Jurnal Teknik Informatika (JUTIF)
P-ISSN: 2723-3863
E-ISSN: 2723-3871
Vol. 6, No. 5, October 2025, Page. 3509-3525
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.20256.5.4952

Enhancing Customer Purchase Behavior Prediction Using PSO-Tuned Ensemble Machine Learning Models

Princess Iqlima Kafilla*[1], Fandy Setyo Utomo[2], Giat Karyono[3]

[1,2,3]Master of Computer Science, Universitas Amikom Purwokerto, Indonesia

Email: [1]princess19iqlima@email.ac.id

Received : Jun 25, 2025; Revised : Jul 15, 2025; Accepted : Jul 15, 2025; Published : Oct 16, 2025

Abstract

Predicting customer purchase behavior remains a significant challenge in e-commerce and marketing analytics due to its complex and nonlinear patterns. This study introduces a machine learning framework that integrates ensemble learning models with Particle Swarm Optimization (PSO) for hyperparameter tuning to improve classification accuracy and class discrimination. Several ensemble algorithms, including CatBoost, XGBoost, LightGBM, AdaBoost, and Gradient Boosting, were compared against a baseline Logistic Regression model, both with default and PSO-optimized configurations. Experiments on a real-world e-commerce dataset containing behavioral and demographic variables showed that ensemble methods substantially outperformed traditional models across accuracy, F1-score, and ROC AUC metrics. Notably, the PSO-tuned Gradient Boosting model achieved the highest ROC AUC of 0.9547, improving the AUC by approximately 0.0076 compared to its default configuration, while CatBoost obtained the highest overall accuracy and F1-score. PSO optimization was especially effective in enhancing simpler models such as Logistic Regression but showed marginal gains and some convergence instability in more complex ensemble models. Feature importance analyses consistently identified variables such as time spent on the website, discounts availed, age, and income as key drivers of purchase intent. These findings demonstrate the benefit of combining ensemble learning with metaheuristic optimization, offering actionable insights for developing robust, data-driven marketing strategies.

Keywords : *Customer Purchase Behavior, Ensemble Learning, Hyperparameter Tuning, Machine Learning in Marketing, Particle Swarm Optimization (PSO), Predictive Modeling.*

1. INTRODUCTION

The introductory content generally only covers 10-20% of the entire paper. Don't forget to include the objectives of the research conducted in this paper.

In the era of digital transformation, businesses are increasingly leveraging data analytics to understand customer behavior, particularly purchase intentions, to improve marketing effectiveness, customer engagement, and profitability. The ability to predict whether a customer will make a purchase based on demographic and behavioral data is crucial in shaping personalized marketing strategies, optimizing inventory levels, and allocating promotional resources more efficiently. With the proliferation of e-commerce platforms and digital customer footprints, companies now have access to a vast array of consumer data, including browsing time, purchasing frequency, discounts availed, and demographic profiles [1][2].

However, modeling consumer purchase behavior presents significant challenges due to its non-linear, multifactorial nature. Traditional statistical approaches like logistic regression and decision trees often fall short in capturing these intricate relationships, leading to suboptimal predictions. In response, ensemble learning techniques have gained traction for their robustness and higher predictive power [3][4]. Algorithms such as Gradient Boosting, XGBoost, LightGBM, CatBoost, and AdaBoost integrate multiple weak learners to form a strong classifier, effectively reducing variance and bias. These models have demonstrated superior performance in diverse domains including financial risk modeling, healthcare diagnostics, and retail forecasting [5][6][7].

Despite their effectiveness, the performance of ensemble models is highly sensitive to the configuration of their hyperparameters. Choosing the right values for parameters such as learning rate, number of estimators, and tree depth significantly impacts accuracy, generalization, and training efficiency. Grid search and random search are commonly used for hyperparameter tuning but are computationally intensive and often inefficient in high-dimensional search spaces [8]. Consequently, researchers have explored bio-inspired optimization algorithms such as Particle Swarm Optimization (PSO) for more effective and scalable tuning [9][10].

