Apash Roy · Dalwin Vinoth Kumar · Jeevanand Edavazhikkal Sarvothama Shenoi · Rajesh Ramachandran · Rupali Sunil Wagh · Saleema Janardhanan Subramanian *Editors*

ACSAR 9

Sustainable Innovations in Statistics and Data Science

First International Conference, SISD 2025, Bangalore, India, March 27–28, 2025, Revised Selected Papers

SISD'25

Springer

Advances in Computer Science Applications and Research 9

Series Editor

Juan E. Gilbert, *Computer & Information Science, University of Florida, Gainesville, FL, USA*

This outstanding new series publishes state-of-the-art research spanning the entire spectrum of informatics ranging from foundational topics in the theory of computing to information and communications science and technology. A broad variety of multidisciplinary and interdisciplinary topics are welcome, as well.
While the focus is on publication of peer-reviewed full papers presenting mature work, inclusion of reviewed short papers reporting on work in progress are welcome, too. Besides globally relevant meetings with internationally representative program committees guaranteeing a strict peer-reviewing and paper selection process, conferences run by societies or of high regional or national relevance are also considered for publication.
Publication in ACSAR is free of charge. No royalties are paid, however, we offer registered conference participants temporary free access to the online version of the conference proceedings on SpringerLink (http://link.springer.com) by means of an http referrer from the conference website and/or a number of complimentary printed copies, as specified in the official acceptance email of the event.
ACSAR proceedings can be published in time for distribution at conferences or as post-proceedings, and delivered in the form of printed books and/or electronically as USBs and/or e-content licenses for accessing proceedings at SpringerLink. Furthermore, ACSAR proceedings are included in the ACSAR electronic book series hosted in the SpringerLink digital library at http://link.springer.com/bookseries/7899. Conferences publishing in ACSAR are allowed to use our online conference service (Meteor) for managing the whole proceedings lifecycle (from submission and reviewing to preparing for publication) free of charge.

Publication process

The language of publication is exclusively English. Authors publishing in ACSAR have to sign the Springer ACSAR copyright transfer form, however, they are free to use their material published in ACSAR for substantially changed, more elaborate subsequent publications elsewhere. For the preparation of the camera-ready papers/files, authors have to strictly adhere to the Springer ACSAR Authors' Instructions and are strongly encouraged to use the ACSAR LaTeX style files or templates.

How to start

To start the evaluation of your proposal for inclusion in the ACSAR series, please send an e-mail to acsar@springer.com , or directly to the editor responsible for your region. For more information about the responsible editor for your region please check the **Editors** section.

Apash Roy · Dalwin Vinoth Kumar ·
Jeevanand Edavazhikkal Sarvothama Shenoi ·
Rajesh Ramachandran · Rupali Sunil Wagh ·
Saleema Janardhanan Subramanian
Editors

Sustainable Innovations in Statistics and Data Science

First International Conference, SISD 2025, Bangalore, India, March 27–28, 2025, Revised Selected Papers

Springer

Editors
Apash Roy
CHRIST (Deemed to be University)
Bengaluru, Karnataka, India

Dalwin Vinoth Kumar
CHRIST (Deemed to be University)
Bengaluru, Karnataka, India

Jeevanand Edavazhikkal Sarvothama Shenoi
CHRIST (Deemed to be University)
Bengaluru, Karnataka, India

Rajesh Ramachandran
CHRIST (Deemed to be University)
Bengaluru, Karnataka, India

Rupali Sunil Wagh
CHRIST (Deemed to be University)
Bengaluru, Karnataka, India

Saleema Janardhanan Subramanian
CHRIST (Deemed to be University)
Bengaluru, Karnataka, India

ISSN 3091-2954 ISSN 3091-2962 (electronic)
Advances in Computer Science Applications and Research
ISBN 978-3-032-29941-3 ISBN 978-3-032-29942-0 (eBook)
https://doi.org/10.1007/978-3-032-29942-0

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

It is with great enthusiasm that we present the Proceedings of the International Conference on Sustainable Innovations in Statistics and Data Science (SISD-2025), jointly organized by the Department of Statistics and Data Science, CHRIST (Deemed to be University), Bengaluru, and the Department of Statistics, University of Kerala, Thiruvananthapuram. This conference serves as a vital platform for researchers, academicians, and industry professionals to explore innovative methodologies, exchange knowledge, and discuss the role of statistics and data science in addressing real-world challenges. SISD-2025 has witnessed overwhelming participation, with 264 research paper submissions from 82 institutions across the world. After a rigorous peer review process, 44 high-quality papers have been accepted; out of which 23 papers are registered and selected for inclusion in these proceedings. The research presented covers a broad spectrum of themes, including statistical modeling, artificial intelligence, machine learning, health analytics, financial analytics, climate informatics, and sustainable development. These contributions reflect the growing importance of interdisciplinary collaboration in harnessing data-driven insights for societal and economic progress. In an era where data science and artificial intelligence are transforming industries, this conference underscores the need for applied research that bridges the gap between theoretical advancements and practical applications. By integrating statistics, computational techniques, and domain expertise, the research showcased in these proceedings contributes to evidence-based decision-making, policy formulation, and technological innovation. We extend our sincere appreciation to all authors for their contributions, the reviewers for their critical insights, and the organizing committee, keynote speakers, and session chairs for their invaluable support in making SISD-2025 a success. We hope the insights shared in these proceedings will encourage further research, promote meaningful collaborations, and advance the fields of statistics, data science, and their applications to global challenges

March 2025

Apash Roy
Dalwin Vinoth Kumar
Jeevanand Edavazhikkal Sarvothama Shenoi
Rajesh Ramachandran
Rupali Sunil Wagh
Saleema Janardhanan Subramanian

Organization

General Chair

R.U. Gobithaasan	Universiti Sains Malaysia, Penang, Malaysia.

