algorithms

MDPI

Article

Enhancing Non-Invasive Blood Glucose Prediction from Photoplethysmography Signals via Heart Rate Variability-Based Features Selection Using Metaheuristic Algorithms

Saifeddin Alghlayini [1], Mohammed Azmi Al-Betar [1,2,*] and Mohamed Atef [3]

[1] Artificial Intelligence Research Center (AIRC), College of Engineering and Information Technology, Ajman University, Ajman P.O. Box 346, United Arab Emirates; 202211804@ajmanuni.ac.ae
[2] Hourani Center for Applied Scientific Research, Al-Ahliyya Amman University, Amman 19328, Jordan
[3] Department of Electrical and Communication Engineering, Collage of Engineering, United Arab Emirates University, Al Ain 15551, United Arab Emirates; moh_atef@uaeu.ac.ae
* Correspondence: m.albetar@ajman.ac.ae

Abstract: Diabetes requires effective monitoring of the blood glucose level (BGL), traditionally achieved through invasive methods. This study addresses the non-invasive estimation of BGL by utilizing heart rate variability (HRV) features extracted from photoplethysmography (PPG) signals. A systematic feature selection methodology was developed employing advanced metaheuristic algorithms, specifically the Improved Dragonfly Algorithm (IDA), Binary Grey Wolf Optimizer (bGWO), Binary Harris Hawks Optimizer (BHHO), and Genetic Algorithm (GA). These algorithms were integrated with machine learning (ML) models, including Random Forest (RF), Extra Trees Regressor (ETR), and Light Gradient Boosting Machine (LightGBM), to enhance predictive accuracy and optimize feature selection. The IDA-LightGBM combination exhibited superior performance, achieving a mean absolute error (MAE) of 13.17 mg/dL, a root mean square error (RMSE) of 15.36 mg/dL, and 94.74% of predictions falling within the clinically acceptable Clarke error grid (CEG) zone A, with none in dangerous zones. This research underscores the efficiency of utilizing HRV and PPG for non-invasive glucose monitoring, demonstrating the effectiveness of integrating metaheuristic and ML approaches for enhanced diabetes monitoring.

Keywords: non-invasive; blood glucose level (BGL); photoplethysmography (PPG); heart rate variability (HRV); feature selection (FS); metaheuristic algorithms; machine learning (ML); improved dragonfly algorithm (IDA); Clarke error grid (CEG); diabetes monitoring

check for updates

Academic Editor: Elena Tanfani

Received: 14 December 2024
Revised: 1 February 2025
Accepted: 4 February 2025
Published: 8 February 2025

Citation: Alghlayini, S.; Al-Betar, M.A.; Atef, M. Enhancing Non-Invasive Blood Glucose Prediction from Photoplethysmography Signals via Heart Rate Variability-Based Features Selection Using Metaheuristic Algorithms. *Algorithms* **2025**, *18*, 95. https://doi.org/10.3390/a18020095

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Diabetes is a chronic medical condition characterized by inadequate insulin production by the pancreas or the body's inability to effectively utilize insulin. This disorder is associated with persistent hyperglycemia, which can lead to severe complications, including cardiovascular disease, renal failure, neuropathy, and lower-limb amputations [1]. In 2021, over 537 million adults were diagnosed with diabetes, with projections indicating an increase to 783 million by 2045. Beyond the human impact, the economic burden of diabetes is significant, with global expenditure surpassing $955 billion annually [2].

Accurate monitoring of blood glucose levels (BGLs) is essential in the management of diabetes and in the prevention of complications associated with the disease. Conventional methods, such as clinical laboratory tests and glucose meters (glucometers), are characterized by invasive procedures that require blood samples, typically obtained from the

patient's finger. Although these methods are reliable, they are invasive, inconvenient, and unsuitable for continuous monitoring [3]. Consequently, there is a pressing demand for non-invasive, precise, and cost-effective alternatives for glucose monitoring.

Previous studies have demonstrated photoplethysmography (PPG) as a promising non-invasive technique for monitoring BGLs [4–12]. PPG is an optical technique that measures variations in blood volume by analyzing changes in light intensity as it passes through or reflects off the tissue. The simplicity, low cost, and ease of integration into wearable devices make PPG an attractive candidate for continuous glucose monitoring. Variations in BGL alter the optical properties of hemoglobin, affecting light absorption and scattering, which can be detected through PPG waveforms. Moreover, heart rate variability (HRV) metrics derived from PPG signals provide additional information on glucose fluctuations by reflecting autonomic nervous system activity [13–21]. Figure 1 shows how the PPG signal is acquired using a finger sensor equipped with a photodiode to capture infrared LED light.

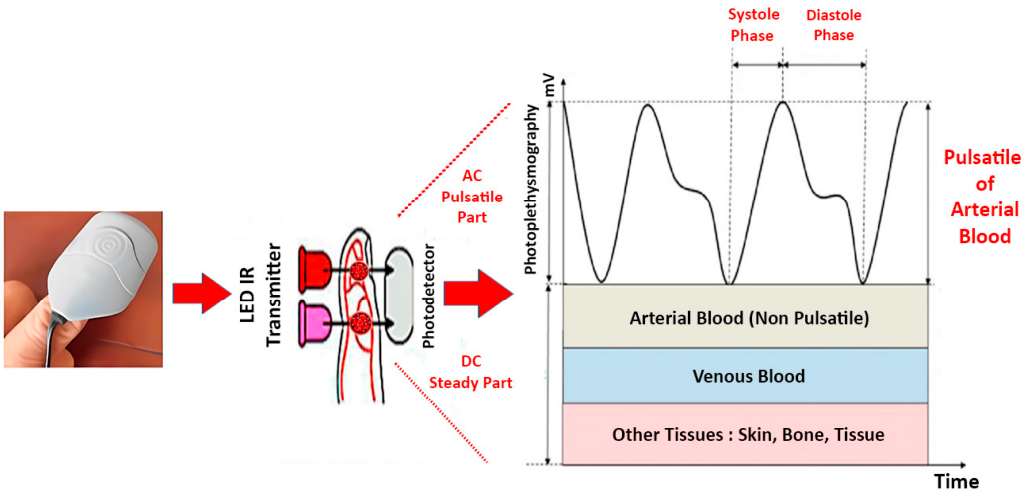


Figure 1. PPG measurement process [22].

By deriving the absorbance related to blood glucose from the AC component of the PPG wave, the BGL is calculated as follows [23]:

$$\triangle OD_{\lambda} = \log\left(1 + \frac{\triangle I_{\lambda}(t_i)}{I_{\lambda}(t_{i+1})}\right). \quad (1)$$

The difference ΔOD_{λ} represents the variance in optical densities at time t_i and t_{i+1}, while $\Delta I_{\lambda}(t_i)$ denotes the pulsatile element at time t_i, and $I(t_i + 1)$ indicates the light intensity at time t_{i+1}.

Despite its significant potential, the efficient application of PPG for glucose monitoring encounters substantial challenges. PPG signals are inherently high-dimensional, encompassing a variety of derived features, such as waveform attributes and HRV metrics. This increased dimensionality can lead to redundancy and noise, reducing the accuracy of predictive models. Previous studies have explored advanced machine learning (ML) approaches, including convolutional neural networks (CNN) and support vector regression (SVR), to analyze PPG data. However, these methods frequently employ complex algorithms and require large datasets, which limit their feasibility in non-clinical environments [4–12]. This highlights the need for preprocessing strategies to address these limitations effectively.

Feature selection (FS) is a vital preprocessing phase in ML-based systems, addressing these challenges by reducing dimensionality and selecting the most informative features.

FS methods can be classified based on two main factors: evaluation criteria and search strategies. In terms of evaluation, FS methods are generally categorized as filter or wrapper techniques. Filter methods, such as Chi-Square, Information Gain, and Gain Ratio, evaluate feature subsets based on their correlation with the target class, without relying on a specific learning algorithm. In contrast, wrapper methods use a learning algorithm in the evaluation process, generally producing more accurate results but requiring greater computational resources, especially for high-dimensional datasets. This computational burden requires innovative approaches to optimize the FS process. The performance of the wrapper methods also depends on the chosen learning algorithm [24].

Metaheuristic algorithms (MHs), inspired by natural phenomena, have demonstrated significant potential in addressing FS problems. These algorithms utilize dynamic search strategies to achieve a balance between exploration and exploitation, which makes them effective in navigating large search spaces [24]. Prominent MHs include Improved Dragonfly Algorithm (IDA), Binary Grey Wolf Optimizer (bGWO), Binary Harris Hawks Optimizer (BHHO), and Genetic Algorithm (GA). These algorithms have been successfully implemented in various optimization problems, including FS, where they efficiently identify subsets of near-optimal features [25–31].

In summary, while the current literature provides a robust foundation for understanding the potential of non-invasive BGL monitoring systems, there is a notable lack of studies that specifically employ advanced metaheuristic algorithms, such as IDA, in the FS process in this context. This research aims to bridge this gap by developing a robust model that enhances the accuracy of BGL estimation while reducing computational complexity. Through the integration of HRV features from PPG signals and systematic evaluation of model performance, the research aims to provide a feasible solution for non-invasive BGL monitoring, ultimately assisting individuals with diabetes in managing their condition.

This study presents a novel methodology that integrates machine learning with metaheuristic algorithms for feature selection and the estimation of non-invasive BGL using PPG signals and HRV features. The main contributions of this work are summarized as follows:

- Introduce a novel approach that utilizes PPG signals and HRV features to enable accurate non-invasive BGL measurement.
- Develop a robust wrapper-based feature selection technique that employs metaheuristic algorithms, including IDA, bGWO, BHHO, and GA, to enhance model performance by selecting significant features and minimizing computational complexity.
- Propose and evaluate a BGL estimation model that integrates IDA with Light Gradient Boosting Machine (LightGBM) alongside other regression models, including Random Forest (RF) and Extra Trees Regressor (ETR), demonstrating enhanced predictive accuracy and reliability for non-invasive glucose monitoring.

The subsequent sections of this paper are organized as follows: Section 2 provides a comprehensive review of related works, including PPG-based glucose monitoring and FS methodologies. Section 3 covers the background. The proposed methodology, including data preparation, FS processes, and regression modeling, is detailed in Section 4. Section 5 presents the experimental setup, results, and discussion. Lastly, Section 6 concludes the study and highlights directions for future research.

2. Related Work

2.1. PPG Signals for Non-Invasive Blood Glucose Estimation

This section summarizes recent studies on the use of PPG signals for non-invasive BGL estimation. By analyzing the characteristics of the PPG waveform, researchers have explored its potential as a tool for glucose monitoring through ML techniques.

Gupta et al. developed a system employing PPG data obtained from 26 people and processed it using an XGBoost (XGB) model, resulting in a clinically accurate glucose estimate [4]. However, the reliance on body mass index (BMI) and technology complexity presented issues. Similarly, Yen et al. used a back-propagation neural network (BPNN) with dual-wavelength PPG and bioelectrical impedance characteristics to obtain excellent accuracy, although they encountered limits in detection range and sensor integration complexity [5].

Islam et al. introduced a smartphone-based solution, converting fingertip videos into PPG signals for BGL prediction using regression models. While effective, this method depended on high-quality smartphone sensors, limiting accessibility [6]. Fang and Liao utilized a CNN-LSTM hybrid model for temporal PPG analysis, achieving 99% accuracy in clinically acceptable zones but were constrained by small datasets and manual hyperparameter tuning [8].

Li et al. introduced a novel multimodal framework for estimating BGL using PPG and electrocardiogram (ECG) data. Using deep neural networks (DNNs) and spatiotemporal decision fusion techniques, their method achieved Zone A + B accuracy of 99.49% but faced scalability challenges due to the small number of participants and the complexity of combining ECG and PPG signals [7]. Shuzan et al. combined PPG signals, blood pressure data, and demographic information into a Bagged Ensemble Trees (BET) model in another research. They achieved 61.88% predictions in Zone A of the Clarke Error Grid (CEG), but reliance on manually inputted blood pressure data limited practicality [10].