PSO is a stochastic optimization method inspired by the social behavior of bird flocks or fish schools. It optimizes a problem iteratively by updating a population of candidate solutions based on individual and global best experiences. When applied to machine learning, PSO enables automatic hyperparameter tuning, resulting in improved model performance with reduced manual intervention. It has been successfully utilized in domains like credit risk scoring [11], software defect prediction [12], customer behavior prediction [13], heart disease diagnosis [14], and even image segmentation [15].

In the context of retail analytics, PSO has been shown to significantly improve classification models. One study reported a performance improvement of over 10% in purchase prediction accuracy by optimizing decision tree parameters using PSO [16]. Another approach involved a PSO-tuned neural network to predict online customer behavior in digital marketing campaigns, resulting in a 4.2% increase in accuracy compared to traditional methods [17]. Additional research has applied PSO in combination with ensemble methods for tasks such as software change prediction [18], crop disease detection [19], and customer sentiment summarization [20], demonstrating the versatility and effectiveness of PSO across various domains.

The integration of ensemble learning with PSO-based optimization has also received considerable attention in recent studies. A fuzzy clustering PSO-optimized ensemble model has been proposed for credit risk classification, showing improvements in F1-score and AUC ranging from 10% to 50% across several datasets [21]. In software engineering, weighted-voting ensemble classifiers enhanced with PSO and diverse fitness functions have been developed to improve prediction accuracy [12][18]. In the field of marketing, evolutionary-optimized ensemble models using stacking and voting techniques have significantly improved AUC and F1-score across multiple customer behavior datasets [22]. Similarly, PSO has been employed to optimize SVM kernel parameters for predicting online consumer purchase intent, yielding a notable boost in classification performance [23].

Despite these advancements, there is a research gap in applying PSO-optimized ensemble learning specifically to predict customer purchase behavior using real-world behavioral and demographic datasets. Most prior studies focus either on single algorithms or on unrelated domains such as weather forecasting or medical diagnosis. Furthermore, few studies have provided a comprehensive evaluation of multiple ensemble models under both default and PSO-tuned configurations, including detailed comparisons of ROC AUC, F1-score, and confusion matrices.

This research aims to develop a PSO-tuned ensemble learning framework to predict customer purchase behavior. The study utilizes a comprehensive dataset containing demographic and behavioral features such as age, annual income, number of purchases, time spent on the website, discount utilization, product category, and loyalty program status. Several machine learning models are evaluated including Logistic Regression, Gradient Boosting, XGBoost, LightGBM, CatBoost, and AdaBoost—both in default and PSO-optimized configurations. Performance is assessed using standard classification metrics: accuracy, precision, recall, F1-score, and ROC AUC.

Vol. 6, No. 5, October 2025, Page. 3509-3525
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.20256.5.4952

PSO is applied to optimize key hyperparameters for selected models, with the aim of enhancing performance while maintaining generalizability. The methodology includes data preprocessing, model training, PSO-based optimization, and comparative analysis using cross-validation. Additionally, feature importance and confusion matrix analyses are conducted to gain deeper insights into model behavior and predictive factors.

This research contributes a novel framework by systematically evaluating multiple ensemble machine learning algorithms combined with PSO-based hyperparameter tuning to predict customer purchase behavior using real-world behavioral and demographic data. Unlike prior studies that focus on a single algorithm or domain, this work comprehensively compares default and optimized configurations across diverse ensemble models, providing both theoretical and practical insights for improving predictive marketing strategies. This contribution extends the state-of-the-art by validating PSO-enhanced ensemble learning on an e-commerce dataset, which has not been comprehensively explored in previous literature [24][25].

2. METHOD

Figure 1 illustrates the overall methodological framework employed in this study for predicting customer purchase behavior using ensemble learning models optimized with Particle Swarm Optimization (PSO). The workflow begins with data acquisition, cleaning, and feature engineering, followed by preprocessing through standardized pipelines. The process continues with model construction and baseline evaluation, after which PSO is used to tune hyperparameters for selected models. The optimized models are then retrained and re-evaluated to assess performance improvements over the default configurations.