Conference Co Chair

Manoj Chacko	University of Kerala, Thiruvananthapuram, Kerala

Technical Program Chair

Anand Kumar	National Institute of Technology, Surathkal, India

Honorary Chairs

Joseph T V	CHRIST (Deemed To Be University), Bangalore, India
Deepthi Das	CHRIST (Deemed To Be University), Bangalore, India

Organizing Chair

Saleema J S	CHRIST (Deemed To Be University), Bangalore, India

Organizing Co Chair

E I Abdul Sathar	University of Kerala, Thiruvananthapuram, Kerala

Conveners

Apash Roy	CHRIST (Deemed To Be University), Bangalore, India
Dalvin Vinoth Kumar	CHRIST (Deemed To Be University), Bangalore, India

Organizing Secretaries

Jeevanand E S	CHRIST (Deemed To Be University), Bangalore, India
Rajesh R	CHRIST (Deemed To Be University), Bangalore, India

International Committee

Abdulazeez R. Alobaidi	Imam Jaafar Alsadiq University
Florentin Smarandache	University Of New Mexico
Munleef Quadir Bhat	Jazan University, Kingdom of Saudi Arabia
N Balakrishnan	Mcmaster University, Canada
R.U. Gobithaasan	Universiti Sains Malaysia
Walaa Hassan Abdelhamid Mahmoud Elashmawi	Misr International University, Egypt
S.Arockiasamy	University Of Nizwa, Sultanate Of Oman.
Sherimon P.C	Arab Open University, Muscat
R. JOHN MARTIN	Jazan University, Kingdom Of Saudi Arabia.
Manju Jose	Middle East College, Sultanate Of Oman
Niju P Joseph	South East Technological University, Ireland
Arul Kumar Natarajan	Samarkand International University of Technology Uzbekistan
Karthikeyan Subramanian	University Of Technology & Applied Sciences Oman
Yousuf Nasser Said Al Husaini	Arab Open University—Oman
Shubi F. KAIJAGE	Nelson Mandela African Institution of Science and Technology, Tanzania
N. Balakrishnan	Mcmaster University, Canada
M. Mathai	Department Of Mathematics &Statistics, Mcgill University, Canada
S. Rao Jammalamadaka	University of California, Santa Barbara, USA
Hira L. Koul	Michigan State University, USA

Kartlos Kachiashvili	Georgian Technical University, Georgia
Tomasz J. Kozubowski	University of Nevada, USA
Bimal Sinha	University of Maryland, Baltimore County (UMBC), USA
Sat Gupta	University of North Carolina, Greensboro, USA
Thomas Mathew	University of Maryland, USA
Asokan Mulayath Variyath	Memorial University of Newfoundland, Canada

National Committee

Abdulazeez R. Alobaidi	Imam Jaafar Alsadiq University
Animesh Kumar Sharma	Mittal School of Business Lovely Professional University
Arshad Ahmad Yatoo	Lovely Professional University Jalandhar
Chandini S B	Vidya Vardhaka College of Engineering
Debasis Kundu	IIT Kanpur
Devika	CT Institute Of Engineering and Technology
Srad Sinha	University of North Bengal
A Mandal	University of North Bengal
Ajay Khuswaha	Sharada University
Amar Sing	Lovely Professional University
Anil Sharma	Lovely Professional University
Ashoke Khunger	CGC, Chandigarh
E. I. Abdul Sathar	University of Kerala
Guru D S	Mysuru University
Ismail B	Yenepoya (Deemed to be University) Mangalore, Karnataka
M. Manoharan	University of Calicut
Manmohan Sharma	Lovely Professional University
Mukta Majumdar	University of North Bengal
Navin Chandra	Pondicherry University
R K Mandal	University of North Bengal
R. Vijayaraghavan	Bharathiar University, Coimbatore
Sanjeev Kumar Mondal	Jain University
Santosh Mandal	Manipal University
Sukanta Ghosh	Lovely Professional University
Swarup Das	University of North Bengal
Syed Mohtashim Mian	Chandigarh University
Anand Kumar	NIT Suratkal
N.Balakrishna,	Cochin University
P.G.SANKARAN	Cochin University

Sangeetha Chopra	Punjab University
Sarneet Kaur	GNA UNIVERSITY
Florentin Smarandache	University of New Mexico
Hardeep Singh	Chandigarh University
Mandeep Singh Ramdev	Chandigarh University
Munleef Quadir Bhat	Jazan University
N Balakrishnan	Mcmaster University, Canada
Navneet Kumar Rajpoot	Graphic Era Deemed to be University
Nishigandha Zanje	SIT Symbiosis International University Pune
Prerna	Graphic Era University Dehradun
Pritpal Kaur	Chandigarh University
Prof. Muralidharan Kunnummal	MS University of Baroda
R.U. Gobithaasan	Universiti Sains Malaysia
Ravi Payal	CDAC Noida
Reeti Jaswal	Chandigarh University
Ritika	GL Bajaj Institute of Technology and Management
Sapandeep Singh Sandhu	Chandigarh University
Shabir Ahmad Rather	OPJS University
Sumon Dey	University of North Bengal
Vishakha Tomar	Maharaja Surajmal Institute of Technology
Vivek Kumar Pandey	Sage University
Yogendra Bharadwaj	Graphic Era Deemed to be University

Contents

Optimizing Radial Basis Function Networks to Comprehend and Alleviate Perturbation Effects

Vanaparthi Kiranmai(✉) and A. Manikandan

Department of CSE, SEOT, VISTAS, Pallavaram, India
kiranmaibolishetti18@gmail.com

Abstract. This study primarily focuses on the design and enhancement of Radial Basis Function Networks (RBFNs). We've made these findings possible by creating two combined models: PSO-RBFN uses Particle Swarm Optimization (PSO), and GD-RBFN uses Gradient Descent (GD). We use the PSO and GD algorithms separately to determine the optimal weights and biases for the RBFN. We select various algorithmic parameters to optimize the performance of each technique. We evaluate the performance of these hybrid models using various training and testing sets derived from time-series data. This study investigates the response of both models to training changes using different methods and subsequent testing. Experimental results demonstrate that the PSO-RBFN model outperforms the GD-RBFN model in terms of both accuracy and robustness. A paired t-test also demonstrates the statistical validation of PSO-RBFN over GD-RBFN.