More recently, Chowdhury et al. created MMG-Net, a multimodal network that uses PPG, electrodermal activity (EDA), and skin temperature data to improve BGL prediction. Their method achieved great accuracy but raised concerns regarding real-world viability due to the reliance on several sensors and the limited sample size [12].

These studies demonstrate the potential of PPG signals for BGL monitoring while also emphasizing constraints such as dataset size, sensor requirements, and the need for comprehensive feature selection to increase prediction accuracy and efficiency.

2.2. Heart Rate Variability and Blood Glucose Monitoring

In this section, we discuss the literature that investigated the association of HRV with BGLs with a focus on the use of HRV as a non-invasive measure of BGL.

Martinez-Delgado et al. combined HRV and pulse oximeter data, achieving a balanced accuracy of up to 82.1% in predicting high glucose levels. Singh et al. documented inverse correlations between HRV parameters, such as SDNN and LF/HF ratio, and fasting glucose levels in diabetic populations [13]. Tarvainen et al. observed that prolonged hyperglycemia reduces HRV, suggesting its potential as a marker for long-term glucose regulation [14].

Vishinov et al. demonstrated that specific HRV parameters, such as SDRMSSD-3, achieved 86.52% accuracy in distinguishing between good and poor glucose regulation, providing evidence of HRV's utility for classifying glycemic control levels [17]. Also, Chen et al. reported that higher glycemic fluctuations were associated with significantly lower HRV parameters, particularly SDNN, in patients with type 2 diabetes and coronary heart disease. These findings underscore the clinical implications of HRV for managing diabetes and cardiovascular complications [16].

Furthermore, Aggarwal et al. used HRV-based artificial neural networks (ANN) and support vector machines (SVM) to classify diabetic conditions in rats, achieving classification accuracies above 95%. This study highlights the feasibility of HRV features for BGL estimation and diabetes prognosis in controlled environments [21].

Although these studies highlight the promise of HRV for glucose monitoring, issues including individual variability, small dataset sizes, and the complex relationship between HRV and BGL levels still need to be addressed.

2.3. Metaheuristic Algorithms for Feature Selection

This section summarizes recent advancements in using MHs for FS, focusing on the four algorithms utilized in this study: IDA, bGWO, BHHO, and GA.

IDA builds on the original Dragonfly Algorithm by introducing enhancements to address premature convergence and improve global search capabilities. IDA performs especially well in FS for high-dimensional datasets, as shown in the studies where it yielded better results compared to conventional algorithms such as Particle Swarm Optimization (PSO) and the standard Dragonfly Algorithm (DA) in terms of feature reduction and accuracy [25,26].

The bGWO adapts the social hierarchy and hunting behavior of gray wolves to binary FS problems. This algorithm efficiently balances exploration and exploitation, making it well-suited for high-dimensional medical datasets such as classifying diabetes and cancer. Studies have shown that bGWO outperforms other algorithms, including GA and PSO, by reducing feature redundancy while maintaining classification accuracy [27–29].

The BHHO, inspired by the cooperative hunting strategies of Harris hawks, has been effectively applied to FS tasks in biomedical datasets. BHHO uses adaptive strategies to explore the search space, demonstrating strong performance in selecting relevant features and achieving high classification accuracy. Its binary adaptation makes it highly effective in discrete optimization tasks like FS [30,31].

The GA remains a foundational optimization method, drawing inspiration from natural selection. GA employs crossover, mutation, and selection operators to explore feature subsets. Despite its simplicity, GA remains a benchmark in FS research and has been widely used in various domains, including microarray data and medical applications, where it has shown reliable performance in reducing dimensionality. Majdi M. Mafarja et al. demonstrated GA's consistent capability to provide reliable feature subset selections, though occasionally falling short of the advancements made with BHHO and IDA in terms of convergence speed and accuracy [26].

These four algorithms have shown that MH methods can be generally applied and efficiently solve FS problems. This study utilizes IDA, bGWO, BHHO, and GA to tackle high-dimensionality challenges in glucose monitoring datasets for non-invasive estimation of glucose level.

2.4. Summary of Challenges and Gaps

This section highlights the key challenges and research gaps in the existing literature on non-invasive glucose monitoring and FS. While PPG signals and HRV features have demonstrated potential for glucose estimation, many studies are limited by small datasets, sensor variability, the complexity of multisensor setups, and restricted detection ranges. Integrating additional physiological parameters, such as bioelectrical impedance, has improved performance in some cases but increased computational complexity.

The research on HRV-based glucose monitoring systems, although promising, is still in its early stages. There is a clear need for more studies that focus on optimizing machine learning models for HRV feature selection and addressing challenges related to data variability and sensor precision. Additionally, the application of metaheuristic algorithms to HRV-based glucose monitoring remains underexplored, particularly for regression tasks, which are crucial for continuous glucose estimation.

Overcoming these challenges is critical for improving the robustness and reliability of non-invasive glucose monitoring systems. This study sets itself apart by utilizing machine learning models combined with metaheuristic algorithms to enhance feature selection for HRV-based glucose monitoring. By focusing on reducing computational complexity while maintaining high accuracy with PPG signals alone, this research addresses critical gaps in the literature and contributes to the development of scalable and efficient non-invasive glucose monitoring solutions.

3. Background

This section outlines the theoretical foundations underlying the methodologies and techniques used in this study. It begins with the formulation of the FS problem, followed by a detailed discussion of the four MHs employed, including IDA, BHHO, bGWO, and GA. These algorithms are presented with an emphasis on their mechanisms, mathematical formulations, and their relevance to the FS process.

3.1. Feature Selection Problem Formulation

Feature selection is an optimization task aimed at reducing the dimensionality of datasets by identifying the most relevant features while eliminating redundant or irrelevant ones. This problem can be represented mathematically as a binary combinatorial optimization problem. Each candidate solution is encoded as a binary vector, where 1 represents the inclusion of a feature and 0 represents its exclusion [24]. Figure 2 illustrates this binary representation.

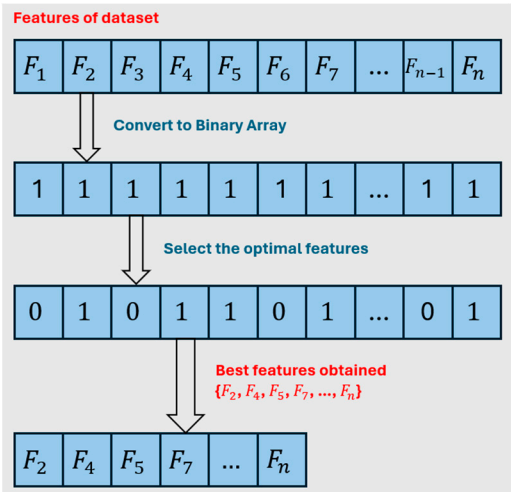


Figure 2. The binary representation of the FS problem.

Consider a dataset D comprising N features, represented as D = $\{f_1, f_2, \ldots, f_N\}$. The goal of FS is to extract a subset S, where S has fewer features than N. The aim of the optimization process is to determine which subset S, which is frequently determined by a particular objective function, maximizes model performance. The FS process lowers computing complexity, enhances predictive performance, and reduces the possibility of overfitting in ML models [24].

FS is an NP-hard problem due to the exponential growth of possible subsets 2ˆN with the number of features N [32,33]. So, exhaustive search is computationally impractical, making optimization algorithms indispensable for solving FS problems efficiently. This limitation underscores the value of metaheuristic algorithms, which leverage stochastic search strategies to efficiently explore large solution spaces and identify near-optimal solutions. Inspired by natural processes, these algorithms have demonstrated superior effectiveness

in addressing the combinatorial challenges of FS, often surpassing the performance of exhaustive or greedy methods [34–36].

According to the No-Free-Lunch (NFL) theorem, no single optimization algorithm is guaranteed to work best for all problems. This means that while metaheuristics have shown success in FS, there is always potential for better results with the development of new algorithms tailored to specific problems [37].

3.2. Improved Dragonfly Algorithm (IDA)

IDA enhances the original Dragonfly Algorithm (DA) by introducing advanced updating mechanisms for the algorithm's coefficients. In the DA, the coefficients are updated using random functions, which can lead to suboptimal convergence and exploration-exploitation balance. IDA addresses this limitation by employing structured updating mechanisms, including linear, quadratic, and sinusoidal functions, to refine the coefficients. This improvement allows IDA to achieve a better balance between exploration and exploitation, which is crucial to effectively navigate the high-dimensional feature space encountered in FS tasks [26].

Inspired by dragonflies' swarming behaviors, IDA models two key behaviors: hunting and migration, governed by principles such as separation (S), alignment (A), cohesion (C), attraction (F), and distraction (E). These behaviors guide the search for optimal solutions. To adapt continuous updates for binary FS tasks, IDA employs a V-shaped transfer function, which effectively transforms continuous position updates into binary solutions. Position updates are calculated iteratively using Equations (2) and (3).

$$\Delta X_{t+1} = (sS_i + aA_i + cC_i + fF_i + eE_i) + w\Delta X_t \,, \tag{2}$$

$$X_{t+1} = X_t + \Delta X_{t+1} \,, \tag{3}$$

where s, a, c, f, and e are weights associated with the operators, and w is the inertia weight. The position of an individual is updated as in Equation (3), where t is the present step.

The process begins with initializing key parameters, including the number of dragonflies (population size), the maximum iteration number, and coefficient values for dragonfly behaviors such as separation, alignment, and cohesion. These coefficients are initialized based on the migration models: Sinusoidal-based (SBDA), Quadratic-based (QBDA), and Linear-based (LBDA). Each migration model defines the initialization and evolution of coefficients. For instance, the sinusoidal model uses trigonometric functions to oscillate coefficients between defined minimum and maximum values, ensuring dynamic adaptation. These migration models guide the dragonflies' interactions, maintaining a balance between exploration and exploitation as they search for optimal feature subsets. The algorithm then randomly generates an initial population of dragonflies to ensure diversity, with each individual representing a potential feature subset.

During each iteration in the iterative optimization phase, the positions of dragonflies are adjusted based on their distances from each other, enhancing their collaborative search behavior. The fitness of each dragonfly is evaluated using a fitness function, directing the search toward optimal solutions. The IDA updates the coefficients using one of the migration models. Additionally, the operators representing dragonfly behaviors, including attraction (F) and distraction (E), are calculated using specific equations described in [26]. The step vectors and positions of dragonflies are updated accordingly, driving the optimization process toward convergence. Once the iterations are complete, the algorithm selects and returns the best feature subset, representing the most optimal solution. The IDA's workflow is detailed in Algorithm 1.

Algorithm 1 Improved Dragonfly Algorithm (IDA)

1. Initialize the population Xi (i = 1, 2, . . ., pop size), with each dragonfly representing a potential solution (feature subset)
2. Initialize Step vector ΔXi (i = 1, 2, . . ., n), determining the movement update for each dragonfly
3. Initialize coefficients (w, s, a, c, f, e) based on the chosen migration model (LBDA, QBDA, SBDA), which defines how these coefficients evolve dynamically
4. while (end condition is not satisfied) do
5. Evaluate the fitness of each dragonfly by computing the objective function value for its current position
6. Update coefficients based on the selected migration model, ensuring a balance between exploration and exploitation
7. Calculate S, A, C, F, and E based on distances and relative positions of neighboring dragonflies, adjusting movement influence dynamically
8. Update the step vector ΔXt + 1 using Equation (2)
9. Binarize the population using the V-shaped transfer function, converting continuous values into binary (0 or 1) for feature selection
10. Update the positions Xt + 1 of dragonflies using Equation (3) to refine feature selection
11. Return the best solution (the optimal feature subset) based on fitness evaluation

IDA has shown superior performance in FS tasks, especially for biomedical datasets, in which it increases the classification accuracy. Because it is able to balance exploration and exploitation very well, it is also very stable and efficient for high-dimensional optimization tasks.

