLP optimization. To put this development into context, Table 1 provides an overview of significant earlier research, showing the improvement of algorithms, datasets, and performance.

***Table 1.*** *Important Metaheuristic Algorithms that have been used in MLP Training.*

| Authors | Algorithm | Purpose | Domain / Dataset | Key Insights |
|---|---|---|---|---|
| **Wienholt** [11] | Evolution Strategies (ES) | Minimize MLP error | General | MLP error minimization is carried out with the help of ES. |
| **Seiffer** [12] | Genetic Algorithm (GA) | Avoid local minima | MLP training | GA helps escape local minima |
| **Abdurrakhman et al.** [13] | MLP + PSO | Optimize MLP for predicting biogas yield | Biogas dataset (pH, moisture, OLR, temperature) | $R^2$=0.90, minimum MSE, and maximized yield prediction achieved; PSO improves the performance of MLP. |
| **Mirjalili et al.** [14] | PSOGSA (PSO + GSA) | Train MLP | Classification tasks | Convergence and accuracy are better than PSO and GSA. |
| **Mirjalili et al.** [15] | Biogeography-Based Optimization (BBO) | Train MLP | 5 classification, 6 approximation datasets | BBO outperforms PSO, GA, ACO, ES, PBIL, BP, and ELM |

| | | | | |
|---|---|---|---|---|
| **Mirjalili** [16] | Grey Wolf Optimizer (GWO) | Train MLP | 5 classification, 3 approximation datasets | GWO outperforms PSO, GA, ACO, ES, PBIL |
| **Amirsadri et al.** [17] | BP + GWO (with Levy flight) | Train MLP | 12 classification/function datasets | BP enhances exploitation, and GWO enhances exploration. |
| **Xu et al.** [18] | ABC-ISB (Improved ABC) | Continuous optimization / MLP training | MLP training | ABC-ISB is superior to basic ABC. |
| **Heidari et al.** [19] | Ant Lion Optimizer (ALO) | Train MLP | Various datasets | ALO outperforms GA, PBIL, DE, PSO |
| **Dalwinder et al.** [20] | ALO + Feature weighting | Train MLP / increase classification rate | 3 breast cancer datasets | Feature weighting and ALO improve accuracy |
| **Faris et al.**[21] | Multi-Verse Optimizer (MVO) | trains MLP | 9 biomedical datasets (UCI) | MVO resembles GA, PSO, DE, Firefly and Cuckoo Search. |
| **Abbas et al.**[22] | FDO-MLP | Enhance training and convergence of MLP. | The data of student performance (287 x 21) | 0.97 accuracy; rapid convergence; good local-minima behavior; better than BP, GWO-MLP, and MGWO-MLP. |

On the whole, the literature demonstrates the fact metaheuristic techniques significantly enhance the training of MLP, and hybrid bio-inspired techniques are more accurate, convergent, and robust. The trends create a strong foundation towards the further development of neural-network optimization.

### 2.1 Research Gap

Current metaheuristic MLP literature normally assumes the use of one optimizer or set of operator parameters without the incorporation of evolutionary adaptive mechanisms or differentiation-evolution methods into a bi-level framework. In addition, the combination of Bayesian hyperparameter maximization and evolutionary weight training and validation of such models against heterogeneous heart-disease data via standard preprocessing and OOF-based thresholding has not been previously performed. Therefore, a methodological gap still exists in the development of a reproducible, well-calibrated LPB-based optimization model to augment discrimination, calibration, and convergence stability in clinical diagnostic prediction.

## 3. Dataset and Preprocessing

The following section describes the ML pipeline that is planned to be used in the LPB-MLP family with HPO, which is an overview of the process of data acquisition, preprocessing, training of the model, and evaluation. It starts with the overall description of data-loading protocol employed in all experiments, as shown in Fig. 1 [23], that includes the description of the workflow including data collection and preprocessing, training, evaluation and deployment.

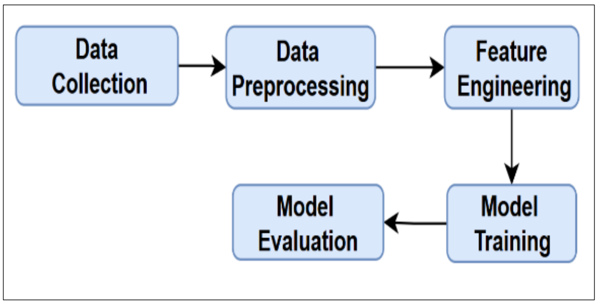



***Fig.1:*** *ML pipeline*

### 3.1. Heart Disease Datasets Description

To establish strong assessment, this research relies on numerous data on heart diseases of different clinical and open sources. Different in terms of demographics, clinical features, and size of features, all datasets have a binary diagnostic task (healthy vs. disease) in common. Their heterogeneity gives them a good base of testing the LPB-family models in a variety of real-life situations. Transparency and reproducibility are guaranteed by publicly available datasets, including the Kaggle source, which was utilized in this work (https://www.kaggle.com/datasets). Table 2 contains the most important characteristics of each dataset.

***Table 2:*** *A summary of the datasets used is presented.*

| Dataset | Samples | Features | Source | License | Target | Notes |
|---|---|---|---|---|---|---|
| **Dataset 1 – Cleveland Heart Disease (2024 update)** | 1025 | 14 | Cleveland Clinic (processed 2024) | Open-access (Updated release) | Binary (0/1) | Cleaned and standardized, the benchmark data set was the main reference. |
| **Dataset 2 – Clinical Heart Disease (Cleveland-derived)** | 304 | 10 | Subset of Cleveland | Public | Binary | Parsimonious and clinically interpretable; can be validated. |
| **Dataset 3 – Kaggle Heart Disease** | 918 | 11 | Kaggle (CC0) | CC0 | Binary | Both genders are feature-rich, and male-dominated (~79%). |
| **Dataset 4 – Multi-source (Statlog, Cleveland, Hungary)** | 1190 | 11 | Combined sources | Public (Compiled) | Binary | Severe imbalance; mild imbalance Fixed in preprocessing. |
| **External Dataset (Clinical Heart Disease, 2024)** | 1048 | 12 | Independent clinical dataset | Research-use (restricted access) | Binary (0 = healthy, 1 = diseased) | Clean, integer, and schema consistent. |

### 3.2. Data Collection and Preprocessing

The datasets were offered in both text and CSV formats and read with the help of the MATLAB functions (detectImportOptions and readtable) without changing the names of the original variables. The target column was automatically defined based on binary classification rules that were predefined. Separate features (X) and labels (Y), that is, target labels, were then separated, and then basic integrity checks, which included the dimensions and the distribution of the classes in the dataset, were performed. A uniform preprocessing pipeline had been used. The median imputation was used to deal with missing or infinite values [24].To avoid the leakage of data, continuous variables were normalized by per-fold Z-score normalization of training-fold statistics. In the case of datasets whose targets are continuous values, binary labels were created based on a median-split threshold [25]. One-hot encoding was used to transform categorical variables to numerical form [26][27]. The entire preprocessing has been carried out in the framework of the K-fold cross-validation to make the models evaluated unbiased.

### 3.4 Feature Engineering and Selection.

The feature engineering was used to have uniform and informative feature representation in all datasets. The initial clinical features were left, and the categorical variables were encodified with one-hot, which led to a moderate growth in the number of input features. This modification allowed the neural network to operate with categorical data without artificial ordinal correlations between category levels. None of the dimensionality reduction methods were utilized, and all of the processed features were used in the training and testing of the models. By preserving the interpretability of the original clinical variables, this methodology offers a stable feature space on which internal cross-validation as well as external testing can be performed, thus enabling the model to generalize reliably and allow reproducibility [28]. To offer quantitative transparency on the process of feature representation, Table 3 gives a summary of how many features the input used before and after preprocessing an individual dataset had. As anticipated, categorical clinical attributes when subjected to one-hot encoding led to a factor space dimensional increase. This is an expansion of categorical variables into a number of binary indicators without distorting the semantic meaning of the initial clinical attributes. The resulting space on features was systematically used in the internal cross-validation and external validation steps to guarantee consistency and reproducibility of methods used in the experimental results.