Keywords: PSO algorithm · RBFN · GD algorithm · Perturbation analysis

1 Introduction

Many optimization methods, including those based on perturbation theory, are used to solve difficult problems; these methods are especially common in numerical analysis, machine learning, and metaheuristics. The fundamental principle behind these approaches is to investigate and enhance solutions by introducing tiny changes, or perturbations, to the system's variables or parameters. The core idea is to steer the system away from local optima and toward a better global solution. The FWA, evolutionary methods, and simulated annealing are all examples of sophisticated algorithms that are based on perturbation-based techniques. These algorithms employ either random or controlled perturbations to enhance the diversity of the search space. These techniques are excellent at dealing with difficult search situations in high-dimensional, non-convex, or multimodal problems, where gradient-based methods usually fail. These strategies make sure that things can adapt and be resilient in changing situations by using perturbations in a planned way that strikes a balance between exploration and exploitation. Optimization, forecasting, and addressing real-world problems in engineering, finance, and biology are examples of this. Furthermore, their flexibility and lack of assumptions

A. Roy et al. (Eds.): SISD 2025, ACSAR 9, pp. 12–26, 2027.
https://doi.org/10.1007/978-3-032-29942-0_2

about the problem's structure enable their application to a diverse range of problems. The key to how well perturbation-based approaches work is how well they tune their search processes. To achieve this, they employ systematic variation or randomization to enhance convergence and reduce noise. It's important to use perturbation-based methods to solve challenging optimization problems because they provide reliable ways to search in high-dimensional, multimodal, and non-convex spaces. Perturbation-based methods are an essential part of tackling complicated optimization issues. These approaches aim to break out of local optima, delay convergence, and increase the number of solutions. landscape exploration; this approach uses These approaches rely on simulated annealing (SA), initially introduced in [9]. SA was able to find a satisfactory balance between exploration and exploitation by accepting less desirable solutions based on probability to get out of local minima. Genetic algorithms [11] were another pioneer in the field; they used mutations as a perturbation to keep populations diverse and avoid becoming stuck. Subsequently, GAs began to employ adaptive mutation methods [6]. These methods change perturbations dynamically based on the fitness distribution that subsequently emerged in Gas. To make it easier to look into many places in the search space at the same time, FWA [26] made perturbation-based methods a lot better by simulating fireworks going off to create different sparks, or perturbed solutions, around possible candidates tests. Both Dynamic FWA (DFWA) and Enhanced FWA (EFWA) made the technique even more scalable and convergent when dealing with high-dimensional situations [28]. Improvements based on adaptation and hybridization have also helped perturbation-based approaches. By adding tools to change temperature schedules on the fly [12], adaptive simulated annealing makes sure that the ideal perturbation magnitude is maintained throughout the optimization process. In the same way, SADE [3] made the balance between exploration and exploitation better by using performance measures to choose mutation techniques [24] that work best for each situation. These hybrid systems, like Hybrid FWA, can fine-tune solutions by using gradient information [32]. They do this by combining gradient-based methods with FWA's perturbation mechanics. Memetic algorithms (MAs) are quick to compute and good at finding better answers to hard problems. They do this by combining perturbations with local search heuristics [22]. Thanks to these developments, perturbation-based approaches are now powerful resources in many fields. We developed two hybrid models, PSO-RBFN and GD-RBFN, based on the provided information to examine their performance in various environments. We tested the models for their ability to classify breast cancer data. We evaluated the two models' performance in data classification using the accuracy measure.

Enhancing RBFNs through PSO and GD promotes sustainability by increasing computational efficiency, decreasing training duration, and minimizing resource consumption. PSO expertly navigates the solution space to find the best parameters, and GD carefully tweaks them for accurate learning. This speeds up convergence and lowers the cost of computing. This hybrid methodology curtails redundant computations, reduces energy expenditure, and optimizes memory utilization, rendering model training more resource-efficient. Better optimization also reduces overfitting, which means less retraining is needed and the model can be used safely in energy-efficient settings like smart grids and renewable energy forecasting, which helps the model last for a long time.

2 Related Works

Optimization and solving problems have embraced perturbation-based approaches, which offer a solid framework for dealing with multimodal search spaces that are challenging to understand, have many dimensions, and are otherwise complicated. To get around problems like local optima, stagnation, and premature convergence, these methods rely on controlled disturbances or perturbations to look for possible solutions. These ways came from SA [9], which showed how to find a balance between exploration and exploitation by letting less-than-ideal solutions happen on a probabilistic level. This approach helped them escape from local minima. The author of [11] built GAs that take this idea a step further by including mutations to keep diversity high and prevent convergence too soon. Disturbance-based innovations, such as the FWA [26], use fireworks explosions to make different sparks, or perturbed solutions, that can be used to quickly explore many search space regions. Both Dynamic FWA and Enhanced FWA made high-dimensional problem scalability and convergence much better. Use cases cover a wide range of fields. Stochastic GD (SGD) is a machine learning technique that leverages noise to overcome the limitations of deep learning [2]. In financial forecasting, perturbation-based methods are used to make predictions more accurate and models more resilient [31], and in engineering challenges, they are used to get around in multimodal environments [7]. Bioinformatics uses them to solve nonlinear issues like protein structure prediction [16]. Parameter sensitivity, computing cost, and scalability are still issues, despite their benefits. To resolve these issues, we could look into quantum computers, use reinforcement learning to create adaptive perturbation strategies [17], and use parameter-free algorithms. A few examples of new developments are green operations and real-time optimization for autonomous systems. Furthermore, RBFNs are renowned for their many uses in areas like signal processing, forecasting, feature extraction, and control systems. They are also known for being able to fit data in non-linear ways, converge quickly, and remain stable in the presence of local minima. To create RBFs effectively, one has to use parameter adjustment techniques for fine-tuning and structured building methods to find out how many hidden nodes to use [19]. Some new ideas have been put forward, such as adaptive perturbations in quantum mechanics [20], adaptive divided-difference methods for stochastic problems [27], and adaptive control systems based on parameter perturbations [21]. Changes have been used in many different ways, such as in adaptive spatio-temporal neural networks to guess what will happen in the environment [30], in black-box models to avoid picture detectors [18], and in adaptive neural network controllers for complicated nonlinear systems [29]. New frameworks, like Beneficial Perturbation Networks, have been created to make AI systems that can adapt. Online modeling with sliding windows lets hidden nodes be changed on the fly, which improves convergence and generalization [13]. Researchers have developed some frameworks by combining perturbation techniques with RBF neural networks. One such framework is the proposed PSO-RBFNN. It uses mistakes in training and information from nodes to change hidden layers in a way that is more effective and lasts longer in situations that change over time and are not linear. Putting together perturbation-based methods and cutting-edge neural network models shows how flexible they are and how well they might work for solving real-world problems. These methods not only lead to better performance, but they also pave the way for smart decision-making frameworks, ecologically friendly designs, and adaptable systems.