3.3. Binary Harris Hawks Optimizer (BHHO)

BHHO is a binary adaptation of the Harris Hawks Optimizer (HHO), tailored for FS in high-dimensional datasets. Inspired by the hunting strategies of Harris hawks, BHHO combines exploration and exploitation phases to identify optimal solutions [30]. During exploration, hawks search broadly across the solution space, with position updates governed by Equation (4).

$$X(t+1) = \begin{cases} X_{rand}(t) - r_1 \left|X_{rand}(t) - 2r_2 X(t)\right|, \ q \geq 0.5 \\ (X_{rabbit}(t) - X_m(t)) - r_3(LB + r_4(UB - LB)), \ q < 0.5 \end{cases}. \tag{4}$$

Here, X(t + 1) is the updated hawk position, $X_{rabbit}(t)$ represents the best solution (rabbit's position), $X_{rand}(t)$ denotes a random hawk's position, r_1, r_2, r_3, r_4 are random variables, and LB and UB are the lower and upper bounds of the search space.

The transition from exploration to exploitation is determined by the energy level (E), modeled using Equation (5).

$$E = 2E_0\left(1 - \frac{t}{T}\right), \tag{5}$$

where E_0 is a random initial energy value, t is the current iteration, and T is the maximum number of iterations.

The exploitation phase refines solutions using strategies like soft and hard besiege. Soft besiege is represented by Equation (6), while hard besiege uses Equation (7).

$$X(t+1) = \Delta X(t) - E|JX_{rabbit}(t) - x(t)|\,, \tag{6}$$

$$X(t+1) = X_{rabbit}(t) - E|\Delta X(t)|, \tag{7}$$

where $\Delta X(t)$= $X_{rabbit}(t) - X(t)$ and J is the jump strength, which defines the magnitude of hawks' movements toward or away from the rabbit. The energy level (E) plays a pivotal role in balancing exploration and exploitation, representing the hawk's capacity to transition between these phases. Both E and J are initially randomly generated to ensure variability and enhance the optimization process. During iterations, the energy level gradually decreases to shift focus from exploration to exploitation, while J is updated based on random factors and the hawk's proximity to the best-known solution.

The algorithm begins by initializing key parameters, such as population size and maximum iterations, followed by generating a diverse population of hawks represented as continuous solutions. In each iteration, the fitness of each hawk is calculated, and positions are updated based on energy levels and the best-known solutions. Exploration occurs when the energy level |E| is greater than or equal to one, while exploitation strategies, including soft besiege, hard besiege, and rapid dives, are applied when |E| is less than one based on a random value r.

Continuous positions are transformed into binary vectors using transfer functions, including S-shaped and V-shaped functions, to meet the binary constraints of FS tasks. As detailed in Algorithm 2, this iterative process continues until a stopping condition, such as reaching the maximum number of iterations, is met, yielding the best solution, $\boldsymbol{X_{rabbit}}$, which represents the optimal feature subset. BHHO's dynamic search capability makes it highly effective for FS, enhancing ML models in applications such as gene selection and image classification.

Algorithm 2 Binary Harris Hawks Optimizer (BHHO)

```
Initialize Xi (i = 1, 2, ..., pop_size), with each hawk representing a candidate
solution in the search space, encoded as a continuous position vector
while stopping condition is not met do
    Calculate fitness values by evaluating the objective function for each hawk Set
Xrabbit as the best solution found so far based on fitness
    for each hawk (Xi) do
        Update E0 and jump strength J ▷ E0 = 2rand() − 1, J = 2(1− rand())
        Update energy E using Equation (5)
        if |E| ≥ 1 then
            Exploration phase: Update location vector using Equation (4)
        else
            Exploitation phase: Update location vector based on random r and energy
level E using Equations (6) and (7)
    Binarize the population using transfer functions, including S-shaped and
V-shaped func-tions, to convert continuous position values into binary values (0 or
1) for feature selection
Return Xrabbit as the best solution, representing the optimal feature subset
```

3.4. Binary Grey Wolf Optimization (bGWO)

bGWO mimics the hierarchical and cooperative hunting behavior of gray wolves to optimize FS. This metaheuristic algorithm adapts the original Grey Wolf Optimization to binary search spaces using a sigmoidal transfer function, enabling effective navigation of feature subsets. The sigmoidal function, shown in Equation (8), converts continuous positions into probabilities. Allowing the updated positions to be expressed as binary values using Equation (9), where *rand* is a random number between 0 and 1, and x_d^{t+1} is the updated binary position in dimension d at iteration t. X_α, X_β, and X_δ are the positions of the top three wolves [27].

$$sigmoid(a) = \frac{1}{1 + e^{-10(x-0.5)}}, \tag{8}$$

$$x_d^{t+1} = \begin{cases} 1, \ if \ sigmoid\left(\frac{x_\alpha + x_\beta + x_\delta}{3}\right) \geq rand \\ 0, \ otherwise \end{cases}. \tag{9}$$

The process starts with initializing a population of n gray wolves, each assigned a random binary position within the range [0, 1], ensuring diverse starting points for an effective exploration of the solution space. After initialization, the algorithm evaluates the fitness of each wolf ($\boldsymbol{wolf_i}$) to identify the best solutions, labeled as α, β, and δ, representing the top three solutions. This identification is crucial for guiding the search towards optimal feature subsets.

During each iteration, the position of every wolf in the pack is updated using the sigmoidal function and stochastic thresholding to ensure binary representation. Parameters *a*, A, and C are updated iteratively to balance exploration and exploitation using Equations (10), (11), and (12), respectively. Specifically, parameter *a* is linearly updated in each iteration to range from 2 to 0, controlling the trade-off between exploration and exploitation. After updating positions, the fitness of all wolves is re-evaluated, and the best solutions, α, β, and δ, are updated accordingly.

$$a = 2 - t\frac{2}{T}, \tag{10}$$

$$A = 2a\,(r_1) - a, \tag{11}$$

$$C = 2(r_2), \tag{12}$$

where t is the current iteration, T is the maximum number of iterations, and $\boldsymbol{r_1}$ and $\boldsymbol{r_2}$ are random numbers in the range [0, 1]. These parameters collectively refine the wolves' positions and encourage variability, enhancing search efficiency.

The algorithm iterates until the stopping criteria are met, which may include reaching the maximum number of iterations or achieving a predefined fitness threshold. Once the stopping condition is satisfied, the procedure returns the optimal binary position $\boldsymbol{x_a}$ and the best fitness value $f\,(\boldsymbol{x_a})$.

This structured approach enables the bGWO algorithm to effectively navigate the feature selection space, using the social dynamics of gray wolves to find optimal or near-optimal solutions. The pseudocode in Algorithm 3 details this process, emphasizing the role of the sigmoidal transfer function in balancing exploration and exploitation. By effectively reducing feature dimensions, bGWO excels in high-dimensional datasets, making it a powerful tool for FS.

Algorithm 3 Binary Grey Wolf Optimization (bGWO)

1. Initialize the population of n wolves randomly within [0,1], where each wolf represents a candidate solution encoded as a position vector
2. Find the α, β, and δ solutions by evaluating the fitness of all wolves and selecting the top three solutions as leaders
3. while (stopping condition is not met) do
4. for each wolfi in the pack do
5. Update wolfi position to a binary position according to Equation (9), applying the sigmoidal transfer function and a random threshold to ensure discrete feature selection
6. Update the adaptive coefficients a, A, and C coefficients using Equations (10) to (12) to balance exploration and exploitation
7. Evaluate the new positions by computing the fitness of each wolf after position updates
8. Update α, β, and δ solutions, replacing them if a better solution is found based on fitness values
9. Return the best solution (xα), representing the optimal feature subset with the highest fitness value

3.5. Genetic Algorithms (GAs)

GAs are optimization techniques inspired by natural selection, where potential solutions, represented as binary chromosomes, evolve through iterative selection, crossover, and mutation processes to explore and exploit the search space efficiently. Guided by a fitness function, typically based on classification accuracy in FS tasks, GAs effectively explore high-dimensional search spaces [38,39].

Fitter individuals, determined by a fitness function fi, are more likely to be selected for reproduction. Selection probabilities, calculated using Equation (13), prioritize these individuals. The offspring of the two parents is created through crossovers described in Equation (14), whereas mutation described in Equation (15) brings new diversity in the genes. These mechanisms lead the population to optimal solutions.

$$S_i = \frac{f_i}{\sum_{j=1}^{P} f_i} \tag{13}$$

$$C' = crossover(C_1, C_2)\ , \tag{14}$$

$$C' = \begin{cases} C\ with\ probability\ 1 - M \\ mutate(C)\ with\ probability\ M \end{cases} \tag{15}$$

Here, C represents the original chromosome, and C′ is the resulting chromosome after mutation.

The algorithm begins with a randomly initialized population, where each individual represents a subset of features encoded as a binary vector. Fitness evaluation guides the selection process, using techniques such as roulette wheel or tournament selection to prioritize individuals with higher scores. Offspring are generated through crossover operations that exchange feature subsets between parents, while mutation introduces random changes, preserving genetic diversity and preventing premature convergence. Crossover and mutation operations are controlled by predefined probabilities, ensuring a balance between exploration and exploitation.

This iterative process tracks the best-performing individual across generations and repeats until a termination condition, such as reaching the maximum iterations, is met. The final output is the optimal feature subset that represents the best solution identified. Algorithm 4 illustrates the GA workflow.

Algorithm 4 Genetic Algorithm (GA) for Feature Selection

1. Initialize population P with random feature subsets as binary vectors
2. Evaluate the fit-ness of each individual in P using the fitness function
3. while stopping condition is not met do
4. Select parents from P based on fitness, using selection mechanisms such as roulette wheel or tournament selection, where individuals with higher fitness have a greater chance of being chosen
5. Apply crossover operation to selected parent pairs with a predefined crossover probabil-ity, generating offspring by exchanging genetic information (feature subsets)
6. Apply mutation to the offspring with a predefined mutation probability, introducing ran-dom changes to maintain diversity and prevent premature convergence
7. Evaluate the fitness of the new individuals to determine their quality as potential solu-tions
8. Replace old population with the newly generated population, selecting the best individu-als to guide the evolutionary process
9. Return the best individual as the optimal feature subset with the highest fitness

Due to the GAs ability to avoid local optima and the fact that they are efficient in high-dimensional search space, GAs are suitable for FS in bioinformatics and other domains such as image processing. However, careful parameter tuning is required to strike a balance between computational cost and convergence speed.

4. Methodology

This section presents the detailed methodology employed in this study to develop a non-invasive BGL estimation model using PPG signals and HRV features. The methodology comprises three primary stages: data preparation and feature extraction, FS using metaheuristic algorithms, and regression-based model evaluation. The general workflow is depicted in Figure 3, highlighting the contribution of this research in optimizing FS using metaheuristic algorithms.

4.1. Data Preparation

4.1.1. Dataset Description

The public dataset [6] utilized in this research comprises PPG signals collected from 52 subjects, including both male and female participants aged between 17 and 61 years, with a male-to-female ratio of 75:25. Each PPG signal was recorded for one minute with a sampling rate of 30 Hz, along with a reference BGL measurement using a commercially available glucometer (Accu-Check). The BGL measurements range from 68 to 211 mg/dL, which is adequate to evaluate the model's performance effectively.