***Table 3.*** *Number of features before and after preprocessing*

| Dataset | Original Features | After One-Hot Encoding |
|---|---|---|
| Cleveland Heart Disease (2024 update) | 14 | 25 |
| Clinical Heart Disease (Cleveland-derived) | 10 | 20 |
| Kaggle Heart Disease | 11 | 22 |
| Multi-source heart disease (Statlog + Cleveland + Hungary) | 11 | 23 |
| External Clinical Dataset (2024) | 12 | 24 |

### 4. Design and Optimization Framework of the Model:

The LPB-MLP, aLPB-MLP, and mLPB-MLP classifiers have a single-neuron sigmoid output, and all of them are biased to binary biomedical classification. Binary cross-entropy is used to train (with optional MSE and L2 regularization), and we initialize weights with a quasi-Xaver scheme that has been adapted to the LPB dynamics of stable gradient flow. The differences between the models are mostly the evolutionary optimization: the LPB-MLP has a fixed LPB routine, the aLPB-MLP has adaptive crossover and mutation, and the mLPB-MLP has multi-stage adaptive control to achieve a better global-local search ratio and quicker convergence. Each of the variants allows you to tune hyperparameters flexibly, which allows you to strongly optimize hyperparameters in either a cross-validation or external evaluation.

#### 4.1 MLP Architecture and LPB-Based Optimization

The MLP used in this study consists of an input layer, a single hidden layer, and a sigmoid output neuron for binary classification [29]. Model parameters (weights and biases)[12], are optimized using the LPB-family evolutionary framework within the proposed bi-level optimization scheme.

#### 4.2 Modified Learner Performance-Based Behavior (mLPB) Algorithm

The mLPB algorithm is an expansion of the original LPB model, which incorporates the use of Differential Evolution (DE) operators to balance exploration and exploitation. Candidate solutions in LPB take the form of groupings (the learners) organized into performance-based categories—Perfect, Good, and Bad—interacting with one another by way of cooperative and competitive learning behaviors [30]. The mLPB variant further improves this structure to use DE-based mutation and crossover to the Combined Perfect Group (CPF) to allow more global search and preserve adaptive diversity among the population. In this hybrid mechanism the DE operators create new trial solutions that are competing with already existing individuals, and those that are fitter are maintained through the generations. By doing so, it can converge faster, stabilize learning dynamics, and optimize performance compared to the original LPB, especially in multimodal search space. Implementing the mLPB-MLP training cycle along with the scheme of the process, such as the system of initialization, fitness measurement, iterative refinements, and converging, is outlined in [31].

#### 4.3 Adaptive LPB (aLPB) Algorithm

The Adaptive Learner Performance-Based Behavior (aLPB) algorithm is an extension of the LPB model that does not rely on constant control parameters and makes use of dynamically varying crossover and mutation operators instead. All these adaptive mechanisms react to population fitness and allow the algorithm to transition between exploration and exploitation during optimization [32]. Parents with high fitness are provided with conservative crossover adjustments in order to narrow the search to optimal regions, and parents with low fitness are provided with larger-scale mixing. The strength of mutation is also controlled, becoming more pronounced when the state of improvement stagnates and reducing when the solution quality improvement takes place, which promotes convergence at a higher rate, keeping stability. Once offspring have been generated by means of adaptive crossover and mutation, truncation-based selection selects and maintains the most promising ones. This self-controlled mechanism discourages their premature convergence and promotes global search efficiency and more rapid and precise optimization than the original LPB and the DE-enhanced mLPB versions. Further details about the procedure and some mathematical formulas may be obtained in [32].

#### 4.4 Training MLP with the mLPB Optimization Algorithm (within the LPB-MLP family)

In this paper, the family of neural optimizers used is the LPB-MLP, LPB-MLP, mLPB-MLP, and aLPB-MLP, which will be used to train MLP. In order to describe the training pipeline in concrete terms, we use mLPB-MLP as an illustrative example, but it can be applied to LPB-MLP (fixed operators) and aLPB-MLP (fully adaptive operators) as well, with the difference that the operator definitions and policies of adaptation of the parameters change. The goal of training an MLP is the ideal weight and bias that leads to a minimization of the objective function of the network. Gradient-based techniques (e.g., back-propagation) can be very slow to converge or can be trapped in local minima when dealing with nonlinear high-dimensional space. An improvement of these limits has been achieved through the use of mLPB-MLP, where metaheuristic search is used to increase the ratio of exploration-exploitation by incorporating DE operators into the original structure of LPB, which increases the rate of convergence of the search, its stability, and overall optimization performance. Here, every population member is a coded representation of a complete candidate set of network weights and biases, and the dimensionality of a search space is as in the case of Eq. 1:

$$D = (n_{in} \times n_h) + (n_h \times n_{out}) + n_h + n_{out} \qquad (1)$$

In which $n_{in}$, $n_h$, and $n_{out}$ are the number of inputs, the number of hidden neurons, and the number of output neurons, respectively.

The mLPB-MLP optimization is a series of steps, consisting of three steps:

- Mutation: interrupts the choice people to experiment with new areas and sustain diversity of populations.
- Crossover: transfers good structures to propagate useful information between candidates.
- Selection: keeps the most successful solutions to the successive generation and gradually guides the population to high-quality weight-bias configurations.

This cycle has the effect of mLPB-MLP gradually improving model parameters, which converges more quickly and reliably than traditional gradient-based training, especially in high-dimensional complex landscapes. The general MLP training process using the LPB family is outlined in Fig.2, where the candidate solutions are judged by iteration, network weights/biases are revised, and the network performance is constantly enhanced.

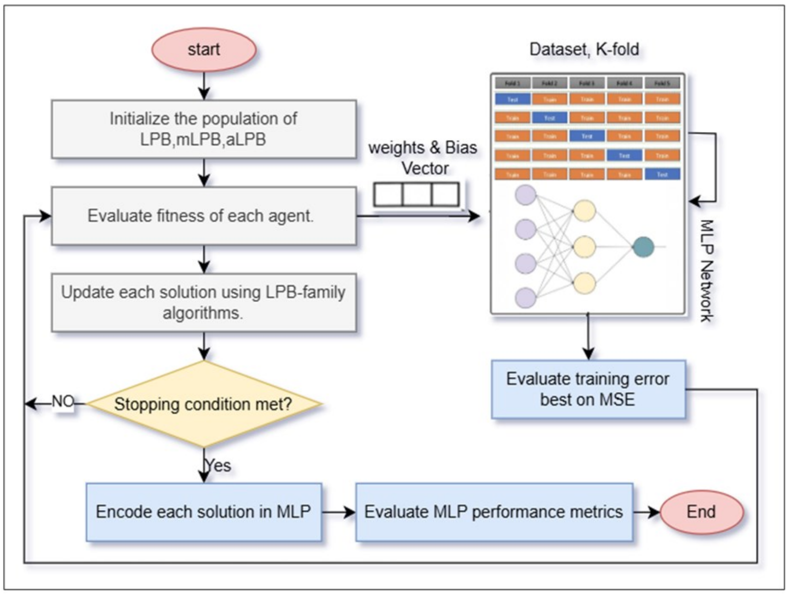



***Fig.2:*** *Overall Workflow of Training an MLP Using the LPB-Family Optimization Algorithms*

### 4.5 Unified Bi-Level Pipeline

The LPB-MLP, mLPB-MLP, and aLPB-MLP models are run in a single bi-level learning pipeline, which combines Bayesian hyperparameter optimization, MLPB-family metaheuristic training, and K-Fold Out-of-Fold validation to provide effective and reproducible performance. The architecture of all the models is the same: 1HL-MLP and preprocessing workflow. only the evolutionary strategy of updating weights and biases varies; fixed LPB exploration-exploitation tradeoffs (fixed and adaptive) are applied to update mLPB and aLPB. All these strategies contribute to the stability of convergence, classification accuracy, and the generalization on internal folds and external datasets. In this end-to-end pipeline, the external data are automatically aligned with the training schema using median imputation, one-hot encoding, and Z-score of per-fold standardization. Outputs in [ε,1−ε] are numerically stabilized by predicting them within [ε,1−ε], with $\varepsilon=10^{-7}$. On inference, all the models load the stored weights, biases, normalization parameters, and feature ordering and provide protection against polarity mismatches identified by ROC analysis. These constituents combine preprocessing and optimization of the LPB family into a seamless bi-level model that maintains consistent scaling, calibration, and implementation.