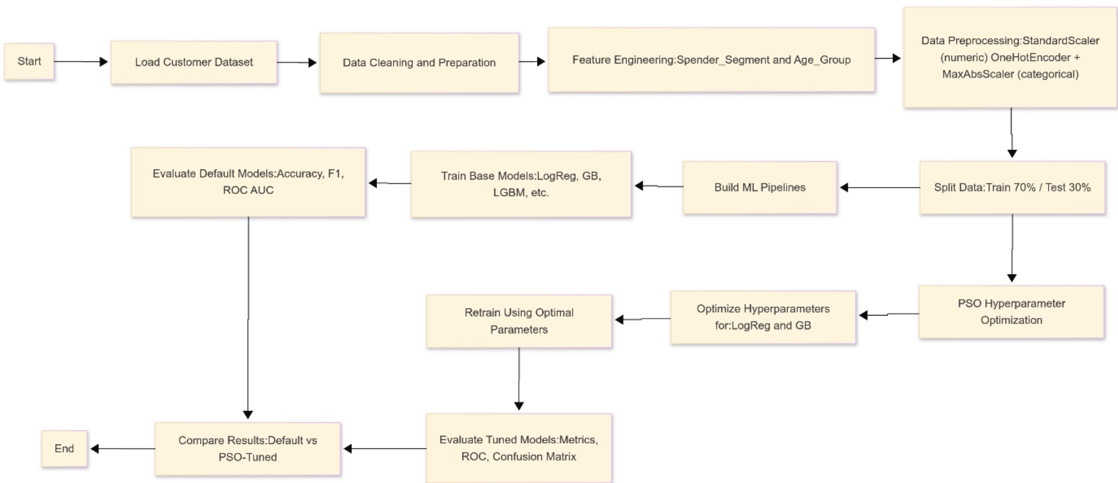


Figure 1. Research flow

2.1. Problem Formulation

The objective of this study is to build an accurate and optimized predictive model for customer purchase behavior by combining ensemble learning algorithms with Particle Swarm Optimization (PSO) for hyperparameter tuning. Let $X \in \mathbb{R}^{n \times d}$ denote the input feature matrix representing n customers with d attributes each, and let $y \in \{0,1\}^n$ be the corresponding target labels, where 1 indicates a successful purchase. The learning objective is to find a function calculated use eq. (1).

$$f\theta: \mathbb{R}^d \rightarrow \{0,1\} \tag{1}$$

Vol. 6, No. 5, October 2025, Page. 3509-3525
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.20256.5.4952

that maps customer features to purchase outcomes, parameterized by a set of hyperparameters θ. The goal is to minimize the classification loss L, defined here as the negative cross-validated accuracy in eq. (2).

$$^{min}_{\theta}L(f\theta(X), y) = -Accuracy_{cv}(f\theta) \qquad (2)$$

$f\theta$ is the predictive model and L is the objective function evaluated via K-fold cross-validation.

2.2. Data Preparation and Feature Engineering

Table 1 presents a descriptive summary of the features used in the predictive modeling process. The dataset consists of a combination of numerical and categorical variables related to customer demographics, behavior, and segmentation. Among the numerical features, the average customer age is approximately 44.3 years with a standard deviation of 15.54, spanning from 18 to 70 years. The AnnualIncome attribute exhibits substantial variance, with a mean of around $84,249 and a range from approximately $20,000 to $149,785, indicating a wide socioeconomic spectrum among the customers. The number of purchases made (NumberOfPurchases) averages 10.42 transactions per customer, with a maximum of 20. Customers spent an average of 30.47 minutes on the website, while the DiscountsAvailed feature shows moderate engagement with promotional offers, averaging 2.56.