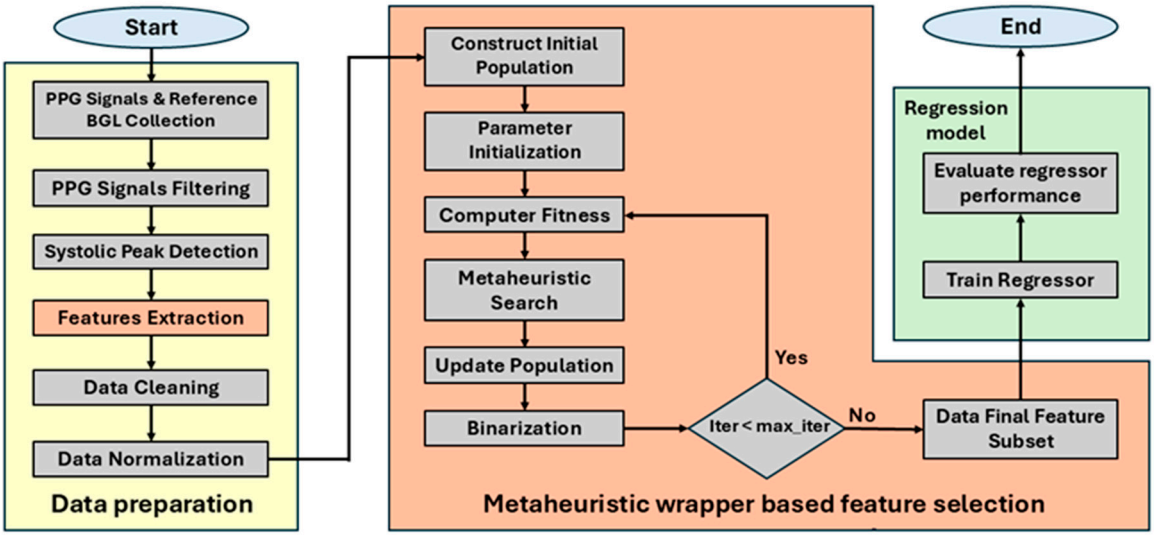


Figure 3. General scheme of the proposed model for non-invasive BGL estimation where the study contribution is highlighted in red color.

To ensure signal quality, the first and last 5 s of the recordings were excluded, resulting in a 20 s segment for analysis. The dataset contains 197 trials, and all subjects have completed at least three trials. This curated dataset offers an appropriate foundation for the validation of the proposed model.

4.1.2. Signal Preprocessing

The preprocessing stage aims to clean the raw PPG signals by removing noise, artifacts, and baseline shifts caused by motion and physiological interference. Examples of raw signals with varying quality are shown in Figure 4. Preprocessing was performed using the NeuroKit2 library [40], employing Elgendi's method [41] to enhance signal quality. A third-order Butterworth bandpass filter (0.5–8 Hz) was applied to remove baseline wander and high-frequency noise. This filter was chosen for its maximally flat response, which preserves essential features such as systolic peaks while minimizing distortion. The bandpass Butterworth filter is mathematically represented in Equation (16) [42],

$$H(s) = \frac{\omega_c^{2n}}{s^{2n} + \sqrt{2}\omega_c^n s^n + \omega_c^{2n}}, \tag{16}$$

where $\omega_c = \sqrt{\omega_L \cdot \omega_H}$ represents the center frequency derived from the low and high cutoff frequencies, s represents the complex frequency variable, and n is the filter order. After filtering, signals were evaluated for quality, and those with unidentifiable key features were discarded.

After filtering, signals were evaluated for quality, and corrupted signals were excluded based on the clarity of vital features like systolic peaks. Cleaned signals were processed for peak detection using Elgendi's algorithm [41], as outlined in the flowchart in Figure 5. The algorithm amplifies systolic peaks by squaring the positive signal values, applies moving averages for smoothing and adaptive thresholding, and identifies the most prominent peak within each waveform. To ensure accuracy, a minimum interval between consecutive peaks was enforced, preventing false detections within a single cardiac cycle.

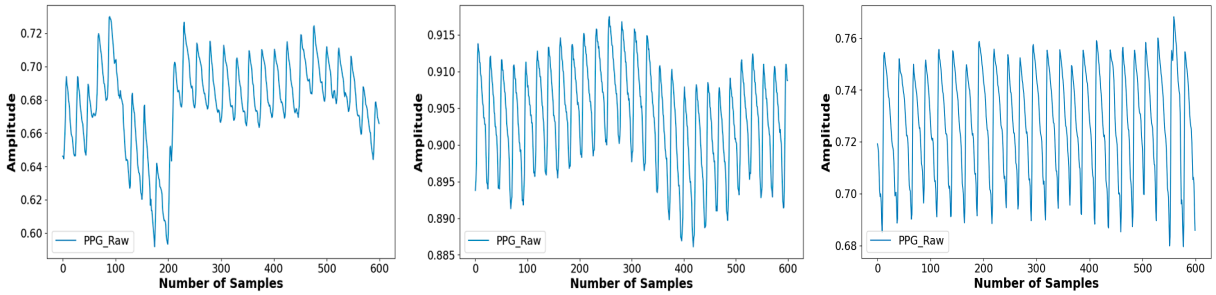


Figure 4. PPG signals with different qualities: a signal with both baseline variations and high-frequency noises (**left**), a signal with baseline variations (**middle**), and a good one with a little baseline variation (**right**).

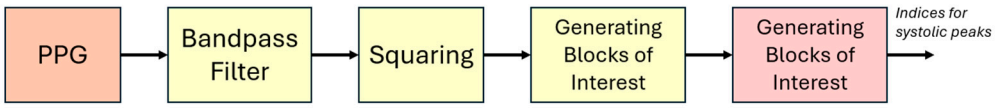


Figure 5. Flowchart for systolic peak time-domain detection algorithm consists of three main stages: preprocessing (bandpass filter), feature extraction (two moving averages), and classification (threshold).

Finally, the processed signals were analyzed to calculate the time interval (period) between consecutive systolic peaks. This interval was used to derive the Heart Rate (HR) in beats per minute (bpm) by dividing 60 by the period. From the calculated HR, the PPG Rate Mean (average HR) and PPG Rate Standard Deviation were determined, providing essential metrics for subsequent analysis. Figure 6 illustrates this process, highlighting how these metrics are extracted from the pre-processed PPG signals, ensuring the data are effectively prepared for feature extraction and machine learning analysis.

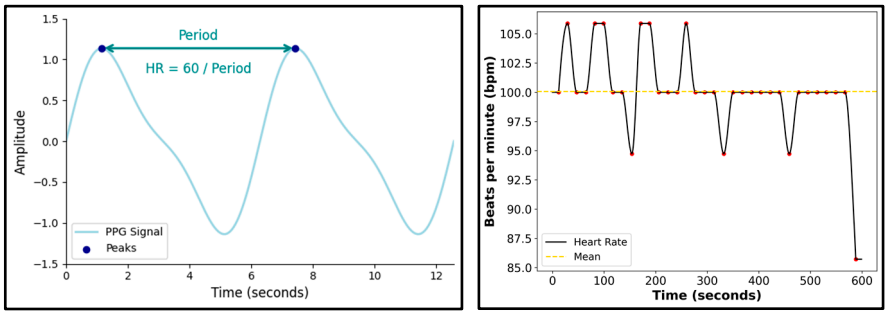


Figure 6. Heart Rate (HR) and its mean value extraction from PPG signal: The time interval (period) between consecutive systolic peaks (**left**), The time interval of bpm to measure the Heart Rate (HR) in beats per minute (bpm) (**right**).

4.1.3. Feature Extraction

HRV parameters, which are strongly associated with cardiovascular and autonomic nervous system responses, exhibit significant correlations with BGLs [11,13–21]. Leveraging this relationship, HRV metrics derived from PPG signals were extracted to predict BGL non-invasively using ML algorithms.

HRV features were categorized into time-domain, frequency-domain, and non-linear domains [43]. Time-domain features, such as the root mean square of successive differences (RMSSD) of peak-to-peak intervals (PPIs), capture temporal variations. Frequency-domain features, derived using methods like the Poincaré plot and Detrended Fluctuation Analysis (DFA), describe the distribution and correlations of HRV metrics, while non-linear methods

characterize dynamic patterns. Table 1 provides an overview of HRV metrics across these domains.

Table 1. An overview of HRV metrics categorized by their analysis domains [44].

Analysis Domain	Sub-Domain	Acronym	Description
Time-Domain	Deviation-based approach	SDNN	Standard deviation of NN intervals
		SDANN	Standard deviation of average NN intervals for each 5 min segment
		SDNN Index	Mean of the standard deviations of NN intervals in 5 min segments
	Difference-based approach	SDSD	Standard deviation of successive NN interval differences
		RMSSD	Root mean square of successive NN interval differences
		pNN20	Proportion of successive NN interval differences larger than 20 ms
		pNN50	Proportion of successive NN interval differences larger than 50 ms
	Geometric approach	HTI	Integral of the density of the NN interval histogram divided by its height
		TTIN	Baseline width of the RR interval histogram
Frequency-domain	Absolute Power	ULF	Power spectrum in the frequency range of $\leq$0.003 Hz
		VLF	Power spectrum in the frequency range of 0.0033–0.04 Hz
		LF	Power spectrum in the frequency range of 0.04–0.15 Hz
		HF	Power spectrum in the frequency range of 0.15–0.4 Hz
	Normalized/Relative Power	LnHF	Natural logarithm of HF
		HFn	Normalized HF
		LFn	Normalized LF
		LF/HF	Ratio of LF to HF power
Time-frequency domain	Linear approach	STFT	Power spectrum estimation using short-term Fourier transform
		WT	Power spectrum estimation using Wavelet transform
	Quadratic approach	WVD	Power spectrum estimation using Wigner—Ville distribution
		SWVD	Power spectrum estimation using Smoothed Wigner–Ville distribution
Non-linear domain	Poincaré Plot	SD1	Poincaré plot standard deviation perpendicular the line of identity
		SD2	Poincaré plot standard deviation along the line of identity
		SD1/SD2	Ratio of SD1 to SD2
	entropy Entropy	ApEn	Estimation of complexity using approximate entropy
		SampEn	Estimation of complexity using sample entropy
		MSE	Estimation of complexity using multiscale entropy
	Fractal Dimensions	DFA	Estimation of signal fluctuations using detrended fluctuation analysis
		CD	Estimation of minimum number of variables to define a dynamic model

The NeuroKit2 library [40] was employed to automate and streamline feature extraction. After computation, the features were consolidated into a database, with BGL as the target variable. Missing values were addressed through mean imputation to ensure dataset consistency [45], and normalization was applied using min-max scaling, transforming values to the range [0, 1] as follows [46,47]:

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{17}$$

where x′ is the normalized value, x is the original value, and x_{min} and x_{max} are the minimum and maximum values, respectively. This preprocessing step made the features scale compatible and reduced the variability because of different scales of the features. Such steps of preparation made the dataset ready for robust analysis and thus allowed predicting the BGL effectively.

4.2. Feature Selection (FS)

FS is a crucial process in ML that optimizes predictive performance by identifying the most relevant features while reducing computational complexity. By eliminating redundant data, FS enhances model accuracy and efficiency. Figure 7 illustrates its key components, including the original dataset, feature subset selection, subset evaluation, selection criteria, and validation [24].

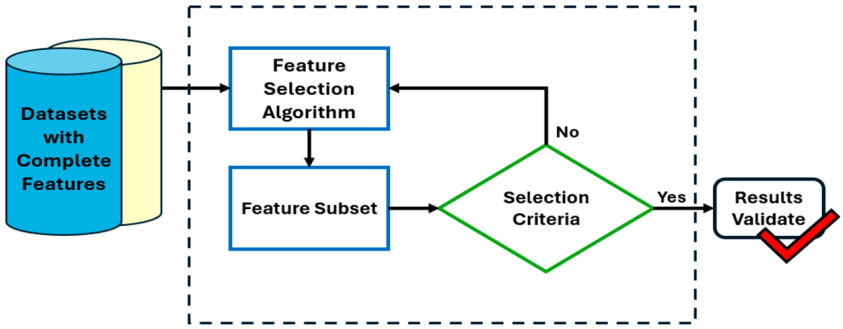


Figure 7. Feature selection process.

In this study, FS was framed as a binary combinatorial optimization problem, where features are represented by binary variables: a value of 1 indicates inclusion in the subset, and 0 indicates exclusion. The FS problem was modeled using a binary vector representation. Each vector's length corresponds to the total number of features, and its binary values determine the inclusion of features in the subset.

The FS process aimed to optimize the subset of features by balancing model accuracy and computational complexity. The performance of each subset was evaluated using a custom objective function designed to maximize the percentage of predictions within Zone A of the CEG, which represents clinically acceptable accuracy for BGLs. The objective function is expressed mathematically as:

$$\text{Objective Function} = \text{Maximize}\left(\frac{N_A}{N_{total}} \times 100\right), \tag{18}$$

where N_A is the number of predictions in Zone A, and N_{total} is the total number of predictions.