### 5. Bi-Level Optimization and Hyperparameter Optimization.

Training of the LPB-family models (LPB-MLP, mLPB-MLP, and aLPB-MLP) is based on a composite bi-level model together with the inner-loop weight optimization and outer-loop Bayesian Hyperparameter Optimization (HPO). This embedded network offers adaptive control of network parameters as well as algorithmic hyperparameters, enhancing stability of convergence and model generalization.

**5.1 Inner Objective: Learning of weights.**

The inner-level goal reduces loss of training, including MSE with L2 regularization on cross-validation folds, which are stratified. This model achieves a tradeoff between predictive accuracy and model complexity and provides consistency across folds [33]. MSE is the loss of choice since it has a smooth optimization space and a steady convergence point, whereas L2 is used to limit the magnitude of the weights to prevent overfitting, which is essential when dealing with noisy or correlated biomedical inputs. Collectively, they constitute a composite goal that enhances scale weights and enhanced generalization in folds that are varied. The stratified 5-fold cross-validation preserves class balance and eliminates bias. Each fold runs Z-score normalization parameters estimated over the training subset to avoid leakage, and each variant of the LPB family applies evolutionary search to optimize its weights. Combining fold-wise validation terms provides a strong approximation of a pre-external prediction of performance, which is reproducible and over-fitting-resistant learning with respect to changes in data partitions

**5.2 Outer Objective—Bayesian Hyperparameter Optimization.**

Bayesian optimization is used in the outer layer of the bi-level framework, searching the space of hyperparameters of the LPB-family models and selecting those that produce the smallest cross-validated validation loss. It is necessary to complement this process with the inner weight-learning loop and to provide adaptive and data-driven tuning of the most important parameters: hidden layer size, population size, limit on iteration, and regularization strength [34].

**5.2.2 Bayesian Optimization Mechanism**

Bayesian HPO is based on the validation loss of a Gaussian Process surrogate, and it picks new hyperparameter candidates with an Expected Improvement (EI) acquisition function. This is a balanced approach where the uncertain regions are explored and the promising areas are exploited so that they converge efficiently to the optimum configurations [35].

**5.2.3 Integration Between Inner and Outer Levels**

The bi-level architecture aligns the inner-loop optimization of weights and the outer-loop optimization of hyperparameters. Although both the LPB-family algorithms improve weights and biases during each fold, Bayesian HPO considers these results to steer the search to hyperparameter settings that offer strong cross-validated performance. This communication facilitates steady fold-specific learning coupled with guiding the global search to stable and high-performing solutions.

**5.3. Integrated Learning and Evaluation Process.**

The two levels are seen to work collectively to create a hierarchical optimization process: the inner loop is an exploitation process that is localized using the LPB-family solvers, and the outer loop is a global exploration process that is done in the hyperparameter space. This jointed process improves the efficiency of convergence, generalization between internal folds and external data, and stability of the calibration process, which are especially significant in biomedical applications, including prediction of heart diseases.

**6. Experimental Protocol, Validation, and Metrics**

It gives a description of the workflow of the experiment of training, validating, and testing the LPB-family of models, the cross-validation procedure, and the adaptive thresholding strategy, as well as the performance metrics family of measures to understand predictive quality and generalization.

**6.1 Cross-Validation Protocol**

The stratified 5-fold cross-validation process, was used to guarantee the unbiased learning and good generalization. The computation of Z-score parameters was carried out only using training subsets to eliminate leakage, and the various LPB-family models conducted evolutionary weight optimization under this scheme. The aggregation of the out-of-fold (OOF) predictions was then carried out to allow the use of an adaptive threshold to be selected and produce diagnostic plots like ROC and PR curves, confusion matrices, and variability boxplots. This protocol favors the same evaluation in the presence of imbalance in classes [36].

**6.2 Adaptive Threshold Optimization**

Because a fixed threshold (e.g., 0.5) may produce suboptimal trade-offs between sensitivity and specificity, especially when an unbalanced dataset is used, the LPB-family classifiers use a data-driven thresholding approach using OOF predictions. The best threshold to use is that which gives the highest F1-score, resulting in a more balanced separation of positive and negative cases and a higher external diagnostic accuracy in heart disease prediction [37].

### 6.3 Evaluation Metrics

A complete set of measures (discrimination, calibration, and imbalance robustness) was used to measure model performance. These are accuracy, precision, recall (sensitivity), specificity, F1-score, binary cross-entropy (BCE), AUC, AUPRC, MSE, MAE, Matthew correlation coefficient (MCC), and Cohen Kappa (k).

Accuracy gives a general correctness measure that is constrained in case of imbalance. True-negative detection is measured with precision and recall, respectively, and true-positive reliability and sensitivity are measured with specificity and recall, respectively. The F1 score is a combination of precision and recall into one harmonic indicator. BCE and MSE/MAE are calibration-based metrics, which penalize overconfident mispredictions and measure the degree to which predicted probabilities are close to true labels, respectively. AUC and AUPRC summarize ranking and positive-class quality at thresholds, the latter being especially useful with biomedical data that is skewed. MCC is an unbiased performance report with all the elements of the confusion matrix, and it is usually more accurate than accuracy or F1 in biased environments. The kappa of Cohen measures the agreement that is not due to chance, which is complementary to the other measures in the case of unequal distribution of the classes [38][39][40].

### 6.4 Comparative Results and Complexity of Computations.

The mLPB-MLP framework's computational cost is determined as a function of the MLP architecture, the dataset size, and the number of individuals in the evolutionary population, as well as the number of function evaluations needed for training. Based on Eq. 2 taken from [8], here is how we calculate the overall training complexity:

$$O_{MLP-mLPB} = O\left(MaxFE_s\,.\left(O(MLP) + O(mLPB)\right)\right) \qquad (2)$$

Where $MaxFE_s$ is the maximum number of function evaluations, The cost of a single MLP forward pass is:

$$O(MLP) = t\,.(h + o) \qquad (3)$$

Collectively, the two equations, 2 and 3, indicate that the size of the model and the parameters of the evolutionary search process determine the complexity of full training and provide a realistic approximation of high-dimensional or large-scale optimization workloads.

### 6.5 Tables of Comparative Model Results and Analysis.

There are two views involved in model evaluation: **a)** Average ± Std metrics of generalization between folds. and **b)** FINAL MODEL (FULL): Final model that is trained on the entire data to be deployed.

***Table 4:*** *Cross-validation performance metrics mean ± Standard Deviation Across multiple datasets for LPB-Family Models.*

| Model / Metrics | Dataset 1 | | | Dataset 2 | | | Dataset 3 | | | Dataset 4 | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **LPB-MLP** | **aLPB-MLP** | **mLPB-MLP** | **LPB-MLP** | **aLPB-MLP** | **mLPB-MLP** | **LPB-MLP** | **aLPB-MLP** | **mLPB-MLP** | **LPB-MLP** | **aLPB-MLP** | **mLPB-MLP** |
| **Accuracy** | 88.49 ±2.92 | 90.73 ±2.78 | 91.80 ±1.52 | 75.91 ±3.96 | 82.19 ±3.07 | 82.50 ±1.56 | 84.86 ±2.33 | 86.93±1.97 | 85.73 ±1.55 | 83.36 ±2.09 | 83.45 ±2.72 | 85.21 ±1.09 |
| **Precision** | 89.31 ±3.67 | 90.82 ±3.64 | 93.39 ±2.74 | 77.16 ±5.72 | 82.15 ±3.13 | 83.05 ±4.08 | 86.17 ±2.23 | 87.61 ±1.45 | 86.55 ±2.91 | 84.23 ±2.88 | 83.35 ±2.53 | 85.66 ±2.32 |
| **Recall (Sensitivity)** | 86.77 ±2.65 | 90.18 ±3.48 | 89.58 ±1.70 | 80.00 ±2.71 | 86.06 ±4.60 | 86.06 ±8.99 | 86.63 ±4.91 | 88.97 ±3.01 | 87.99 ±2.01 | 84.42 ±1.83 | 85.85 ±3.43 | 86.65 ±3.61 |
| **Specificity** | 90.12 ±3.44 | 91.25 ± .73 | 93.91±2.75 | 71.06 ±9.72 | 77.59 ±4.36 | 78.36 ±8.53 | 82.68 ±3.70 | 84.39 ±2.00 | 82.93 ±4.31 | 82.17 ±4.06 | 80.74 ±3.22 | 83.60 ±3.59 |
| **F1-Score** | 88.01 ±3.03 | 90.46 ±2.88 | 91.42±1.51 | 78.41 ±2.69 | 84.00 ±3.02 | 84.15 ±2.30 | 86.32 ±2.37 | 88.26±1.86 | 87.23 ±1.26 | 84.30 ±1.76 | 84.56 ±2.61 | 86.08 ±1.20 |