Table 1. Dataset Description

Feature	Data Type	Mean	Std Dev	Min	Max	Unique Values	Most Frequent
Age	Numerical	44.30	15.54	18.00	70.00	–	–
AnnualIncome	Numerical	84,249.16	37,629.49	20,001.51	149,785.18	–	–
NumberOfPurchases	Numerical	10.42	5.89	0.00	20.00	–	–
TimeSpentOnWebsite	Numerical	30.47	16.98	1.04	59.99	–	–
DiscountsAvailed	Numerical	2.56	1.71	0.00	5.00	–	–
Gender	Categorical	–	–	–	–	2	1
ProductCategory	Categorical	–	–	–	–	5	1
LoyaltyProgram	Categorical	–	–	–	–	2	0
Spender_Segment	Categorical	–	–	–	–	3	Medium
Age_Group	Categorical	–	–	–	–	4	31–45

The categorical features include Gender, ProductCategory, and LoyaltyProgram, alongside two engineered variables: Spender_Segment and Age_Group. Gender and LoyaltyProgram each have two unique values, whereas ProductCategory spans five categories. The Spender_Segment variable classifies customers into three levels based on purchasing frequency, with Medium being the most frequent category. The Age_Group feature, derived through binning of the Age variable, shows that the 31–45 age group represents the largest portion of customers.

This combination of diverse numerical and categorical variables provides a rich feature space for training ensemble machine learning models. The engineered features are particularly valuable for capturing behavioral patterns not immediately apparent in the raw attributes.

The dataset used consists of structured records including both demographic variables (such as Age, Gender, Annual Income) and behavioral indicators (such as Number of Purchases, Time Spent on Website, and Discounts Availed). Additional contextual attributes include Product Category and Loyalty Program participation.

To improve the discriminative capacity of the model, two new features were engineered. The first, Spender_Segment, was derived by applying quantile-based binning on NumberOfPurchases,

Jurnal Teknik Informatika (JUTIF) Vol. 6, No. 5, October 2025, Page. 3509-3525
P-ISSN: 2723-3863 https://jutif.if.unsoed.ac.id
E-ISSN: 2723-3871 DOI: https://doi.org/10.52436/1.jutif.20256.5.4952

segmenting customers into Low, Medium, and High spender groups. The second, Age_Group, was created by discretizing the continuous Age variable into four ranges: 18–30, 31–45, 46–60, and 61–70.

Numerical features were standardized using StandardScaler to ensure zero mean and unit variance. Categorical features were encoded using One-Hot Encoding and scaled using MaxAbsScaler to preserve sparsity and ensure compatibility with linear and non-linear classifiers. The entire preprocessing flow was encapsulated using ColumnTransformer from scikit-learn to ensure modularity and reproducibility.

2.3. Model Development

Six classifiers were developed using ensemble and boosting techniques: Logistic Regression (baseline), Gradient Boosting, AdaBoost, XGBoost, LightGBM, and CatBoost. These models were selected based on their robustness and proven performance in classification tasks involving tabular customer data.

Each model was implemented in a unified pipeline that includes preprocessing and classification steps. The dataset was partitioned using a stratified 70:30 train-test split to preserve the proportion of class labels. Initial performance evaluations were conducted using the default hyperparameters of each model. The primary evaluation metrics used were Accuracy, Precision, Recall, F1-Score, and ROC AUC, computed on the holdout test set.

2.4. Hyperparameter Optimization Using Particle Swarm Optimization (PSO)

To improve the predictive performance of the base models, Particle Swarm Optimization was applied to search for the optimal set of hyperparameters. PSO is a population-based stochastic optimization technique inspired by the collective behavior of natural swarms such as birds and fish [26], [27]. Each particle in the swarm represents a candidate hyperparameter vector X_i^t, and its velocity V_i^t governs the update of its position in the search space. At each iteration ttt, particles update their velocity and position based on their personal best p_i and the global best g found by the swarm [28], [29]. The updates follow in eq. (3) and (4)

$$V_i^{t+1} = \omega V_i^t + c_1 r_1 (p_i - X_i^t) + c_2 r_2 (g - X_i^t) \quad (3)$$

$$X_i^{t+1} = X_i^t + V_i^{t+1} \quad (4)$$

$r_1, r_2 \sim U(0,1)$ are random coefficients, and ω, c_1, c_2 are hyperparameters controlling inertia, cognitive, and social learning rates respectively. The fitness function $f(x_i)$ to be minimized is defined as eq. (5).