Given the NP-hard nature of FS [32,33] and the impracticality of exhaustive search, as previously mentioned, four metaheuristic algorithms, including IDA, BHHO, bGWO, and GA, were employed. These algorithms were selected for their capability to efficiently explore large search spaces while balancing exploration and exploitation [26,27,30,38,39].

Each metaheuristic algorithm employed a wrapper approach using three ML models, including RF, ETR, and LightGBM, to evaluate feature subsets iteratively and identify the most relevant feature subsets.

The open-source Zoofs library [48] was used to implement these algorithms. Each algorithm iteratively refined feature subsets, optimizing them for the regression task. The ML models were not only used as wrappers but also for comparative evaluation of the selected subsets. The best-performing subsets were identified based on their predictive performance. The parameter configurations for all metaheuristic algorithms were chosen based on thorough experimentation to ensure optimal performance. Details of these configurations and their comparative analysis are discussed in Section 5, "Results and Discussion".

4.3. Regression Models

Regression models were applied to predict BGLs using the selected feature subsets. For this study, the three previously mentioned regression models, including RF, ETR, and LightGBM, were chosen for their proven effectiveness in comparable prediction tasks. Moreover, their robust nature and ability to identify complex patterns also contributed to their selection [49–51]. Below, a description of the regression models is presented.

4.3.1. Random Forest (RF)

Breiman introduced the RF Regressor in 2001, which is an ensemble learning approach that integrates several decision trees to increase prediction accuracy and stability. Each tree is trained using a bootstrap sample of data, with random feature subsets chosen at each split, and the final predictions are averaged across all trees [49]. This technique decreases variance, captures complicated patterns, and minimizes overfitting, making RF particularly useful for noisy and missing data. However, its ensemble nature might result in increased computing complexity and reduced interpretability when compared to individual decision trees [52,53].

4.3.2. Extra Trees Regressor (ETR)

The ETR, developed by Geurts et al. in 2006, is similar to RF but includes more randomization to improve computing efficiency and decrease overfitting. Unlike RF, ETR does not use bootstrapping and selects both features and split points randomly for each tree, creating diverse and simplified ensembles. While this increases speed and reduces complexity, it may result in slightly lower accuracy for capturing complex patterns compared to methods like LightGBM [50].

4.3.3. Light Gradient Boosting Machine (LightGBM)

The LightGBM, created by Microsoft, is an advanced gradient boosting algorithm designed for scalability and efficiency. Leveraging Gradient-based One-Side Sampling (GOSS) to prioritize high-gradient samples and Exclusive Feature Bundling (EFB) to compress features, LightGBM achieves fast training and low memory usage. It is particularly suited for large datasets and high-dimensional problems but requires careful hyperparameter tuning. Although it offers excellent predictive accuracy, its complexity can pose challenges for interpretability [51,54].

4.4. Algoirthm Performance

The performance of the regression models was evaluated using three well-established metrics: Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and Clarke Error Grid (CEG) analysis. These metrics were chosen for their ability to provide quantitative and clinically meaningful insights into the accuracy and reliability of BGL predictions.

Their widespread use in ML studies ensures comparability with prior research and clinical applications [4–12].

MAE, as shown in Equation (19), quantifies the average absolute difference between predicted and actual values, offering an interpretable measure of overall prediction accuracy [10]. RMSE, represented in Equation (20), measures the standard deviation of prediction errors, providing insights into the magnitude of both small and large deviations, with greater sensitivity to outliers [10]. The CEG provides a clinically relevant evaluation by categorizing predictions into five zones (A to E) based on their potential impact on treatment decisions. Predictions in Zones A and B are considered accurate or acceptable, while those in Zones C, D, and E represent errors of increasing severity, with Zones D and E posing significant risks due to potential misclassification of hyperglycemia and hypoglycemia [55,56].

$$\text{MAE} = \sqrt{\frac{\sum_1^n |yi - \hat{y}i|}{n}} \tag{19}$$

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n} \left(\hat{y}i - yi\right)^2} \tag{20}$$

where n is the number of samples, $\hat{y}i$ represents the predicted value, and yi is the actual value.

The dataset was split into training and testing subsets, with 80% of the samples used to train the regression models and the remaining 20% for testing. This splitting also made it possible to test the model on unseen data and therefore gave a fair estimation of the accuracy of the model in making its predictions. The results of the regression models and feature selection algorithms are detailed in Section 5, "Results and Discussion".

5. Experimental Results and Discussion

This section presents the experimental results obtained using the proposed non-invasive BGL estimation methodology. The analysis includes the outcomes of the preprocessing, feature extraction, and feature selection phases, followed by regression model performance. The results are compared with existing methodologies to show how the proposed method is better than the existing one. Every experiment was performed in Google Colab [57] using the Python programming language [58]; the platform availed a 12.7 GB memory and Intel Xeon 4th Gen (Sapphire Rapids) processor.

5.1. Results of Data Preprocessing Stage

The preprocessing stage was vital to enhance the signal quality of the PPG data. The dataset used, as described in Section 4.1.1, underwent preprocessing to remove noise and artifacts, as detailed in Section 4.1.2. The application of a third-order Butterworth filter with a bandpass of 0.5–8 Hz effectively eliminated baseline wander and high-frequency noise while retaining the physiological components essential for identifying systolic peaks.

Figure 8 compares raw and filtered PPG signals, demonstrating significant noise reduction. This filtering step was critical for accurately detecting systolic peaks, as shown in Figure 9, which illustrates the detected peaks in a filtered PPG signal. After preprocessing, the dataset size was refined to 190 trials by removing samples with excessive noise or unclear peaks. These clean signals formed the foundation for robust feature extraction.

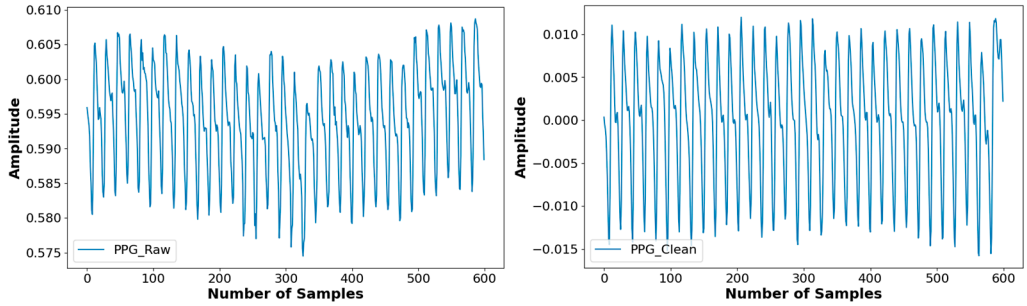


Figure 8. Comparison of raw (**left**) and filtered (**right**) PPG signals.

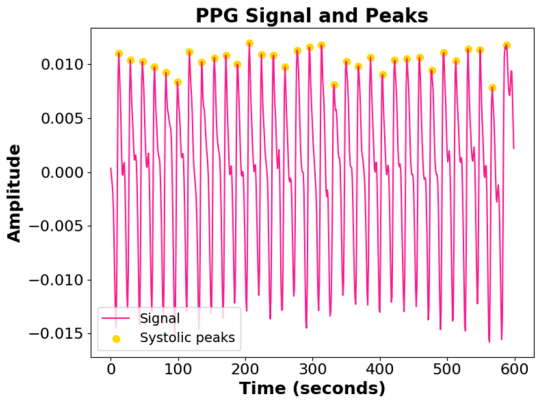


Figure 9. Detection of systolic peaks in a filtered PPG signal.

5.2. Results of Feature Extraction

Following signal preprocessing, the next phase involved extracting meaningful features from the PPG signals. A total of 69 features were extracted, including 19 time-domain features, 5 frequency-domain features, and 43 non-linear HRV features, as well as 2 features directly derived from the waveform, including PPG Rate Mean (HR) and PPG Rate Standard Deviation, as described in Section 4.1.3. Table 2 presents a summary of the extracted features, and Supplementary Table S1 provides detailed lists of the HRV features computed [59]. The final feature vector can be derived as Equation (21) for each frame f of the PPG signals. These features were then reduced by metaheuristic-based feature selection algorithms in a later stage.

$$S = \begin{bmatrix} \text{PPG_Rate_Mean} & \text{PPG_Rate_SD} & \text{RMSSD} & \text{MeanNN} & \text{SDNN} & \text{SDSD} & \text{CVNN} \\ \text{MadNN} & \text{MCVNN} & \text{IQRNN} & \text{pNN50} & \text{pNN20} & \text{HTI} & \text{TINN} \\ \text{MinNN} & \text{MaxNN} & \text{SDRMSSD} & HF & VHF & TP & HFn \\ SD2 & SD1/SD2 & S & CSI & CVI & \text{CSI_Modified} & PIP \\ PAS & GI & SI & AI & PI & C1d & C1a \\ C2d & C2a & SD2d & SD2a & Cd & Ca & SDNNd \\ ApEn & ShanEn & FuzzEn & CD & HFD & KFD & LZC \\ (Ma1)_Delta & (Ma1)_Max & (Ma1)_Asym. & (Ma1)_Fluct. & (Ma1)_Incr. & \text{CVSD} & \text{MedianNN} \\ \text{Prc20NN} & \text{Prc80NN} & LnHF & SD1 & SD1 & PSS & SD1d \\ SD1a & SDNNa & DFA_alpha1 & MFDFA_alpha1(Ma1)_Peak & (Ma1)_Mean & & \end{bmatrix}. \quad (21)$$

Table 2. Summary of extracted features.

Feature Type	Number of Features	Example Features
Time-Domain	19	RMSSD, PNN50
Frequency-Domain	5	LF/HF Ratio, Power Spectral Density
Non-linear Methods	43	DFA Alpha, Poincare SD1/SD2
PPG Derived Features	2	PPG Rate Mean, PPG Rate Standard Deviation

5.3. Metaheuristic Algorithms and Machine Learning Models (Parameter Settings)

Each metaheuristic algorithm operated with uniform configurations of 50 iterations and a population size of 50 to ensure consistency across all experiments. The parameter configurations for each algorithm were derived from prior literature to balance exploration and exploitation effectively [26,27,30,48]. Table 3 summarizes these configurations. IDA utilized a sinusoidal function for dynamic exploration-exploitation balancing, as recommended in the IDA paper [26], while BHHO applied a beta value of 0.5 for Levy flight-based exploration, as suggested by Zoofs hyperparameter settings [48]. bGWO employed a sigmoidal function with stochastic thresholding for binary position updates [27], and GA was configured with a selective pressure of two, an elitism of two, and a mutation rate of 0.05 to optimize performance [48].

Table 3. Parameter settings of metaheuristic algorithms.

Metaheuristic Algorithm	Selected Parameters
IDA	n_iteration = 50, population_size = 50, method = 'sinusoidal', beta = $\pi/2$
BHHO	n_iteration = 50, population_size = 50, beta = 0.5
bGWO	n_iteration = 50, population_size = 50, method = 2 (sigmoidal function)
GA	n_iteration = 50, population_size = 50, selective_pressure = 2, elitism = 2, mutation_rate = 0.05

For the regression models, hyperparameter tuning was performed iteratively to minimize MAE, ensuring optimal model performance. These ML models served as wrappers during feature selection and for comparative evaluation of the metaheuristic algorithms. Table 4 presents the detailed parameter settings for the ML models.

Table 4. Parameter settings of regression models.

Regression Model	Selected Parameters
RF	random_state = 162, n_estimators = 60, max_depth = 7, min_samples_split = 3
ETR	random_state = 7, ccp_alpha = 1.2, n_estimators = 70, criterion = 'squared_error', max_depth = 19, min_impurity_decrease = 0.1
LGBM	num_leaves = 31, verbosity = −1, learning_rate = 0.01, n_estimators = 500, max_depth = 5, min_child_samples = 10, reg_alpha = 0.25, reg_lambda = 0.1

This systematic experimental setup ensured a comprehensive evaluation of the feature selection and regression models, setting the foundation for the comparative analysis in subsequent sections.