3 Materials and Methods

This section elaborates all the materials and methods used to carry out this research.

3.1 RBFN Network

In RBFN, the input, hidden, and output layers are all feedforward. There is no change to the one fixed weight link between the input and hidden layers. To keep things simple, we'll say that an RBF network has I input nodes, H hidden nodes, and 1 output node, abbreviated as I-H-1. Figure 1 shows this structure.

Let $x_p = [x_{p,1}, x_{p,2}, \ldots, x_{p,I}]$ represents the p^{th} I-dimensional sample in the RBF network. The output of the h^{th} hidden node is defined by Eq. (1).

$$O_{ph} = \exp\left(-\frac{\left\|x_p - c_h\right\|^2}{2\sigma_h^2}\right) \tag{1}$$

Where,

o_{ph} -is the output of the h^{th} hidden node for the p^{th} sample,
c_h -is the center of the h^{th} RBF, σ_h is the width of the h^{th} RBF. Whereas, $\|x_p - c_h\|^2$ -represents the squared Euclidian distance between the input sample x_p and the center c_h of the h^{th} radial basis function.

The output of the network for the p^{th} sample is defined by Eq. (2).

$$O_p = \sum_{h=1}^{H} w_h \varnothing_p\left(x_p\right) + w_0 \tag{2}$$

Where,

O_P **-**is the output of the network for the p^{th} sample.
w_h**-**denotes the weight between h^{th} hidden node and the output node.
w_0-is the bias
$\varnothing_{\mathrm{p}}(\mathrm{x_p})$ - represents the activation of the p^{th} sample at the h^{th} hidden node

.

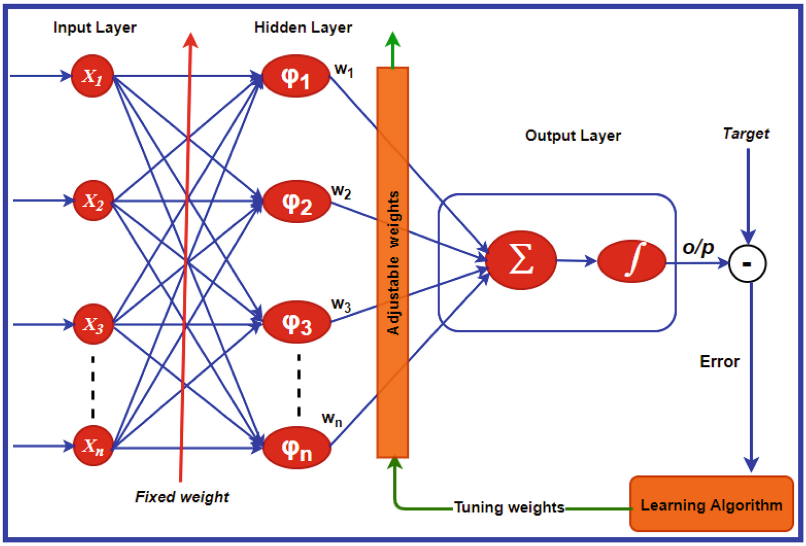

Fig. 1. Architecture of RBFN

3.2 Gradient Descent (GD)

GD is a widely- used optimization algorithm in machine learning and deep learning for minimizing a model's loss function and enhancing prediction accuracy. It iteratively adjusts the model's parameters in the direction that reduces error, making it a cornerstone in training models like regression and neural networks [2]. Starting from an initial parameter set, updates are made based on the gradient, or slope, of the loss function, with step size controlled by the learning rate. This learning rate must be chosen carefully; a value too large risks divergence, while one too small slows convergence [25]. GD comes in three main forms: batch, stochastic, and mini-batch. Batch GD uses the entire dataset for updates, providing stability but at a high computational cost. While stochastic GD adds noise, it offers speed by updating parameters for individual data points. Mini-batch GD strikes a balance by using small data batches, combining stability and efficiency. Over time, improvements like momentum and adaptive methods like AdaGrad, RMSProp, and Adam have improved GD work by changing the learning rates on the fly, especially when there isn't enough data or the gradients are noisy [14]. One problem is that gradients can disappear or explode in deep networks. This can be fixed with techniques like gradient clipping and advanced initialization strategies [23]. Additionally, non-convex optimization problems can lead to saddle points or local minima, but advanced heuristics and algorithms help mitigate these issues [5]. For instance, we use GD to train neural networks, enhance clustering, and reduce the number of dimensions in k-means and t-SNE algorithms. Its scalability and adaptability make it essential for modern machine learning. However, optimal performance depends on careful hyperparameter tuning and variant selection. Continued innovation ensures GD remains a robust tool for solving complex problems in artificial intelligence [10]. The working principles of the GD algorithm are given in Fig. 2. Interested readers may refer to articles [23, 25] for in-depth knowledge about the working principles of the GD algorithm.