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **BCE Loss** | 0.302 ±0.08 | 0.258 ±0.02 | 0.223 ±0.03 | 0.542 ±0.10 | 0.399 ±0.08 | 0.391±0.06 | 0.380 ±0.07 | 0.340 ±0.03 | 0.352 ±0.03 | 0.381 ±0.02 | 0.372 ±0.01 | 0.363 ±0.02 |
| **AUC (ROC)** | 0.944 ±0.02 | 0.959 ±0.06 | 0.967±0.09 | 0.852 ±0.04 | 0.902 ±0.08 | 0.909 ±0.05 | 0.910 ±0.04 | 0.926 ±0.07 | 0.924 ±0.03 | 0.911 ±0.09 | 0.914 ±0.009 | 0.919 ±0.014 |
| **AUPRC** | 0.939 ±0.05 | 0.954 ±0.06 | 0.962 ±0.01 | 0.852 ±0.05 | 0.894 ±0.03 | 0.898±0.01 | 0.907 ±0.02 | 0.928 ±0.06 | 0.924 ±0.00 | 0.905 ±0.04 | 0.913 ±0.021 | 0.918 ±0.016 |
| **MSE** | 0.083 ±0.0149 | 0.074 ±0.0913 | 0.063 ±0.0138 | 0.164 ±0.0262 | 0.126 ±0.0267 | 0.120 ±0.0170 | 0.114 ±0.0211 | 0.102 ±0.0126 | 0.107 ±0.0096 | 0.118 ±0.0083 | 0.117 ±0.0083 | 0.113 ±0.0090 |
| **MAE** | 0.157 ±0.0272 | 0.160 ±0.0153 | 0.139 ±0.0303 | 0.271 ±0.0289 | 0.251 ±0.0245 | 0.229 ±0.0237 | 0.216 ±0.0294 | 0.208 ±0.0059 | 0.208 ±0.0056 | 0.218 ±0.0114 | 0.229 ±0.0102 | 0.225±0.0092 |
| **MCC** | 0.769 ±0.05 | 0.815 ±0.05 | 0.836±0.03 | 0.514 ±0.08 | 0.641 ±0.06 | 0.656 ±0.03 | 0.695 ±0.04 | 0.735±0.03 | 0.711 ±0.03 | 0.666 ±0.04 | 0.668 ±0.05 | 0.704±0.02 |
| **Kappa** | 0.769 ±0.05 | 0.814 ±0.05 | 0.835 ±0.03 | 0.511 ±0.08 | 0.639 ±0.06 | 0.646 ±0.02 | 0.693 ±0.04 | 0.735 ±0.03 | 0.710 ±0.03 | 0.666 ±0.04 | 0.667 ±0.05 | 0.703 ±0.02 |

In Table 4, across datasets, mLPB-MLP has better generalization, with more accurate results (e.g., 91.80% ±1.52), BCE loss (0.223 ±0.03), and calibration results (AUC = 0.967 ± 0.009). The lower MSE/MAE values support that it shows better control over errors compared to LPB-MLP and aLPB-MLP. In Table 5, mLPB-MLP shows the best performance (single-fold performance) throughout all datasets. One example: Dataset 1—F1= 93.88, Precision = 95.83, AUC = 0.981. This shows that it can obtain optimal behavior when exposed to good training conditions.

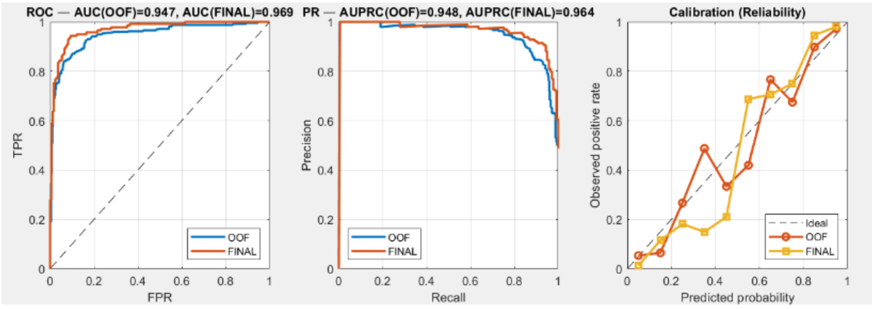
***Table 5:*** *Comparison of Performance of Fully Trained Models (FINAL MODEL FULL) (Represents deployment-ready configuration).*

| Model / Metrics | Dataset 1 | | | Dataset 2 | | | Dataset 3 | | | Dataset 4 | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **LPB-MLP** | **aLPB-MLP** | **mLPB-MLP** | **LPB-MLP** | **aLPB-MLP** | **mLPB-MLP** | **LPB-MLP** | **aLPB-MLP** | **mLPB-MLP** | **LPB-MLP** | **aLPB-MLP** | **mLPB-MLP** |
| **Accuracy** | 89.85 | 92.29 | 94.63 | 84.82 | 88.78 | 87.79 | 87.15 | 86.71 | 86.06 | 82.18 | 85.21 | 85.13 |
| **Precision** | 85.71 | 91.02 | 94.05 | 80.51 | 86.59 | 84.41 | 84.70 | 85.35 | 84.17 | 77.84 | 82.59 | 85.31 |
| **Recall** | 94.99 | 93.39 | 94.99 | 95.15 | 93.94 | 95.15 | 93.70 | 91.73 | 92.13 | 92.69 | 91.26 | 86.80 |
| **Specificity** | 84.98 | 91.25 | 94.30 | 72.64 | 82.61 | 78.99 | 79.02 | 80.49 | 78.54 | 70.41 | 78.43 | 83.24 |
| **F1-Score** | 90.11 | 92.19 | 94.52 | 87.22 | 90.12 | 89.46 | 88.97 | 88.43 | 87.97 | 84.62 | 86.71 | 86.05 |
| **BCE Loss** | 0.230 | 0.234 | 0.194 | 0.323 | 0.298 | 0.280 | 0.334 | 0.327 | 0.329 | 0.356 | 0.340 | 0.351 |
| **AUC (ROC)** | 0.968 | 0.967 | 0.979 | 0.936 | 0.948 | 0.953 | 0.927 | 0.932 | 0.931 | 0.922 | 0.928 | 0.923 |
| **AUPRC** | 0.964 | 0.962 | 0.974 | 0.937 | 0.944 | 0.952 | 0.921 | 0.938 | 0.936 | 0.929 | 0.931 | 0.925 |
| **MSE** | 0.067 | 0.066 | 0.051 | 0.098 | 0.086 | 0.083 | 0.098 | 0.098 | 0.100 | 0.109 | 0.106 | 0.109 |
| **MAE** | 0.152 | 0.152 | 0.130 | 0.203 | 0.203 | 0.189 | 0.192 | 0.204 | 0.196 | 0.209 | 0.222 | 0.224 |
| **MCC** | 0.801 | 0.846 | 0.892 | 0.703 | 0.775 | 0.758 | 0.742 | 0.731 | 0.718 | 0.652 | 0.705 | 0.701 |
| **Kappa** | 0.797 | 0.845 | 0.892 | 0.688 | 0.771 | 0.750 | 0.736 | 0.728 | 0.714 | 0.638 | 0.701 | 0.701 |