5.4. Comparative Performance Analysis of Proposed Models with Metaheuristic Algorithms and Regression Models

This section evaluates the proposed methodology for predicting BGLs through a comprehensive analysis of the combinations of metaheuristic algorithms and regression models. The evaluation was conducted in two phases to systematically assess and refine the models. Phase One focused on evaluating all metaheuristic-regression combinations through 30 independent runs to ensure robust comparisons, while Phase Two refined the best-performing combinations using 10 independent runs to optimize accuracy, feature sets, and computational efficiency further.

In terms of selected features, MAE, RMSE, CEG Zone A percentage, and time costs, the results of the models were comprehensively analyzed with mean, standard deviation, and the best values across multiple runs to capture their clinical relevance. Although computational time was documented as a measure of efficiency, it is not a primary criterion in this study.

5.4.1. Phase One: Initial Evaluation of Model-MH Combinations

The results of the 12 combinations of metaheuristic algorithms and regression models are summarized in Table 5. Among the algorithms tested, the IDA consistently demonstrated superior performance across all ML models. IDA stood out for its ability to balance feature selection with predictive accuracy, making it the most reliable and versatile choice for non-invasive BGL estimation.

Table 5. Performance of the models during phase 1 in terms of selected features, fitness, RMSE, MAE, and time costs, evaluated on the testing split of the dataset. The data showcasing the best performance for each metric are highlighted in bold.

Model	Matric	Selected Features	Fitness (Zone A%)	RMSE (mg/dL)	MAE (mg/dL)	Time Costs (Sec)
LightGBM	-	69	68.42	18.10	16.16	1
IDA-LightGBM	Mean	35.2	**86.14**	17.68	**14.81**	468.10
	STD	**4.28**	1.91	1.22	0.87	27.94
	Best	33	**89.47**	17.03	14.64	494
BHHO-LightGBM	Mean	46.40	75.53	18.31	16.38	690.65
	STD	5.22	10.90	3.66	1.44	105.44
	Best	50	86.84	**16.37**	**13.42**	745.35
bGWO-LightGBM	Mean	46.23	84.47	**17.63**	15.21	521.51
	STD	5.41	**1.42**	**0.65**	**0.66**	65.73
	Best	48	86.84	16.92	14.36	543.51
GA-LightGBM	Mean	**34.6**	84.65	17.65	15.00	**97.70**
	STD	4.88	2.26	1.01	0.85	22.71
	Best	**32**	89.47	17.36	14.71	**64.2**
ETR	-	69	76.32	16.52	14.61	1
IDA-ETR	Mean	**34.17**	**84.74**	**17.11**	**15.04**	392.59
	STD	4.84	1.25	0.61	0.68	83.53
	Best	**27**	86.84	**15.7**	**12.61**	428.09
BHHO- ETR	Mean	45.5	76.66	17.95	15.96	550.28
	STD	4.31	7.28	0.92	0.83	132.40
	Best	38	86.84	17.18	15.08	387.08
bGWO-ETR	Mean	46.1	84.12	17.17	15.27	482.62
	STD	**3.37**	**1.07**	**0.58**	**0.50**	**13.26**
	Best	47	86.84	15.71	14.19	480.67
GA-ETR	Mean	34.7	84.39	17.16	15.15	99.82
	STD	4.20	1.91	0.72	0.62	22.17
	Best	37	86.84	16.18	14.22	88.49
RF	-	69	68.42	18.12	16.02	1

Table 5. *Cont.*

Model	Matric	Selected Features	Fitness (Zone A%)	RMSE (mg/dL)	MAE (mg/dL)	Time Costs (Sec)
IDA-RF	Mean	**33.13**	**81.84**	18.37	16.06	611.93
	STD	4.59	0.3	0.65	0.65	37.58
	Best	**31**	84.21	18.13	**15.04**	623.64
BHHO-RF	Mean	45.3	72.9	19.08	16.8	969.58
	STD	4.98	8.86	1.21	1.05	52.42
	Best	35	84.21	17.45	15.70	995.61
bGWO-RF	Mean	45.83	81.32	18.33	16.16	719.44
	STD	4.37	1.57	0.50	0.43	16.99
	Best	46	84.21	**17.15**	15.43	726.55
GA-RF	Mean	33.43	81.58	**18.28**	**15.95**	131.84
	STD	3.34	2.04	0.64	0.46	23.59
	Best	38	84.21	17.64	15.35	121.39

The IDA-LightGBM combination emerged as the top-performing model, achieving the lowest mean MAE of 14.81 mg/dL with a standard deviation of 0.87 mg/dL. In its best run, it recorded an RMSE of 17.03 mg/dL, an MAE of 14.64 mg/dL, and the highest CEG Zone A percentage of 89.47%, affirming its strong clinical relevance. The model effectively reduced the feature set by 52.2%, selecting 33 features in its best run. Furthermore, it required a mean computational time of 468.10 s, showcasing a well-rounded performance in terms of accuracy, efficiency, and feature reduction.

The IDA-ETR combination closely followed, achieving a mean MAE of 15.04 mg/dL with a standard deviation of 0.68 mg/dL. In its best run, it recorded a MAE of 12.61 mg/dL and a CEG Zone A percentage of 86.84%. Across all runs, the model selected an average of 34.17 features. Its mean computational time was 392.59 s, offering competitive results while maintaining a balance between feature reduction and accuracy.

In contrast, models incorporating the Genetic Algorithm (GA) exhibited lower performance, with higher mean MAE and RMSE values. While GA-LightGBM demonstrated the shortest computational time of 97.70 s, its performance metrics were not competitive. These results underscore the superiority of IDA in achieving a good balance between feature reduction, predictive accuracy, and computational efficiency.

5.4.2. Phase 2: Refining Top-Performing Models

Phase Two focused on refining the top-performing combinations from Phase One. The IDA-LightGBM and IDA-ETR combinations were selected for further evaluation based on their clinical relevance and computational efficiency. The results of this phase are summarized in Table 6. Both combinations demonstrated strong predictive accuracy and feature reduction, but IDA-LightGBM emerged as the most clinically significant and computationally efficient model.

In its best run, IDA-LightGBM selected only 18 features while achieving a MAE of 13.96 mg/dL and a RMSE of 15.76 mg/dL. It also maintained a high CEG Zone A percentage of 94.74%, ensuring that all predictions were clinically acceptable. The hyperparameters of the LightGBM model were fine-tuned to further improve performance, with *n_estimators* set to 670 and *reg_alpha* set to 3.1. This refinement lowered the MAE to 13.17 mg/dL and the RMSE to 15.36 mg/dL. The mean computational time was recorded at 311.72 s, underscoring the model's robustness and efficiency.

Table 6. Comparative analysis of best-performing models during phase 2 in terms of selected features, fitness, RMSE, MAE, and time costs, evaluated on the testing split of the dataset. The data showcasing the best performance for each metric are highlighted in bold.

Model	Matric	Selected Features	Fitness (Zone A%)	RMSE (mg/dL)	MAE (mg/dL)	Time Costs (Sec)
IDA-ETR	Mean	**14.9**	91.05	**15.375**	13.345	321.32
	STD	**2.07**	**1.29**	0.81	0.85	**2.96**
	Best	**13**	92.11	**14.39**	**12.43**	319.15
IDA-LightGBM	Mean	20.2	**92.89**	15.48	**13.30**	**311.72**
	STD	3.34	1.68	**0.47**	**0.42**	5.88
	Best	18	**94.74**	15.76	13.96	**298**
	Best (Modified)			15.36	13.17	

IDA-ETR also performed well, selecting 13 features in its best run with a MAE of 12.43 mg/dL and a RMSE of 14.39 mg/dL. The model achieved a best-case CEG Zone A percentage of 92.11% and required a mean computational time of 321.32 s.

Although IDA-ETR demonstrated superior feature reduction and accuracy, IDA-LightGBM emerged as the best model due to its greater clinical significance, based on the higher percentage of Zone A and balanced performance metrics. Despite the iterative optimization of the IDA and the gradient boosting mechanism of LightGBM, the overall computational complexity was effectively managed, as the model achieved efficient runtime and significant reduction in the feature set. This balance of computational efficiency, predictive accuracy, and clinical relevance underscores the practicality of IDA-LightGBM for non-invasive blood glucose monitoring.

5.4.3. Convergence Evaluation and Clarke Error Grid Analysis (CEG)

The convergence curves of the top models, presented in Figure 10, illustrate the rapid optimization capability of IDA-LightGBM, which reached convergence within the first half of the iterations. This efficiency underscores its ability to balance exploration and exploitation effectively.

The CEG analysis was used to verify the clinical safety of the two best models from phase two, IDA-LightGBM and IDA-ETR. As shown in Figure 11, the IDA-LightGBM model exhibited superior clinical safety with 94.74% of predictions in Zone A and no predictions in Zones C, D, or E. IDA-ETR followed with 92.11% of predictions in Zone A and similarly no predictions in dangerous zones. These results confirm the outcomes from Phase Two, confirming IDA-LightGBM as the best-performing model.

5.4.4. Accuracy of Predicted BGL Values for Best Model

The best IDA-LightGBM model exhibited superior performance by selecting 18 features from an initial set of 69 HRV features derived from PPG signals, as denoted by the feature vector D in Equation (22). This reduction of 73.91% in the feature set significantly decreases computational demands while enhancing predictive accuracy. In contrast to the ETR model, which utilized all 69 features and achieved an RMSE of 16.52 mg/dL, an MAE of 14.61 mg/dL, and a CEG Zone A percentage of 76.32%, the IDA-LightGBM model demonstrated substantial improvements across all key metrics.

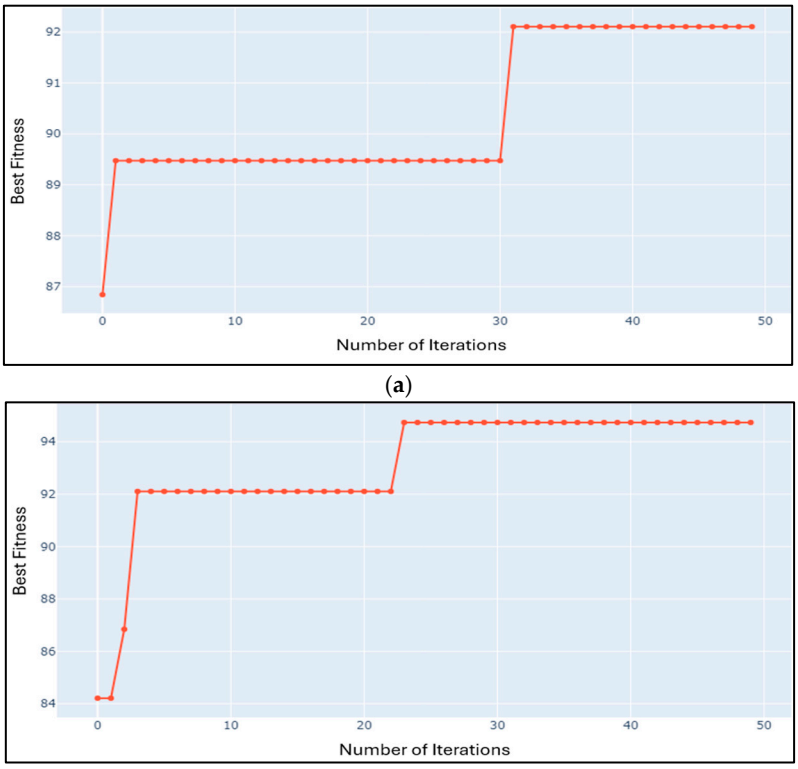
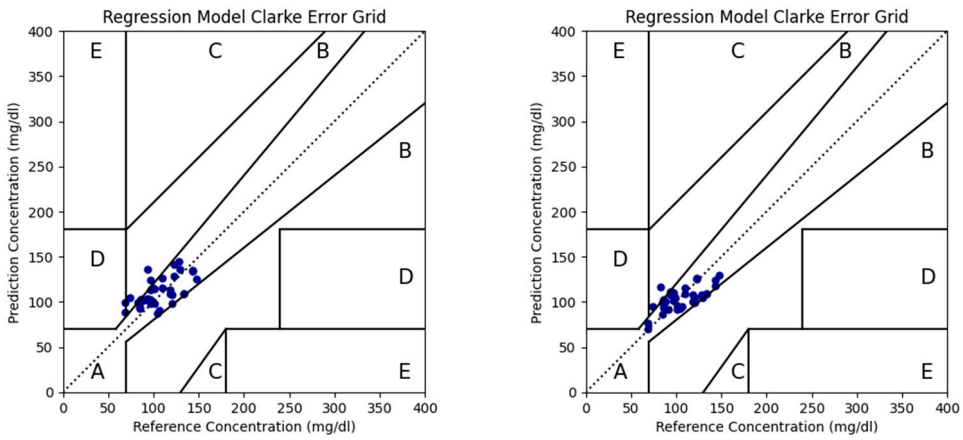


Figure 10. Convergence curves of the best performing models of phase 2: (**a**) best IDA-ETR; (**b**) best modified IDA-LightGBM (best overall model).