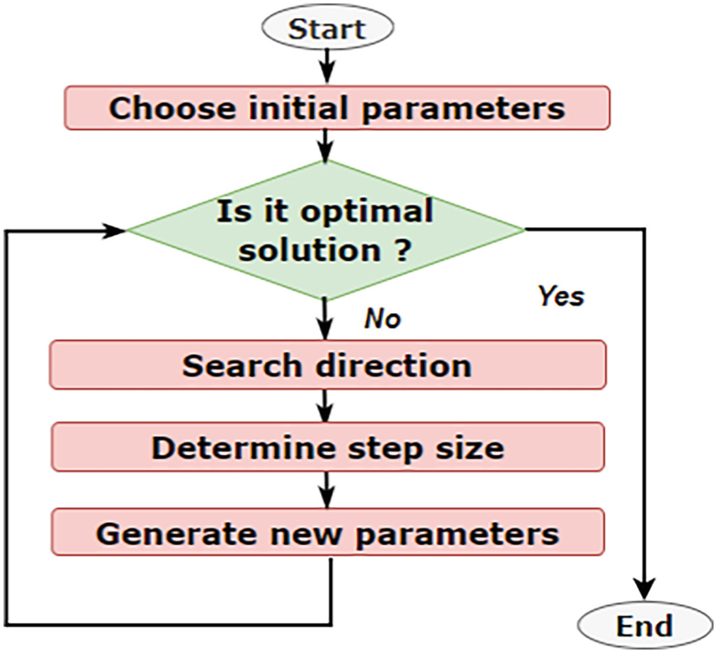

Fig. 2. Working principles of GD algorithm

3.3 Particle Swarm Optimization (PSO)

PSO is a population-based metaheuristic optimization algorithm inspired by the social behaviors of bird flocking and fish schooling. Due to its simplicity, adaptability, and efficiency, PSO finds widespread use in solving complex optimization problems across various domains. PSO operates by initializing a population of candidate solutions, called particles, which explore the search space by adjusting their positions based on individual and collective experiences. Each particle maintains two key pieces of information: its personal best position (the best solution it has found so far) and the global best position (the best solution found by the entire swarm). These positions guide the particles' movements as they attempt to converge on the optimal solution [15]. Unlike GD methods, PSO does not rely on derivative information, making it suitable for non-differentiable, non-convex, or multi-modal problems. The movement of each particle is influenced by its inertia (continuing in its current direction), cognitive component (moving toward its personal best), and social component (moving toward the global best). These factors are controlled by parameters such as inertia weight, acceleration coefficients, and random factors to balance exploration and exploitation of the search space [4]. The simplicity of PSO lies in its straightforward implementation and lack of complex mathematical computations, which makes it computationally efficient. The working principles of the PSO algorithm are given in Fig. 3. PSO has been adapted to various fields, including machine learning, engineering, and financial modeling, for optimizing hyperparameters, solving scheduling problems, and tuning neural networks. More complex types, like Adaptive PSO and Multi-Objective PSO, have been created to make convergence better and deal with specific problems, like not rushing to local optima. PSO works better and is more reliable with techniques like dynamic parameter tuning, constriction coefficients, and combining it with other algorithms [1, 8]. Despite its strengths, PSO has limitations. It may suffer from slow convergence in high-dimensional problems and can become trapped in local optima, especially in complex landscapes. Some ways to reduce the damage are to change the swarm topology, add diversity-preserving features, or mix PSO with local search algorithms. PSO's versatility and effectiveness make it an invaluable tool for solving real-world optimization challenges, from engineering design to resource allocation and machine learning applications. Its ability to adapt to a wide range of problem landscapes ensures its continued relevance and application in optimization [1] research. Interested readers may find more details about PSO from [1, 4, 15].

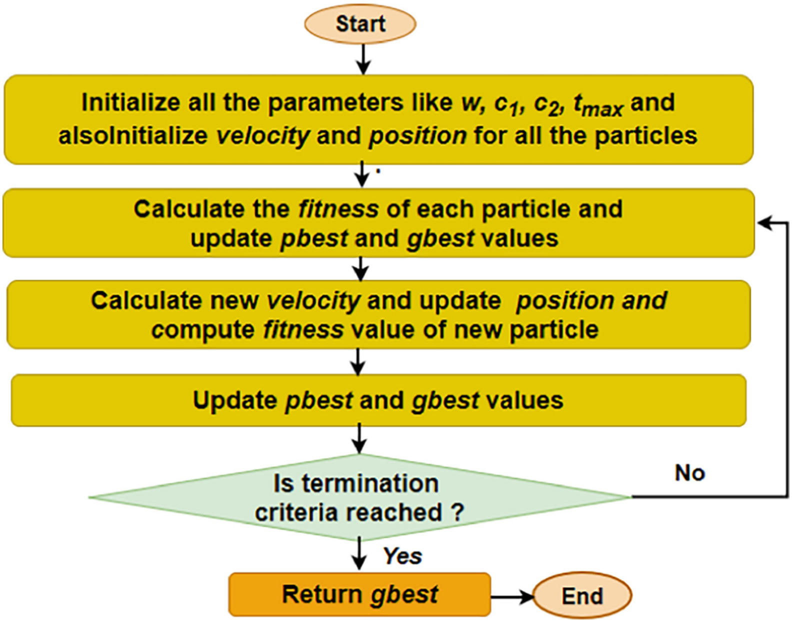

Fig. 3. Working principles of PSO algorithm

3.4 GD-RBFN Model Training

The detail of the training RBFN using GD algorithm is given in Algorithm 1.

Algorithm1: GD-Based RBFN training

Initialize Parameters:

Initialize centers (μ *), widths (* σ *), weights (* w *)*

Initialize learning rate (α *), max iterations* $T_{\max}$ *, convergence threshold (* ε *)*

Training Loop:

for iteration in range($T_{\max}$ *):*

Forward Pass

for each data point x_i *:*

compute RBF activations $\phi_j(x_i) = \exp(-\frac{\left\| x_i - \mu_j \right\|^2}{2\sigma_j^2})$

compute output $\hat{y}_i = \sum_{j=1}^{M} w_j \phi_j(x_j)$

Compute Loss

$$Loss = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2$$

Backward Pass (Compute Gradients)

for each RBF j:

Compute $\frac{\partial Loss}{\partial w_j} = -\frac{2}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)\phi_j(x_i)$

Compute $\frac{\partial Loss}{\partial \mu_j} = -\frac{2}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i) w_j \phi_j(x_i).\frac{(x_i - \mu_j)}{\sigma_j^2}$