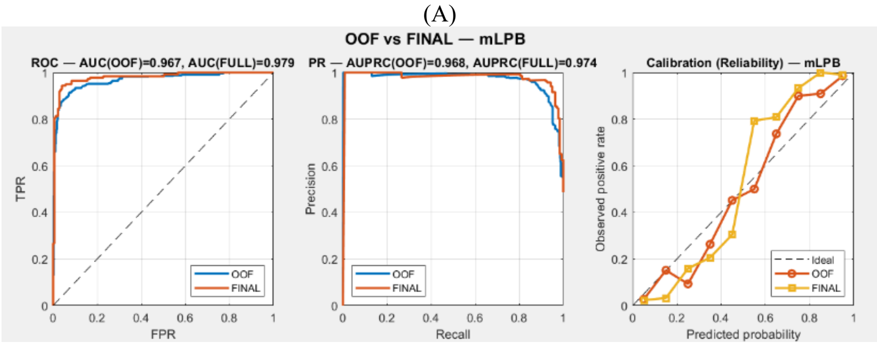
Under full-data training, mLPB-MLP is the best performer with a report of accuracy of 94.63, BCE = 0.194, and high discriminatory values of AUC = 0.979 and AUPRC = 0.974 on Dataset 1, as shown in Table 5. This is a trend in all datasets and indicates high convergence stability and better probability calibration than the LPB-MLP and aLPB-MLP. Together, the findings in Table 5 prove that mLPB-MLP provides the strongest discrimination, calibration, and generalization across all cross-validation and full-training situations and is hence the most trustworthy model when doing medical classification.

**6.6 Internal Validation Analysis and Out-of-Fold (OOF) Interpretation**

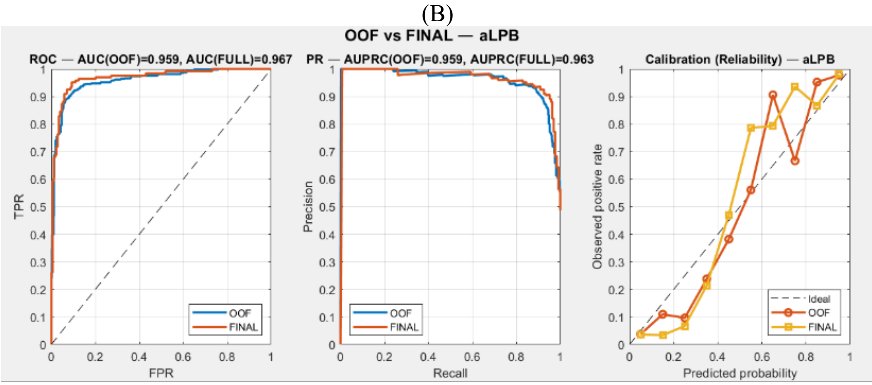
Models produce OOF predictions that are not based on the training samples of the same sample and therefore give an unbiased estimate of generalization. This lowers variance and assists robust threshold tuning and hyperparameter optimization, eliminating overly optimistic performance ratings [41].




(A)




(B)




(C)

***Fig.3 (A, B, and C):** Comparative Visualization of the Internal Performance of Dataset 1.*

The summary of internal validation of Dataset 1 through the three models of the LPB family is as represented in Fig. 3(A-C):

**Fig. 3(A) - LPB-MLP:** LPB highly discriminates (AUC, AUPRC > 0.94), and the calibration curve is in proximity to the desired diagonal, which is an indicator of reliable probability estimates.

**Fig. 3(B)-mLPB-MLP:** Best overall internal performance (AUC ≈ 0.967; AUPRC ≈ 0.97) and the nearest calibration to the ideal line, with the same probability alignment within each fold.

**Fig. 3(C)-aLPB-MLP:** Well-behaved calibration and high performance (AUC ≈ 0.96) without folds of validation.

In general, Fig. 3 indicates a steady high degree of discrimination, precision-recall behavior, and calibration throughout all models, with mLPB-MLP having the most sound and stable OOF generalization.

### 6.7 Graphical Results and Discussion.

In the next section, the graphical results of the internal validation experiments will be covered and will consist of visual analysis of the model discrimination, model calibration, and cross-fold stability. The resulting figures that follow the results display very critical, crucial behaviors of the LPB-family models on the ROC, PR, and calibration plots; cross-validation dashboards; and the best-fold assessment. 6.7.1 LPB-MLP Performance Overview.

#### 6.7.1 LPB-MLP Performance Overview

The cross-validation dashboard of LPB-MLP is provided in Fig. 4. that has six subpanels that lead in series (a) through (f) in left-to-right, top-to-bottom sequence. The curves of ROC and PR (Figs. 4 a-b) have a small cluster, which means that there are steady discrimination and positive-class behavior. Monotonic and aperiodic reduction of the loss curve is demonstrated using convergence curves (Fig. 4c). Variation at the fold level (Fig. 4d-e) is minimal with AUC/AUPRC = ± 0.01. The confusion matrix (Fig. 4f) located in the bottom-right panel, indicates an accuracy of around 88.49 with balanced FP/FN counts. Taken together, Figs. 4 verify that LPB-MLP has good discrimination (AUC = 0.94 or higher), good calibration, and consistent fold-wise performance, and no indication of overfitting.

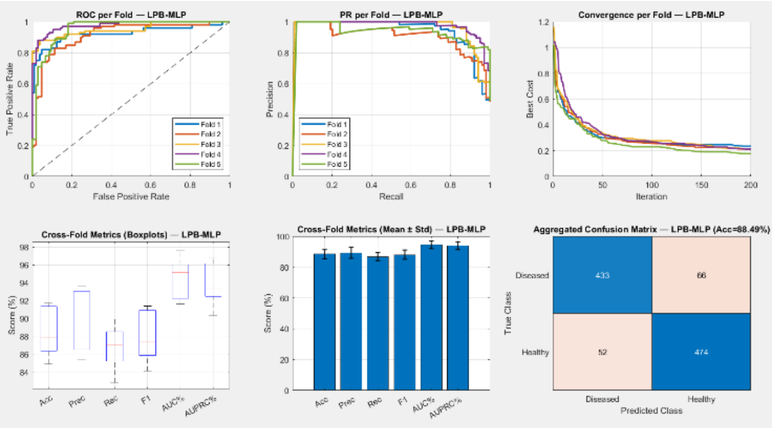



***Fig.4:*** *Cross-Validation for LPB–MLP*

#### 6.7.2 mLPB-MLP Performance Analysis

mLPB-MLP has the best internal performance, and the values of AUCOOF ≈ 0.967, AUCFINAL ≈ 0.979, and AUPRC ≈ 0.97-0.98 are very close to the theoretical diagonal. Fig. 5 indicates stable fold-wise ROC/PR clustering, fast convergence after 40 or so iterations, and stable metrics (accuracy, sensitivity, specificity, F1, AUC, and AUPRC all are in the 90-98 percent range). The confusion matrix (aggregated) reveals equal classification (near 91.8% accuracy). Therefore, Figs. 5 support the fact that mLPB-MLP is the most powerful model of discrimination, calibration, and cross-fold generalization of the models of the LPB family.