A	B	C	D	E
92.11%	7.89%	0%	0%	0%

(**a**)

A	B	C	D	E
94.74%	5.26%	0%	0%	0%

(**b**)

Figure 11. CEG analysis of the best performing models of phase 2: (**a**) best IDA-ETR; (**b**) best modified IDA-LightGBM (best overall model).

Specifically, the IDA-LightGBM model achieved an RMSE of 15.36 mg/dL, an MAE of 13.17 mg/dL, and a significantly higher Zone A percentage of 94.74%, indicating a 24.13% enhancement in clinically relevant predictions over the full feature set. Furthermore, the IDA-LightGBM model ensured that no predictions fell into dangerous CEG Zones, further confirming its clinical reliability.

Hyperparameter tuning of the LightGBM model also contributed to these enhancements. The increase in boosting iterations (n_estimators) to 670 improved the model's ability to capture complex patterns within the reduced feature set. Moreover, the L1 regu-

larization parameter (reg_alpha) was modified to 3.1, reducing overfitting and promoting sparsity. This optimized configuration achieved an effective balance between computational efficiency and predictive accuracy. These findings are illustrated in Table 7, which compares actual and predicted glucose values, highlighting the model's potential for non-invasive diabetes management applications.

$$D = \begin{bmatrix} \text{ppg_Rate_SD} & SD1d & \text{SDRMSSD} & \text{MeanNN} & FuzzEn \\ \text{SDSD} & \text{CVN} & TP & CVI & \text{MinNN} \\ \text{MadNN} & \text{MCVNN} & PI & SI & \text{MaxNN} \\ \text{ShanEn} & \text{Prc20NN} & MFDFA_alpha1_Mean & & \end{bmatrix} \quad (22)$$

Table 7. Performance measure for the best BGL estimation model (IDA-LightGBM), evaluated on the testing split of the dataset.

Actual BGL (mg/dL)	Predicted BGL (mg/dL)	Difference	Difference Percentage
68.4	70.28	1.88	2.75
86.4	93.68	7.28	8.43
97.2	110.12	12.92	13.29
109.8	115.85	6.05	5.51
122.4	124.75	2.35	1.92

5.5. Comparative Performance Analysis of Proposed Models with Existing Studies Utilizing PPG Signals

This section compares the proposed IDA-LightGBM model for non-invasive BGL estimation against existing studies utilizing PPG signals, focusing on accuracy, clinical relevance, simplicity, and practicality. A summary is provided in Table 8.

Table 8. Comparison with current work on estimating blood glucose.

Reference	Algorithm	RMSE	MAE	Zone A & B%	Input Features	FS Method	Feature Set Size	Number of Subject	Detection Range
J. Li et al. [7]	DNN	26.85		99.49%	ECG and PPG	Three FS techniques		21	39.64–392.8
C.-Y. Liao and W.-C. Fang [8]	LRCN	11.146	4.7	99%	PPG (1 min)	Convolutional layers		15	75–225
Yen et al. [5]	BPNN	5.08	6.38	100%	PPG & bioimpedance (3 min) & physiological		30	40	100–150
Islam et al. [6]	PLS	17.71			PPG (60 s)			52	50–150
Alghlayini et al. [9]	CNN	26.1	17.26	100% **(92.85% Zone A)**	PPG (30 s)	Convolutional layers		52	50–150
Shuzan et al. [10]	BET	30.74	22.90	100% **(61.88% Zone A)**	PPG, BP, demographic	MRMR	44	139	70–350
Guzman et al. [11]	SVR	18.66	16.45		PPG, BMI, Gender	Filtering and DEA	20	5	
Chowdhury et al. [12]	NN	17.26	13.51	99.43% **(79.03% Zone A)**	PPG, EDA, ST, Food logs	Convolutional layers		15	60–160
This work	LightGBM	15.36	13.17	100% **(94.74% Zone A)**	PPG (30 s)	IDA	18	52	50–150

The IDA-LightGBM model achieved an RMSE of 15.36 mg/dL, MAE of 13.17 mg/dL, and 94.74% predictions in CEG Zone A, outperforming previous models in both predictive accuracy and clinical safety. Unlike other multisensor approaches that use electrodermal

activity, bioimpedance, or skin temperature, as demonstrated by Chowdhury et al. [12] and Yen et al. [5], the proposed model solely utilizes the PPG signal and chooses 18 features to build the model, thus simplifying the system.

In contrast to Li et al. [7], whose deep learning approach had predictions in dangerous CEG zones, our model ensured all predictions were within safe zones. Compared to Islam et al.'s [6] smartphone-based PPG approach, which exhibited lower predictive accuracy, our model provides better reliability with a broader detection range of 50–150 mg/dL. Studies like Guzman et al. [11], which used additional physiological inputs such as BMI and blood pressure, achieved lower accuracy and higher system complexity, underscoring the streamlined effectiveness of our approach.

Unlike models requiring larger feature sets or additional user input, our approach combines metaheuristic algorithms and machine learning to balance accuracy, feature reduction, and simplicity. Furthermore, while many prior studies were limited by small sample sizes, our model was validated on a larger and more diverse dataset, enhancing its generalizability and applicability.

By integrating metaheuristic algorithms with ML models, this study advances the field of non-invasive glucose monitoring. The streamlined IDA-LightGBM model balances accuracy, feature reduction, and simplicity, providing a practical solution for diabetes management. To the best of our knowledge, this is the first study to leverage metaheuristic optimization for BGL estimation using PPG signals, marking a significant step forward in non-invasive diabetes care.

6. Conclusions and Future Work

This study introduced a novel approach for non-invasive blood glucose level (BGL) estimation utilizing HRV features derived from PPG signals. By integrating advanced preprocessing techniques, metaheuristic-based feature selection, and machine learning (ML) models, the proposed approach demonstrated high accuracy and clinical significance. The IDA-LightGBM model was identified as the most effective combination, achieving a mean absolute error (MAE) of 13.17 mg/dL and a root mean squared error (RMSE) of 15.36 mg/dL while selecting only 18 features. It achieved 94.74% of predictions within Clarke Error Grid (CEG) Zone A, with the remaining predictions falling into Zone B, and none in the dangerous Zones C, D, or E. Furthermore, in comparison to other monitoring systems that may require the use of multi-sensing or a greater number of features, this approach offers a practical and broader detection range of 50–150 mg/dL utilizing solely PPG signals.

Despite its promising findings, the study's results are limited by the relatively small sample size of 52 subjects, as well as the examined glucose range that excludes conditions of severe hyperglycemia or hypoglycemia. Expanding this work to include more diverse populations and more comprehensive glucose levels would enhance the validity and reliability of the model.

Future work will involve testing the model in realistic settings, for example, an application that tracks glucose levels through wearable technology, and exploring other algorithms to improve the performance of the system. These advancements can enhance the validity and practicality of non-invasive glucose monitoring and therefore enhance diabetes management.

Supplementary Materials: The following supporting information can be downloaded at https://www.mdpi.com/article/10.3390/a18020095/s1, Table S1: Heart rate variability (HRV) metrics computed in this study [59].

Author Contributions: Conceptualization, S.A. and M.A.A.-B.; methodology, S.A.; software, S.A.; validation, S.A. and M.A.; formal analysis, M.A.A.-B.; investigation, M.A.A-B.; resources, M.A.; data curation, M.A.; writing—original draft preparation, S.A.; writing—review and editing, S.A. and M.A.A.-B.; visualization, M.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Ajman University grant number 2024-IRG-ENIT-16, and the APC was funded by Ajman University.

Data Availability Statement: The original contributions presented in this study are included in the article/Supplementary Material. Further inquiries can be directed to the corresponding author(s).

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

References

1. World Health Organization: WHO. World Health Organization: WHO Diabetes. Available online: https://www.who.int/news-room/fact-sheets/detail/diabetes (accessed on 9 November 2024).
2. International Diabetes Federation What Is Diabetes | International Federation of Diabetes. Available online: https://www.idf.org/aboutdiabetes/what-is-diabetes.html (accessed on 9 November 2024).
3. Diabetes Tests | ADA. Available online: https://www.diabetes.org/healthy-living/medication-treatments/blood-glucose-testing-and-control (accessed on 9 November 2024).
4. Sen Gupta, S.; Kwon, T.-H.; Hossain, S.; Kim, K.-D. Towards non-invasive blood glucose measurement using machine learning: An all-purpose PPG System design. *Biomed. Signal Process. Control* **2021**, *68*, 102706. [CrossRef]
5. Yen, C.-T.; Chen, U.-H.; Wang, G.-C.; Chen, Z.-X. Non-invasive blood glucose estimation system based on a neural network with dual-wavelength photoplethysmography and bioelectrical impedance measuring. *Sensors* **2022**, *22*, 4452. [CrossRef] [PubMed]
6. Islam, T.T.; Ahmed, M.S.; Hassanuzzaman, M.; Bin Amir, S.A.; Rahman, T. Blood glucose level regression for Smartphone PPG Signals Using Machine Learning. *Appl. Sci.* **2021**, *11*, 618. [CrossRef]
7. Li, J.; Ma, J.; Omisore, O.M.; Liu, Y.; Tang, H.; Ao, P.; Yan, Y.; Wang, L.; Nie, Z. Noninvasive blood glucose monitoring using spatiotemporal ECG and PPG Feature Fusion and weight-based Choquet Integral Multimodel approach. *IEEE Trans. Neural Netw. Learn. Syst.* **2024**, *35*, 14491–14505. [CrossRef] [PubMed]
8. Liao, C.-Y.; Fang, W.-C. LRCN-based noninvasive blood glucose level estimation. In Proceedings of the 2023 IEEE International Symposium on Circuits and Systems (ISCAS), Monterey, CA, USA, 21–25 May 2023; pp. 1–5.
9. Alghlayini, S.; Hosni, A.; Atef, M. Photoplethysmography based blood glucose estimation using convolutional neural networks. In Proceedings of the 2023 Advances in Science and Engineering Technology International Conferences (ASET), Dubai, United Arab Emirates, 20–23 February 2023; pp. 1–5.
10. Shuzan, M.N.; Chowdhury, M.H.; Chowdhury, M.E.; Abualsaud, K.; Yaacoub, E.; Faisal, M.A.; Alshahwani, M.; Al Bordeni, N.; Al-Kaabi, F.; Al-Mohannadi, S.; et al. QU-GM: An IoT Based Glucose Monitoring System from Photoplethysmography, Blood Pressure and Demographic Data using Machine Learning. *IEEE Access* **2024**, *12*, 77774–77790. [CrossRef]
11. Guzman, L.; Cazares, A.M.; Martinez-Torteya, A. Model for glycemic level detection using heart rate variability in a Mexican sample. In Proceedings of the 2020 IEEE-EMBS Conference on Biomedical Engineering and Sciences (IECBES), Langkawi Island, Malaysia, 1–3 March 2021; pp. 505–510.
12. Chowdhury, M.H.; Chowdhury, M.E.H.; Alqahtani, A. MMG-Net: Multi modal approach to estimate blood glucose using multi-stream and cross modality attention. *Biomed. Signal Process. Control* **2024**, *92*, 105975. [CrossRef]
13. Martinez-Delgado, G.H.; Martinez-Torteya, A. Predicting high levels of blood glucose through heart rate variability and machine learning. In Proceedings of the 2022 E-Health and Bioengineering Conference (EHB), Iasi, Romania, 17–18 November 2022; pp. 01–04.
14. Tarvainen, M.P.; Laitinen, T.P.; Lipponen, J.A.; Cornforth, D.J.; Jelinek, H.F. Cardiac autonomic dysfunction in type 2 diabetes—Effect of hyperglycemia and disease duration. *Front. Endocrinol.* **2014**, *5*, 130. [CrossRef] [PubMed]
15. Singh, J.P.; Larson, M.G.; O'Donnell, C.J.; Wilson, P.F.; Tsuji, H.; Lloyd-Jones, D.M.; Levy, D. Association of hyperglycemia with reduced heart rate variability (the Framingham Heart Study). *Am. J. Cardiol.* **2000**, *86*, 309–312. [CrossRef] [PubMed]
16. Chen, Y.; Jia, T.; Yan, X.; Dai, L. Blood glucose fluctuations in patients with coronary heart disease and diabetes mellitus correlates with heart rate variability: A retrospective analysis of 210 cases. *Niger. J. Clin. Pract.* **2020**, *23*, 1194. [CrossRef] [PubMed]
17. Vishinov, I.; Gusev, M.; Poposka, L.; Vavlukis, M.; Ahmeti, I. Distribution analysis of long-term heart rate variability versus blood glucose. In Proceedings of the 2021 44th International Convention on Information, Communication and Electronic Technology (MIPRO), Opatija, Croatia, 27 September–1 October 2021; pp. 328–333.