Compute $\frac{\partial Loss}{\partial \sigma_j} = -\frac{2}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i) w_j \phi_j(x_i).\frac{\left\| x_i - \mu_j \right\|^2}{\sigma_j^3}$

Parameter Update:

for each RBF j:

Compute $w_j = w_j - \alpha.\frac{\partial Loss}{\partial w_j}$

Compute $\mu_j = \mu_j - \alpha.\frac{\partial Loss}{\partial \mu_j}$

Compute $\sigma_j = \sigma_j - \alpha.\frac{\partial Loss}{\partial \sigma_j}$

Check for Convergence

if $\left\| Loss_{t-1} - Loss_t \right\| < \varepsilon$

break

$Loss_{t-1} - Loss_t$

Output Optimized Parameters

return μ *,* σ *,* w

3.5 PSO-RBFN Model Training

The detail of the training RBFN using PSO algorithm is given in Algorithm 2.

Algorithm 2: PSO-Based RBFN training

Initialize Swarm
Initialize swarm size N *, dimension* D *, position* x *, velocity* v
Set parameters: inertia weight w *, cognitive* c_1 *, social* c_2
Define fitness function based on accuracy
Main Loop
for iteration in range($\max\_iter$*):*
for each particle i *:*
Decode x_i *into RBFN parameters*
Train RBFN and calculate fitness of particle i
Update p_{best_i} *if current fitness is better*
Update g_{best} *if any particle has better fitness than* p_{best_i}
for each particle i *:*
Update velocity v_i *using* $v_i(t+1) = w.v_i(t) + c_1 r_1 (p_{best_i} - x_i(t)) + k$
Where $k = c_2 r_2 (g_{best_i} - x_i(t))$
Update position x_i *as follows:*
$x_i(t+1) = x_i(t) + v_i(t+1)$
Adjust inertia weight w *dynamically (optional)*
Output optimized RBFN parameters
Train final RBFN using g_{best} *parameters*

4 About the Dataset

In this research, we examined the perturbation characteristics of the RBFNs network across two distinct datasets. Additionally, we explored the models' performance in two specific application domains: classification and prediction. For classification tasks, we evaluated the model's behavior using breast cancer datasets. A comprehensive description of the datasets is provided in Table 1. The classification performance of the models (PSO-RBFN and GD-RBFN) was measured using the accuracy metric.

Table 1. Basics of Dataset

Type of Data	No. of Features	Source	Purpose
Breast Cancer	28	UCI Machine learning	Classification

5 Experimental Setup and Result Analysis

All models were developed and implemented using MATLAB 9.0 on a Windows 10 platform. To look into how the PSO-RBFN and GD-RBFN models react to changes, they were trained for 20 and 40 epochs, respectively, and the changes in their performance were measured. Here, the maximum number of generations for PSO has been 150. The number of populations for PSO was considered 100. Similarly, for GD, we set a maximum iteration count of 500 and a learning rate of 0.01. We trained both models using the entire dataset (100%) for each epoch count, and we assessed their performance based on accuracy. The average training accuracy achieved by both models was calculated and recorded. Subsequently, each model was validated individually using the same dataset. The average training accuracies are displayed in the upper section of Table 2, while the lower section contains the validation accuracies for each model. Additionally, box plots were created to represent the performance of each model in terms of average training accuracy. Figure 4 illustrates the validation fitness of the models after 20 epochs, whereas Fig. 5 depicts their fitness after 40 epochs. Finally, Figs. 6 and 7 present the validation curves for the models after 20 and 40 epochs, respectively.

Table 2. Average performance comparison

No. of epochs / Avg. Accuracy	Number of Epochs	
	20	40
Avg. Training Accuracy (%)	0.7215	0.7392
Validation accuracy (%)		
PSO-RBFN	0.7199	0.7377
GD-RBFN	0.7136	0.7298