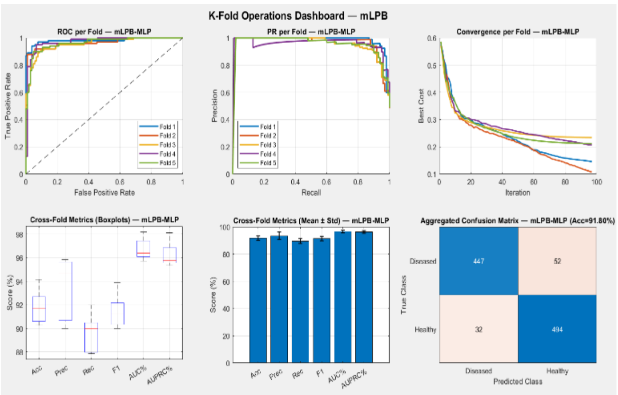



***Fig. 5:*** *Cross-Validation for mLPB-MLP*

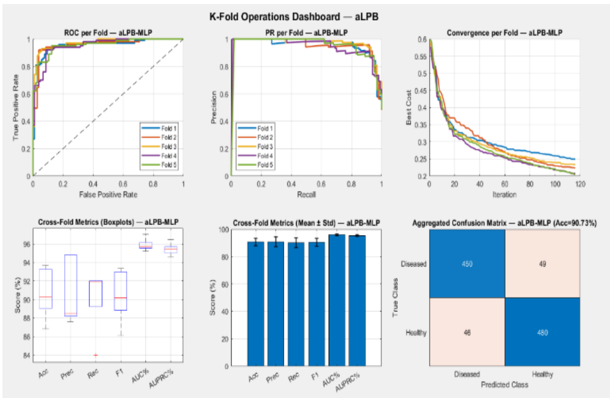



***Fig. 6:*** *Cross-Validation for aLPB-MLP*

#### 6.7.3 aLPB-MLP Performance

The aLPB-MLP model also has high generalization (AUCOOF=0.959 and AUCFINAL=0.967), good calibration, and close ROC/PR curves. The results of Fig. 6 present a quick convergence (in approximately 20-40 steps) and consistent fold-wise statistics (accuracy, recall, F1, AUC, and AUPRC are all in the 92-94% range). Balanced classification: the aggregated confusion matrix (~90.7% accuracy) indicates balanced classification. Figs. 6 show that aLPB-MLP also ensures high discrimination, convergence stability, and sound probability calibration, which makes the method suitable for sensitive medical diagnostics.

### 6.8 Comparative Assessment of the Three LPB-Family Models.

In all the graphical analyses and fold-wise assessments, mLPB-MLP consistently outperforms LPB-MLP and aLPB-MLP in discrimination, calibration, and generalization. Its ROC-AUC (≈ 0.98) and AUPRC (≈ 0.97-0.98) are also higher than the values of LPB-MLP (≈ 0.95) and aLPB-MLP (≈ 0.97), and its aggregated confusion matrices have the lowest false-positive and false-negative rates, resulting in the highest cross-fold accuracy of approximate of 91.8%. The mLPB-MLP calibration curves are also closest to the ideal reliability line, which is indicative of better probability calibration. Convergence patterns also indicate a higher rate and more constant optimization of mLPB-MLP and the lowest inter-fold variability, which is shown by boxplots and mean ± SD measures, indicating high robustness and low overfitting. Though each of the three models has good diagnostic capacity, mLPB-MLP is the most predictive and reliable and has the best balance of discrimination, stability, and calibration for heart-disease classification.

### 6.9 .Computational Implications and Practical Applicability.

The complexity of the suggested framework in terms of computational costs is primarily dictated by the evolutionary optimization of the LPB family applied in the training of neural weights and the Bayesian hyperparameter optimization in the outer loop. This cost is based on the number of generations, the population, and the network parameters. In spite of the fact that the training phase is computationally more challenging than the traditional gradient-based learning, this step is carried out offline as part of a model development. After training, the final MLP does inference by basic feedforward calculations, which are lightweight and can be used to do real-time prediction. Thus, the suggested framework can still be used in clinical decision-support systems where offline training is carried out and real-time prediction is conducted.

### 6.10 Sensitivity Analysis

In order to determine the strength of the proposed models, various types of sensitivity analyses were included in the experimental protocol. In particular, the stability of model performance across training folds was tested using stratified K-fold cross-validation, and the consistency of performance was tested using fold-level measures of variance (AUC/AUPRC variability). Moreover, the out-of-fold (OOF) threshold optimization and external dataset testing were used to determine how the models are sensitive to the decision thresholds and possible changes in the data distribution. All of these analyses suggest that proposed LPB-family models, especially mLPB-MLP, have consistent predictive behavior over different training and evaluation settings.

## 7. External Validation

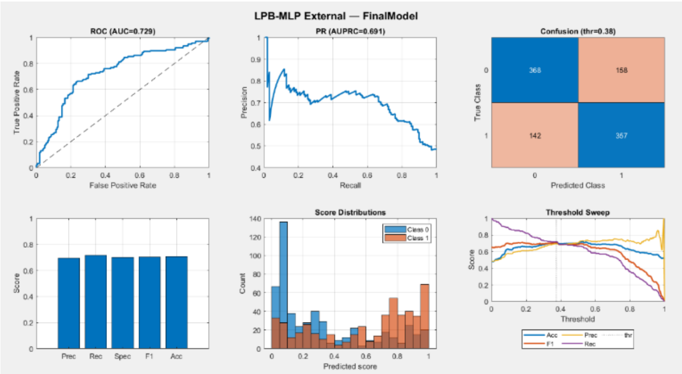
The last and most strict part of model evaluation is external validation, which involves the evaluation of the plausibility and usefulness of proposed neural models. This stage is a contrast to the internal cross-validation in which fully trained models are applied to an entirely independent dataset without retraining or optimization of parameters. External evaluation was conducted in this paper by applying the External Clinical Heart Disease dataset (2024) defined in Table 2 that includes 1048 samples and 12 clinical features gathered by an independent clinical source. This was not a dataset used in training or the cross-validation. Performing testing on this hidden data gives an impartial prediction of generalization performance and is resistant to population heterogeneity, institutional variation, and possible information leakage. Clinical predictive model reliability is one of the major areas of assessment by such external evaluation.

### 7.1 Consistency of Preprocessing and Evaluation, Workflow and Reproducibility.

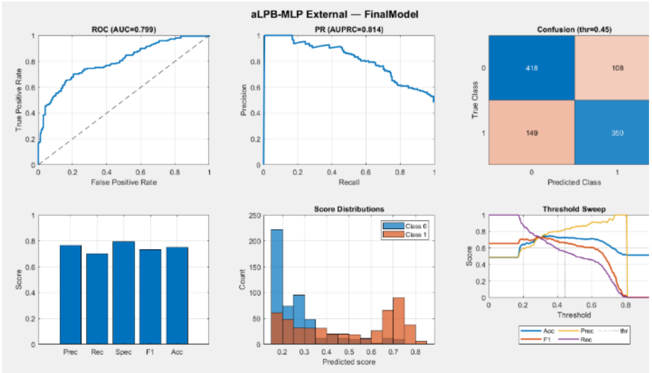
The external evaluation will duplicate the exact preprocessing pipeline that was used during training, and which includes:

- Normalization using Z-score without any reference to training-set statistics.
- Strict name-based feature alignment that includes one-hot encoded features.
- Automated correcting of polarities and fixing when required.

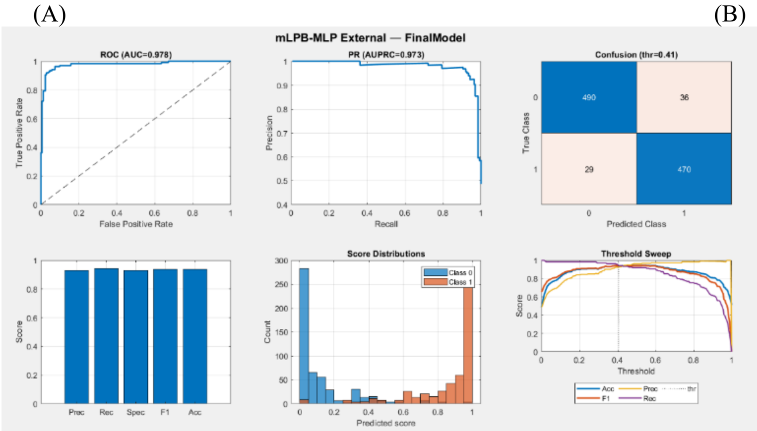
The same metric suite as internal validation supports the performance evaluation, and it is backed up with a 6-in-1 diagnostic panel. Outputs are exported in reproducible formats (JSON, mat) during experimentation and have high comparability and can be generalized reliably even out of the development environment.