18. Nickel, A.; Buresh, R.; McLester, C.; Canino, A.; Wilner, G.; Vaughan, K.; Chung, P.; Kliszczewicz, B. The relationship between heart rate variability and glucose clearance in healthy men and women. *PLoS ONE* **2024**, *19*, e0303346. [CrossRef]
19. Benichou, T.; Pereira, B.; Mermillod, M.; Daniela, P.; Tauveron, I.; Maqdasy, S.; Dutheil, F. Heart rate variability in type 2 diabetes mellitus: A systematic review and meta-analysis. *Ann. D'endocrinologie* **2018**, *79*, 465–466. [CrossRef]
20. Ferrario, M.; Raimann, J.G.; Thijssen, S.; Signorini, M.G.; Kruse, A.; Diaz-Buxo, J.A.; Cerutti, S.; Levin, N.W.; Kotanko, P. Effects of dialysate glucose concentration on heart rate variability in chronic hemodialysis patients: Results of a prospective randomized trial. *Kidney Blood Press. Res.* **2011**, *34*, 334–343. [CrossRef] [PubMed]
21. Aggarwal, Y.; Das, J.; Mazumder, P.M.; Kumar, R.; Sinha, R.K. Heart rate variability time domain features in automated prediction of diabetes in rat. *Phys. Eng. Sci. Med.* **2020**, *44*, 45–52. [CrossRef] [PubMed]
22. Susana, E.; Ramli, K.; Purnamasari, P.D.; Apriantoro, N.H. Non-invasive classification of blood glucose level based on photoplethysmography using time–frequency analysis. *Information* **2023**, *14*, 145. [CrossRef]
23. Hossain, S.; Debnath, B.; Biswas, S.; Al-Hossain, M.J.; Anika, A.; Zaman Navid, S.K. Estimation of blood glucose from PPG signal using Convolutional Neural Network. In Proceedings of the 2019 IEEE International Conference on Biomedical Engineering, Computer and Information Technology for Health (BECITHCON), Dhaka, Bangladesh, 28–30 November 2019; pp. 53–58.
24. Nssibi, M.; Manita, G.; Korbaa, O. Advances in nature-inspired metaheuristic optimization for feature selection problem: A comprehensive survey. *Comput. Sci. Rev.* **2023**, *49*, 100559. [CrossRef]
25. Mafarja, M.M.; Eleyan, D.; Jaber, I.; Hammouri, A.; Mirjalili, S. Binary dragonfly algorithm for feature selection. In Proceedings of the 2017 International Conference on New Trends in Computing Sciences (ICTCS), Amman, Jordan, 11–13 October 2017; pp. 12–17.
26. Hammouri, A.I.; Mafarja, M.; Al-Betar, M.A.; Awadallah, M.A.; Abu-Doush, I. An improved Dragonfly Algorithm for feature selection. *Knowl. -Based Syst.* **2020**, *203*, 106131. [CrossRef]
27. Emary, E.; Zawbaa, H.M.; Hassanien, A.E. Binary grey wolf optimization approaches for feature selection. *Neurocomputing* **2016**, *172*, 371–381. [CrossRef]
28. Selvakumari Jeya, I.J.; Praveen Kumar, M.; Saravanan, B.; Niranjani, V.; Muneeswaran, V. Cloud computing based diabetes classification using binary grey wolf optimization. In Proceedings of the 2023 First International Conference on Advances in Electrical, Electronics and Computational Intelligence (ICAEECI), Tiruchengode, India, 19–20 October 2023; pp. 1–7.
29. Noor Muhammed, N.; Omar Saber, Q. Improving cancer diseases classification using a hybrid filter and wrapper feature subset selection. *Ann. Proteom. Bioinform.* **2020**, *4*, 006–011. [CrossRef]
30. Thaher, T.; Heidari, A.A.; Mafarja, M.; Dong, J.S.; Mirjalili, S. Binary Harris Hawks optimizer for high-dimensional, low sample size feature selection. In *Evolutionary Machine Learning Techniques: Algorithms and Applications*; Springer: Singapore, 2020; pp. 251–272.
31. Tariq Ibrahim, H.; Jalil Mazher, W.; Mahmood Jassim, E. Feature selection: Binary harris hawk optimizer based biomedical datasets. *Intel. Artif.* **2022**, *25*, 33–49. [CrossRef]
32. Faris, H.; Al-Zoubi, A.M.; Heidari, A.A.; Aljarah, I.; Mafarja, M.; Hassonah, M.A.; Fujita, H. An intelligent system for spam detection and identification of the most relevant features based on evolutionary random weight networks. *Inf. Fusion* **2019**, *48*, 67–83. [CrossRef]
33. Chantar, H.; Mafarja, M.; Alsawalqah, H.; Heidari, A.A.; Aljarah, I.; Faris, H. Feature selection using binary grey wolf optimizer with elite-based crossover for Arabic Text Classification. *Neural Comput. Appl.* **2019**, *32*, 12201–12220. [CrossRef]
34. Guyon, I.; Elisseeff, A. An Introduction to Variable and Feature Selection. *J. Mach. Learn. Res.* **2003**, *3*, 1157–1182.
35. Zelinka, I. A survey on evolutionary algorithms dynamics and its complexity—Mutual relations, past, present and future. *Swarm Evol. Comput.* **2015**, *25*, 2–14. [CrossRef]
36. Mafarja, M.; Mirjalili, S. Whale optimization approaches for wrapper feature selection. *Appl. Soft Comput.* **2018**, *62*, 441–453. [CrossRef]
37. Wolpert, D.H.; Macready, W.G. No free lunch theorems for optimization. *IEEE Trans. Evol. Comput.* **1997**, *1*, 67–82. [CrossRef]
38. Varan, M.; Erduman, A.; Menevşeoğlu, F. A grey wolf optimization algorithm-based optimal reactive power dispatch with wind-integrated power systems. *Energies* **2023**, *16*, 5021. [CrossRef]
39. Mitchell, M. *An Introduction to Genetic Algorithms*; The MIT Press: Cambridge, MA, USA, 1998.
40. Makowski, D.; Pham, T.; Lau, Z.J.; Brammer, J.C.; Lespinasse, F.; Pham, H.; Schölzel, C.; Chen, S.H. NeuroKit2: A python toolbox for Neurophysiological Signal Processing. *Behav. Res. Methods* **2021**, *53*, 1689–1696. [CrossRef] [PubMed]
41. Elgendi, M.; Norton, I.; Brearley, M.; Abbott, D.; Schuurmans, D. Systolic peak detection in acceleration photoplethysmograms measured from emergency responders in tropical conditions. *PLoS ONE* **2013**, *8*, e76585. [CrossRef] [PubMed]
42. Oppenheim, A.V.; Schafer, R.W.; Buck, J.R. *Discrete-Time Signal Processing*, 2nd ed.; Prentice-hall Englewood Cliffs: New Jersey, NJ, USA, 1999.
43. Malik, M.; Bigger, J.T.; Camm, A.J.; Kleiger, R.E.; Malliani, A.; Moss, A.J.; Schwartz, P.J. Heart rate variability: Standards of measurement, physiological interpretation, and clinical use. *Eur. Heart J.* **1996**, *17*, 354–381. [CrossRef]

44. Pham, T.; Lau, Z.J.; Chen, S.H.; Makowski, D. Heart rate variability in psychology: A review of HRV Indices and an analysis tutorial. *Sensors* **2021**, *21*, 3998. [CrossRef]
45. Li, W.; Liu, Z. A method of SVM with normalization in intrusion detection. *Procedia Environ. Sci.* **2011**, *11*, 256–262. [CrossRef]
46. Sessa, J.; Syed, D. Techniques to deal with missing data. In Proceedings of the 2016 5th International Conference on Electronic Devices, Systems and Applications (ICEDSA), Ras Al Khaimah, United Arab Emirates, 6–8 December 2016; pp. 1–4.
47. Waqas, I.; Syed Saeed-Ur-Rahman, S.; Imran, M.J.; Rehan, A. Treatment of missing values in data mining. *J. Comput. Sci. Syst. Biol.* **2015**, *9*, 2.
48. jaswinder9051998 JASWINDER9051998/zoofs: Zoofs is a Python Library for Performing Feature Selection Using a Variety of Nature-Inspired Wrapper Algorithms. The Algorithms Range from Swarm-Intelligence to Physics-Based to Evolutionary. It's Easy to Use, Flexible and Powerful Tool to Reduce your Feature Size. Available online: https://github.com/jaswinder9051998/zoofs (accessed on 9 November 2024).
49. Breiman, L. Random Forests. *Mach. Learn.* **2001**, *45*, 5–32. [CrossRef]
50. Geurts, P.; Ernst, D.; Wehenkel, L. Extremely randomized trees. *Mach. Learn.* **2006**, *63*, 3–42. [CrossRef]
51. Ke, G.; Meng, Q.; Finley, T.; Wang, T.; Chen, W.; Ma, W.; Ye, Q.; Liu, T.-Y. LightGBM: A Highly Efficient Gradient Boosting Decision Tree. In *Advances in Neural Information Processing Systems 30*; Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R., Eds.; Curran Associates, Inc.: Nice, France, 2017; pp. 3146–3154.
52. Criminisi, A. Decision forests: A unified framework for classification, regression, density estimation, manifold learning and semi-supervised learning. *Found. Trends Comput. Graph. Vis.* **2011**, *7*, 81–227. [CrossRef]
53. Rokach, L. Decision forest: Twenty Years of research. *Inf. Fusion* **2016**, *27*, 111–125. [CrossRef]
54. Welcome to LIGHTGBM's Documentation! Available online: https://lightgbm.readthedocs.io/en/latest/ (accessed on 9 November 2024).
55. Clarke Error Grid. Available online: https://en.wikipedia.org/wiki/Clarke_Error_Grid (accessed on 9 November 2024).
56. Jernelv, I.L.; Milenko, K.; Fuglerud, S.S.; Hjelme, D.R.; Ellingsen, R.; Aksnes, A. A review of optical methods for continuous glucose monitoring. *Appl. Spectrosc. Rev.* **2018**, *54*, 543–572. [CrossRef]
57. Google Colab Colab.Google. Available online: https://colab.google/ (accessed on 9 November 2024).
58. Welcome to Python.Org. Available online: https://www.python.org/ (accessed on 9 November 2024).
59. Frasch, M.G. Comprehensive HRV estimation pipeline in python using neurokit2: Application to sleep physiology. *MethodsX* **2022**, *9*, 101782. [CrossRef]

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