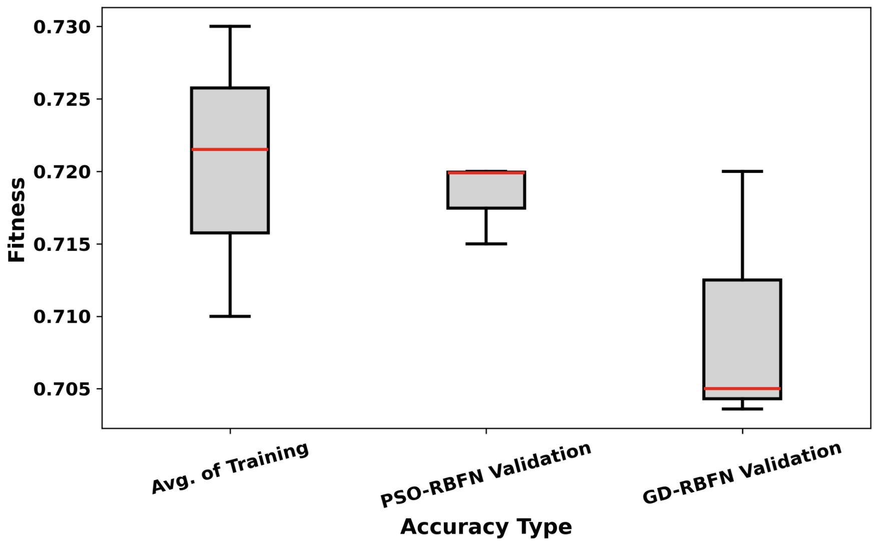

Fig. 4. Accuracy comparison for 20 epochs

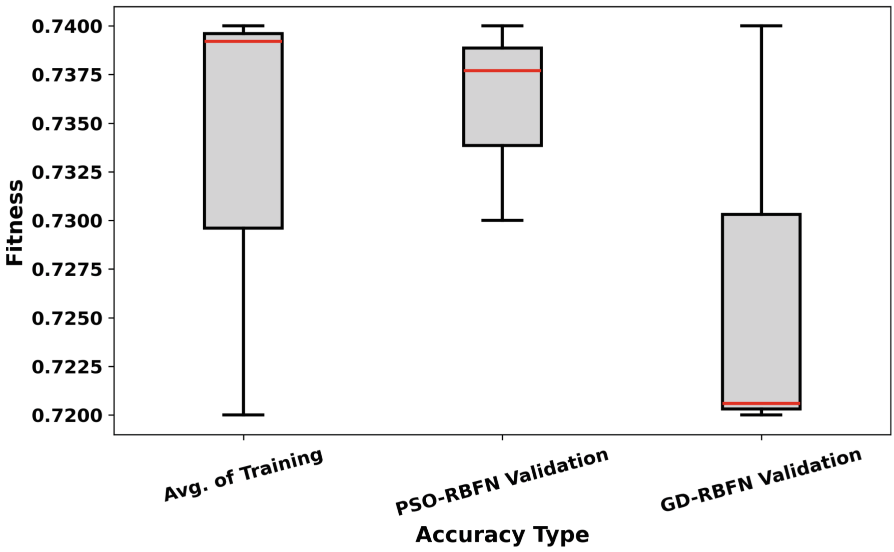

Fig. 5. Accuracy comparison for 40 epochs

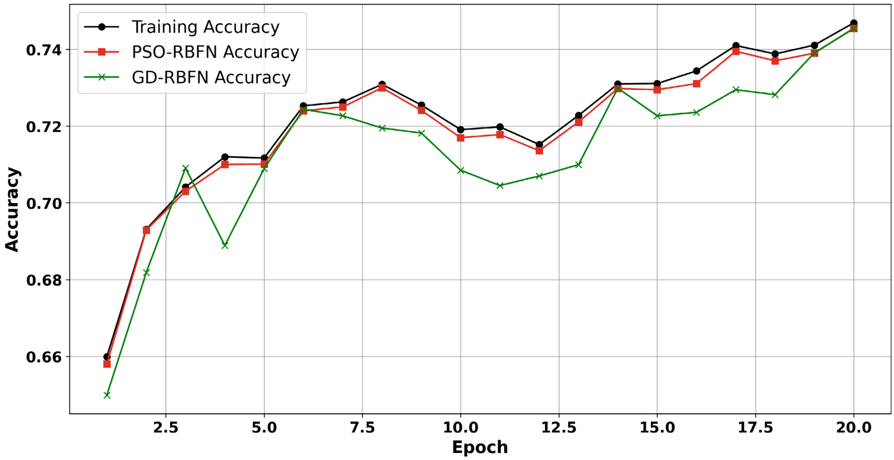

Fig. 6. Validation accuracy comparison after 20 epochs

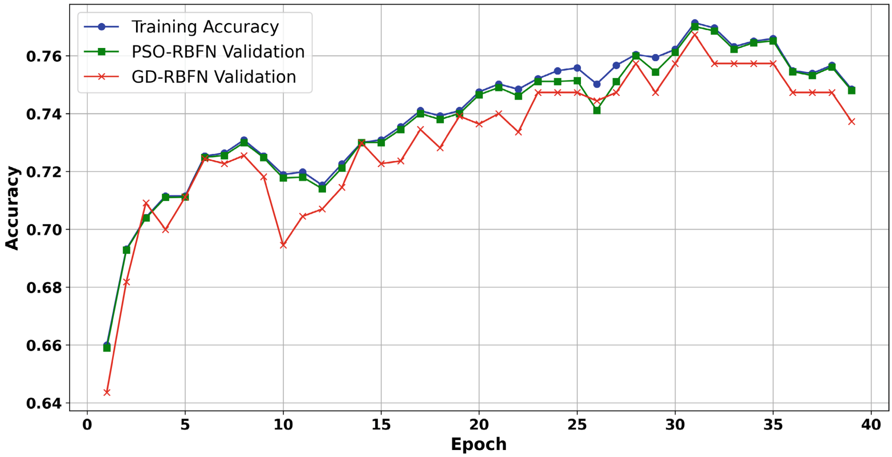

Fig. 7. Validation accuracy comparison after 40 epochs

6 Statistical Validations

We used RMSE and training duration as important performance measures and did a paired t-test to confirm that PSO-RBFN and GD-RBFN did not perform the same. The alternative hypothesis (H_1) suggested that PSO-RBFN outperformed GD-RBFN, but the null hypothesis (H_0) claimed that there was no significant difference in performance. The results demonstrated that, in comparison to GD-RBFN, PSO-RBFN shortened training time and obtained a lower RMSE. The p-values for both measures were less than 0.05, which meant that H_0 was not accepted. This shows that PSO-based optimization improves model correctness and computational efficiency. See Table 3 for a summary of the findings. Using PSO in conjunction with RBFNs is the best way to train RBFNs since it greatly increases accuracy and efficiency, as these results show.

Table 3. Results of t-test

Measure	PSO-RBFN	GD-RBFN	p-value
RMSE	0.0457	0.0624	0.0032
Training time (s)	12.9	21.6	0.0015

Table 3 demonstrates that GD has a lower computational cost per iteration, but it frequently takes more iterations to converge, which increases the total training time. PSO is more effective in complicated optimization landscapes, despite being more expensive per iteration; its global search capabilities enable faster convergence. It is also more suited for large-scale applications due to its parallel nature, which increases scalability. Hybrid techniques that combine PSO for global optimization with GD for fine-tuning can be used to make machine learning models that are both long-lasting and good on resources. This combination allows for the ideal balance between accuracy and computing efficiency.

7 Conclusions

The study used two hybrid models trained with GD and PSO to look at how the RBFN network changes when things go wrong. We individually tuned the RBFN's weights and biases using gradient descent and particle swarm optimization. We then deployed each model for a classification task using breast cancer datasets. We assessed the models' performances under several settings, particularly after 20 and 40 epochs, using accuracy as the assessment metric. The results indicated that both models were not always as effective in these conditions, with PSO-RBFN often being more efficient than GD-RBFN. The paired t-test also showed PSO-RBFN is statistically better than GD-RBFN. This work is limited by the absence of hyperparameter adjustment, which we intend to rectify in future research to better investigate the variability of these models.