(A)




(B)




(C)

***Fig. 7:** The LPB-MLP, aLPB-MLP, and mLPB-MLP Models External Test Results.*

Fig. 7 (A-C) demonstrates a comparative diagnostic visualization of the three LPB-family models, including the discrimination capability (ROC and PR curves), the class-level prediction behavior (confusion matrices), the calibration characteristics, the patterns of score distribution, and the stability of the threshold. Out of the assessed models, the strongest external generalization performance is indicated by the mLPB-MLP (Fig. 7C), which has the highest-performing ROC and PR curves, more distinct separation of predicted score distributions, and fewer cases of false-positive and false-negative predictions. The aLPB-MLP model (Fig. 7B) also demonstrates stable calibration and threshold behavior, but the discrimination has a slightly lower value compared with mLPB-MLP. Conversely, the weaker discrimination of the LPB-MLP model (Fig. 7A) and greater overlap of the distributions of predicted scores are natural because of its lower external predictive accuracy. On the whole, these findings support the observed performance improvements across the LPB-family variants, and mLPB-MLP offers the most consistent and clinical external classification of heart diseases. Each model was tested on separate external data that was preprocessed with the same standardized pipeline. The final predictive performance of LPB-MLP, aLPB-MLP, and mLPB-MLP is described in Table 6, which presents the results of the models in the primary evaluation metrics.

***Table 6:** External Evaluation (Final Models Comparison - LPB Family)*

| **Metric (%)*** | **LPB-MLP (Final)** | **aLPB-MLP (Final)** | **mLPB-MLP (Final)** |
|---|---|---|---|
| **Accuracy *** | 70.73 | 74.93 | 93.66 |
| **Precision *** | 69.32 | 76.42 | 92.89 |
| **Recall / Sensitivity*** | 71.54 | 70.14 | 94.19 |
| **Specificity*** | 69.96 | 79.47 | 93.16 |
| **F1-Score *** | 70.41 | 73.15 | 93.53 |
| **AUC (ROC)** | 0.7293 | 0.7993 | 0.9782 |

| | | | |
|---|---|---|---|
| **AUPRC** | 0.6908 | 0.8136 | 0.9732 |
| **BCE Loss** | 0.7398 | 0.4571 | 0.1934 |

The external outcomes prove that mLPB-MLP is the most powerful model with the greatest discrimination (AUC = 0.9782, AUPRC = 0.9732), the balanced sensitivity and specificity, and the lowest BCE loss, which means that the model shows better calibration. Though aLPB-MLP is better than LPB-MLP, it still is significantly worse than mLPB-MLP, which is the most valid and clinically appropriate model of predicting heart disease in the LPB family.

### 7.2 Reference Comparison with Previous Study

***Table 7.*** *Reference performance comparison*

| Model | FDO-MLP | LPBSA-MLP | FDO-CMLP | LPBSA-CMLP | LPB-CMLP | mLPB-MLP |
|---|---|---|---|---|---|---|
| **Accuracy (%)** | 96.09 | 100 | 91.85 | 98.37 | 97.06 | 94.63 |

To provide additional context for the obtained results, a reference comparison was made on the same heart-disease dataset (1025 samples) that was used in study [42]. As shown in Table 7, a direct comparison was made between the proposed mLPB-MLP model and the former reported models, because the mLPB-MLP model had the best performance as compared to the LPB-family variants. Although the study uses optimization-based neural training, models in this work conform to a more experimental protocol, such as having a single preprocessing pipeline, bi-level optimization based on Bayesian hyperparameter optimization with evolutionary weight learning, and rigid assessment in terms of cross-validation and external testing. This framework reduces overly optimistic estimates and provides a more reliable assessment of generalization performance. Thus, the recorded accuracies may appear less optimistic but are more realistic for predictive behavior.

### 8. Conclusion and Future Work

This paper offered an optimization framework of the LPB family, including LPB-MLP, mLPB-MLP, and aLPB-MLP, that incorporates the concept of Bayesian hyperparameter optimization alongside the concept of metaheuristic training to boost convergence and generalization in heart-disease classification. Experiments with several datasets showed a high level of accuracy, a high discriminatory level (AUC/AUPRC exceeding 0.95), and a good estimate of probabilities. mLPB-MLP was also the most effective in terms of balance of sensitivity, specificity, and convergence stability, and it can be inferred that mLPB-MLP is the most appropriate in clinical diagnostic modeling. This will be expanded to multi-class, multi-modal medical data, cross-institutional domain adaptation, and low-weight implementations to provide real-time clinical decision support in the future. Additionally, future efforts will focus on realistic implementation levels, such as cost of computation, integration of software, and organizational viability in practice in clinical settings. In general, the LPB-family framework offers a reproducible and scalable optimization-based methodology that offers to enhance intelligent medical diagnostics in terms of interpretability, efficiency, and clinical applicability.

## Acknowledgment

The authors are thankful for the patronage of the institutional assistance of the Sulaimani Polytechnic University and the Informatics Institute of the postgraduate studies of the University of Information Technology and Communications (UoITC), Baghdad, Iraq, that enabled the accomplishment of this research. The authors, too, wish to acknowledge the presence of the anonymous reviewers and their constructive and insightful comments that helped to enhance the quality, clarity, and rigor of this work to a considerable degree.

### Reference:

[1] S. M. Ali, S. Dick, S. Dillon, M. L. Jones, J. Penn, and R. Staley, "Histories of artificial intelligence: a genealogy of power," *BJHS Themes*, vol. 8, pp. 1–18, 2023.

[2] T. Nakaura, T. Higaki, K. Awai, O. Ikeda, and Y. Yamashita, "A primer for understanding radiology articles about machine learning and deep learning," *Diagn. Interv. Imaging*, vol. 101, no. 12, pp. 765–770, 2020.

[3] H. El-Sofany, B. Bouallegue, and Y. M. A. El-Latif, "A proposed technique for predicting heart disease using machine learning algorithms and an explainable AI method," *Sci. Rep.*, vol. 14, no. 1, p. 23277, 2024.

[4] A. Al Bataineh and S. Manacek, "MLP-PSO hybrid algorithm for heart disease prediction," *J. Pers. Med.*, vol. 12, no. 8, p. 1208, 2022.

[5] O. S. Albahri *et al.*, “Systematic review of artificial intelligence techniques in the detection and classification of COVID-19 medical images in terms of evaluation and benchmarking: Taxonomy analysis, challenges, future solutions and methodological aspects,” *J. Infect. Public Health*, vol. 13, no. 10, pp. 1381–1396, 2020.

[6] J.-R. Zhang, J. Zhang, T.-M. Lok, and M. R. Lyu, “A hybrid particle swarm optimization--back-propagation algorithm for feedforward neural network training,” *Appl. Math. Comput.*, vol. 185, no. 2, pp. 1026–1037, 2007.

[7] R. Kumar *et al.*, “A comprehensive review of machine learning for heart disease prediction: challenges, trends, ethical considerations, and future directions,” *Front. Artif. Intell.*, vol. 8, p. 1583459, 2025.

[8] A. C. Cinar, “Training feed-forward multi-layer perceptron artificial neural networks with a tree-seed algorithm,” *Arab. J. Sci. Eng.*, vol. 45, no. 12, pp. 10915–10938, 2020.

[9] B. Irmak and \cSaban Gülcü, “Training of the feed-forward artificial neural networks using butterfly optimization algorithm,” *MANAS J. Eng.*, vol. 9, no. 2, pp. 160–168, 2021.

[10] X. Zhang, X. Wang, H. Chen, D. Wang, and Z. Fu, “Improved GWO for large-scale function optimization and MLP optimization in cancer identification,” *Neural Comput. Appl.*, vol. 32, no. 5, pp. 1305–1325, 2020.

[11] W. Wienholt, “Minimizing the system error in feedforward neural networks with evolution strategy,” in *International conference on artificial neural networks*, 1993, pp. 490–493.

[12] U. Seiffert, “Multiple layer perceptron training using genetic algorithms.,” in *ESANN*, 2001, pp. 159–164.

[13] A. Abdurrakhman, L. Sutiarso, M. Ainuri, M. Ushada, and M. P. Islam, “A Multilayer Perceptron Feedforward Neural Network and Particle Swarm Optimization Algorithm for Optimizing Biogas Production,” *Energies*, vol. 18, no. 4, p. 1002, 2025.

[14] S. Mirjalili, S. Z. M. Hashim, and H. M. Sardroudi, “Training feedforward neural networks using hybrid particle swarm optimization and gravitational search algorithm,” *Appl. Math. Comput.*, vol. 218, no. 22, pp. 11125–11137, 2012.

[15] S. Mirjalili, S. M. Mirjalili, and A. Lewis, “Let a biogeography-based optimizer train your multi-layer perceptron,” *Inf. Sci. (Ny).*, vol. 269, pp. 188–209, 2014.

[16] S. Mirjalili, “How effective is the Grey Wolf optimizer in training multi-layer perceptrons,” *Appl. Intell.*, vol. 43, no. 1, pp. 150–161, 2015.

[17] S. Amirsadri, S. J. Mousavirad, and H. Ebrahimpour-Komleh, “A Levy flight-based grey wolf optimizer combined with back-propagation algorithm for neural network training,” *Neural Comput. Appl.*, vol. 30, pp. 3707–3720, 2018, [Online]. Available: https://api.semanticscholar.org/CorpusID:23425227

[18] F. Xu, C.-M. Pun, H. Li, Y. Zhang, Y. Song, and H. Gao, “Training Feed-Forward Artificial Neural Networks with a modified artificial bee colony algorithm,” *Neurocomputing*, vol. 416, pp. 69–84, 2020, [Online]. Available: https://api.semanticscholar.org/CorpusID:198466565

[19] A. A. Heidari, H. Faris, S. M. Mirjalili, I. Aljarah, and M. M. Mafarja, “Ant Lion Optimizer: Theory, Literature Review, and Application in Multi-layer Perceptron Neural Networks,” in *Nature-Inspired Optimizers*, 2019. [Online]. Available: https://api.semanticscholar.org/CorpusID:68242905

[20] S. Dalwinder, S. Birmohan, and K. Manpreet, “Simultaneous feature weighting and parameter determination of Neural Networks using Ant Lion Optimization for the classification of breast cancer,” *Biocybern. Biomed. Eng.*, vol. 40, pp. 337–351, 2020, [Online]. Available: https://api.semanticscholar.org/CorpusID:213414406

[21] H. Faris, I. Aljarah, and S. M. Mirjalili, “Training feedforward neural networks using multi-verse optimizer for binary classification problems,” *Appl. Intell.*, vol. 45, pp. 322–332, 2016, [Online]. Available: https://api.semanticscholar.org/CorpusID:10314269

[22] D. K. Abbas, T. A. Rashid, K. H. Bacanin, and A. Alsadoon, “Using fitness dependent optimizer for training multi-layer perceptron,” *arXiv Prepr. arXiv2201.00563*, 2022.

[23] P. Koukaras and C. Tjortjis, “Data Preprocessing and Feature Engineering for Data Mining: Techniques, Tools, and Best Practices,” *AI*, 2025, [Online]. Available: https://api.semanticscholar.org/CorpusID:281756295

[24] L. O. Joel, W. Doorsamy, and B. S. Paul, “A comparative study of imputation techniques for missing values in healthcare diagnostic datasets,” *Int. J. Data Sci. Anal.*, vol. 20, pp. 6357–6373, 2025, [Online]. Available:

https://api.semanticscholar.org/CorpusID:279338273

[25] U. Oza, B. Gohel, and P. Kumar, "Evaluation of Normalization Algorithms for Breast Mammogram Mass Segmentation," in *ICMLDE*, 2023. [Online]. Available: https://api.semanticscholar.org/CorpusID:270193226

[26] D. Iacobucci, S. S. Posavac, F. Kardes, M. J. Schneider, and D. Popovich, "Toward a More Nuanced Understanding of the Statistical Properties of a Median Split," *Econom. Econom. \& Stat. Methods - Gen. eJournal*, 2015, [Online]. Available: https://api.semanticscholar.org/CorpusID:10789659

[27] E. Poslavskaya and A. Korolev, "Encoding categorical data: Is there yet anything 'hotter' than one-hot encoding?," *ArXiv*, vol. abs/2312.1, 2023, [Online]. Available: https://api.semanticscholar.org/CorpusID:266573193

[28] O. Ghoneim, P. Dobiá\vs, and O. Romain, "Survey of neural network optimization methods for sustainable AI: From data preprocessing to hardware acceleration," *Mach. Learn. with Appl.*, 2025, [Online]. Available: https://api.semanticscholar.org/CorpusID:282303418

[29] S. Mohapatra, M. Panda, P. P. Sarangi, B. S. P. Mishra, N. Parihar, and B. Sahoo, "MLP-FOX: A Metaheuristic Approach-Based Training Multilayer Perceptron for Data Classification," *2024 1st Int. Conf. Cogn. Green Ubiquitous Comput.*, pp. 1–6, 2024, [Online]. Available: https://api.semanticscholar.org/CorpusID:269954273

[30] C. M. Rahman and T. A. Rashid, "A new evolutionary algorithm: Learner performance based behavior algorithm," *ArXiv*, vol. abs/2009.1, 2020, [Online]. Available: https://api.semanticscholar.org/CorpusID:221856669

[31] A. M. Ahmed and T. A. Rashid, "mLPB: A Modified Learner Performance-Based Behavior Algorithm Through Differential Evolution," *J. Soft Comput. Data Eng.*, 2025.

[32] A. M. Ahmed and T. A. Rashid, "Adaptive Crossover and Mutation Mechanisms for Enhanced LPB Algorithm Performance," *Iraqi J. Comput. Sci. Math.*, 2025, [Online]. Available: https://api.semanticscholar.org/CorpusID:281781901

[33] S. A. Salih and T. A. Rashid, "Learner Performance-Based Behavior Optimization Algorithm: A Functional Case Study," in *International Conference on Interactive Collaborative Robotics*, 2022.

[34] M. S. Advani and A. M. Saxe, "High-dimensional dynamics of generalization error in neural networks," *Neural Networks*, vol. 132, pp. 428–446, 2017, [Online]. Available: https://api.semanticscholar.org/CorpusID:38293418

[35] J. Snoek, H. Larochelle, and R. P. Adams, "Practical Bayesian Optimization of Machine Learning Algorithms," in *Neural Information Processing Systems*, 2012. [Online]. Available: https://api.semanticscholar.org/CorpusID:632197

[36] M. S. Advani, A. M. Saxe, and H. Sompolinsky, "High-dimensional dynamics of generalization error in neural networks," *Neural Networks*, vol. 132, pp. 428–446, 2020.

[37] D. Chicco and G. Jurman, "The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation," *BMC Genomics*, vol. 21, 2020, [Online]. Available: https://api.semanticscholar.org/CorpusID:209528322

[38] "Confusion Matrix-Based Performance Evaluation Metrics," *African J. Biol. Sci.*, 2024, [Online]. Available: https://api.semanticscholar.org/CorpusID:273625019

[39] T. F. Monaghan *et al.*, "Foundational Statistical Principles in Medical Research: Sensitivity, Specificity, Positive Predictive Value, and Negative Predictive Value," *Medicina (B. Aires).*, vol. 57, 2021, [Online]. Available: https://api.semanticscholar.org/CorpusID:235229140

[40] A. J. Viera and J. M. Garrett, "Understanding interobserver agreement: the kappa statistic.," *Fam. Med.*, vol. 37 5, pp. 360–363, 2005, [Online]. Available: https://api.semanticscholar.org/CorpusID:38150955

[41] I. Tsamardinos, E. Greasidou, M. Tsagris, and G. Borboudakis, "Bootstrapping the out-of-sample predictions for efficient and accurate cross-validation," *Mach. Learn.*, vol. 107, pp. 1895–1922, 2017, [Online]. Available: https://api.semanticscholar.org/CorpusID:13698873

[42] D. R. Hamad and T. A. Rashid, "LPBSA: Pre-clinical data analysis using advanced machine learning models for disease prediction," *Egypt. Informatics J.*, vol. 30, p. 100690, 2025.