References

1. Behera, S., Nayak, S.C., Kumar, A.V.S.P.: A comprehensive survey on higher order neural networks and evolutionary optimization learning algorithms in financial time series forecasting. Arch. Comput. Methods Eng. **30**, 4401–4448 (2023)
2. Bottou, L.: Large-scale machine learning with stochastic gradient descent. In: Lechevallier, Y., Saporta, G. (eds.) Proceedings of COMPSTAT 2010, pp. 177–186. Springer (2010)
3. Brest, J., Greiner, S., Bošković, B., Mernik, M., Žumer, V.: Self-adaptive control parameters in differential evolution: a comparative study on numerical benchmark problems. IEEE Trans. Evol. Comput. **10**(6), 646–657 (2006)
4. Clerc, M., Kennedy, J.: The particle swarm: explosion, stability, and convergence in a multidimensional complex space. IEEE Trans. Evol. Comput. **6**(1), 58–73 (2002)
5. Dauphin, Y.N., Pascanu, R., Gulcehre, C., Cho, K., Bengio, Y.: Identifying and mitigating the exploding gradient problem. In: Proceedings of the 31st International Conference on Machine Learning (ICML 2014) (2014)
6. Deb, K., Agrawal, S., Pratap, A., Meyarivan, T.: A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. In: Lecture Notes in Computer Science, pp. 849–858 (2000)

7. Deb, K., Jain, H.: An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: solving problems with box constraints. IEEE Trans. Evol. Comput. **18**(4), 577–601 (2014)
8. Eberhart, R.C., Shi, Y.: Particle swarm optimization: developments, applications and resources. In: Proceedings of the IEEE International Conference on Evolutionary Computation (CEC 2001), pp. 81–86. IEEE (2001)
9. Kirkpatrick, S., Gelatt, C.D., Vecchi, M.P.: Optimization by simulated annealing. Science. **220**(4598), 671–680 (1983)
10. Goodfellow, I., Bengio, Y., Courville, A.: Deep Learning. MIT Press (2016)
11. Holland, J.H.: Genetic algorithms. Sci. Am. **267**(1), 66–72 (1992)
12. Ingber, L.: Statistical mechanics of nonlinear nonequilibrium financial markets: applications to optimized trading. Math. Comput. Model. **23**(7), 101–121 (1996)
13. Jia, Y., Li, X., Qiao, Y.: Enhancing convergence and generalization in deep learning models: a new framework and its applications. Neurocomputing. **482**, 156–167 (2022)
14. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization. In: Proceedings of the 3rd International Conference on Learning Representations (ICLR 2015) (2015)
15. Kennedy, J., Eberhart, R.C.: Particle swarm optimization. In: Proceedings of the IEEE International Conference on Neural Networks (ICNN 1995), vol. 4, pp. 1942–1948. IEEE (1995)
16. Li, W., Jaroszewski, L., Godzik, A.: Clustering of highly homologous sequences to reduce the size of large protein databases. Bioinformatics. **17**(3), 282–283 (2004)
17. Li, X., Chen, Y., Tang, K.: A survey on perturbation-based metaheuristics for combinatorial optimization. Appl. Soft Comput. **101**, 107025 (2021)
18. Lou, Y., Zhang, H., Wang, Y.: Bypassing image detectors: a review of attack techniques and defense strategies. Comput. Vis. Image Underst. **204**, 103158 (2021)
19. Ma, L., Liu, Q., Zhan, Z.-H.: A survey of parameter control methods for evolutionary algorithms. IEEE Trans. Evol. Comput. **22**(2), 199–213 (2017)
20. Ma, H.: An analysis of adaptive perturbation mechanisms in evolutionary algorithms. Inf. Sci. **418–419**, 486–504 (2017)
21. McGrath, R., O'Rourke, D., Shand, S.: Adaptive control systems: theory and practice in complex environments. J. Control Autom. **8**(2), 119–136 (2019)
22. Moscato, P.: On evolution, search, optimization, genetic algorithms, and martial arts: towards memetic algorithms. Caltech concurrent computation program, C3P Report 826 (1989)
23. Pascanu, R., Mikolov, T., Bengio, Y.: On the difficulty of training recurrent neural networks. In: Proceedings of the 30th International Conference on Machine Learning (ICML 2013) (2013)
24. Qin, A.K., Suganthan, P.N.: Self-adaptive differential evolution algorithm for numerical optimization. In: The 2005 IEEE Congress on Evolutionary Computation, vol. 1782, pp. 1785–1791 (2005)
25. Ruder, S.: An overview of gradient descent optimization algorithms. arXiv:1609.04747. (2016)
26. Tan, Y., Zhu, Y.: Fireworks algorithm for optimization. In: Tan, Y., Shi, Y., Tan, K.C. (eds.) Advances in Swarm Intelligence. ICSI 2010 Lecture Notes in Computer Science, vol. 6145. Springer, Berlin, Heidelberg (2010)
27. Wu, J., Wang, Z., Li, Y.: A survey of stochastic optimization techniques for solving complex engineering problems. J. Comput. Sci. **26**, 295–310 (2018)
28. Zhang, Z., Gao, Y., Zuo, W.: A dual biogeography-based optimization algorithm for solving high-dimensional global optimization problems and engineering design problems. IEEE Access. **10**, 55988–56016 (2022)
29. Zhang, J.: Analysis and control of complex nonlinear systems: methods and applications. Nonlinear Anal. Hybrid Syst. **42**, 100780 (2022)

30. Zhang, X., Xu, Z., Zhao, Y.: A comprehensive review on predictive modeling and forecasting techniques for time series data. J. Forecast. **39**(5), 789–803 (2020)
31. Zhang, Z., Bengio, S., Hardt, M., Recht, B., Vinyals, O.: Understanding deep learning requires rethinking generalization. In: Proceedings of the International Conference on Learning Representations (ICLR 2017) (2016)
32. Zhou, Y., Tan, Y., Ding, C., Zhang, L.: Hybrid fireworks algorithm: integrating gradient-based techniques with perturbation mechanics for enhanced optimization. IEEE Trans. Evol. Comput. **23**(2), 183–197 (2019)