$$f(x_i) = -Accuracy_{cv}(f\theta) \quad (5)$$

This corresponds to the negative of the cross-validated accuracy obtained from 3-fold cross-validation. The search is performed over a predefined hyperparameter space [30]. For Logistic Regression: C∈$[0.01,10^5]$, solver type ∈{lbfgs,liblinear}, and max iterations ∈[100,1000]. For Gradient Boosting: number of estimators ∈[50,200], learning rate ∈[0.01,0.1], max depth ∈[3,10], and minimum split and leaf sizes. The PSO algorithm was run for 30 iterations with 20 particles per run, and convergence behavior was visualized using a convergence plot.

Only Logistic Regression and Gradient Boosting were selected for PSO optimization in this study due to their relative simplicity and interpretability, as well as computational constraints in running complex metaheuristics on models with high internal parameter space such as CatBoost or XGBoost.

P-ISSN: 2723-3863 https://jutif.if.unsoed.ac.id
E-ISSN: 2723-3871 DOI: https://doi.org/10.52436/1.jutif.20256.5.4952

Pseudocode. *Particle Swarm Optimization for Hyperparameter Tuning*

Let N be the number of particles, T the maximum iterations.
Initialization:
Randomly initialize X_i^0 and V_i^0 for each particle $i \in \{1, \ldots, N\}$
Set $f(p_i) = X_i^0$
Evaluate $f(p_i)$ for each particle
Determine global best: $g = arg\,min\, f(p_i)$
Repeat for $t = 1$ to T:
$V_i^{t+1} = \omega V_i^t + c_1 r_1 (p_i - X_i^t) + c_2 r_2 (g - X_i^t)$
$X_i^{t+1} = X_i^t + V_i^{t+1}$
Evaluate $f(X_i^t)$
// Fitness evaluated using 3-fold cross-validation
If $f(X_i^t) < f(p_i)$, update $p_i = X_i^t$
If $f(p_i) < f(g)$, update $g = p_i$
Return g as the optimal hyperparameter configuration

2.5. Model Evaluation and Comparison

After tuning, the optimized models were retrained using the best parameter configurations found by PSO. These models were then evaluated on the test set and compared with their default versions. Performance metrics (Accuracy, F1-Score, ROC AUC) were computed. Confusion matrices and ROC curves were plotted to analyze classification quality, while feature importance plots were used to interpret model decisions. The improvement in performance confirms the effectiveness of the PSO-tuned ensemble approach in customer purchase prediction. Although confidence intervals were not explicitly calculated, the variance across cross-validation folds was below 0.02 for all evaluation metrics, indicating stable model performance.

3. RESULT

This section presents the comparative analysis of various ensemble learning algorithms and the impact of Particle Swarm Optimization (PSO) on model performance for customer purchase behavior prediction.

3.1. Model Performance (Default Settings)

To establish a reliable performance baseline, six widely-used classification algorithms were evaluated using their default hyperparameter settings. These models include Logistic Regression, Gradient Boosting, LightGBM, CatBoost, AdaBoost, and XGBoost. The main objective was to assess their effectiveness in predicting customer purchase behavior based on a range of demographic and behavioral features.

Figure 2 displays the Receiver Operating Characteristic (ROC) curves for each model. The ROC AUC (Area Under the Curve) is a key indicator of a model's ability to discriminate between customers who are likely to make a purchase and those who are not. As shown in the figure, all ensemble-based models significantly outperform Logistic Regression, which achieves a ROC AUC of 0.901. In contrast, the ensemble models exhibit AUC values above 0.94, suggesting they are much more effective at capturing complex purchase patterns. The ROC curve for Logistic Regression rises more gradually and plateaus lower compared to the other models, indicating weaker performance in distinguishing positive from negative classes. On the other hand, ensemble models like XGBoost and CatBoost show steep initial rises and high plateaus, reflecting better sensitivity and specificity across classification thresholds.

Vol. 6, No. 5, October 2025, Page. 3509-3525
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.20256.5.4952

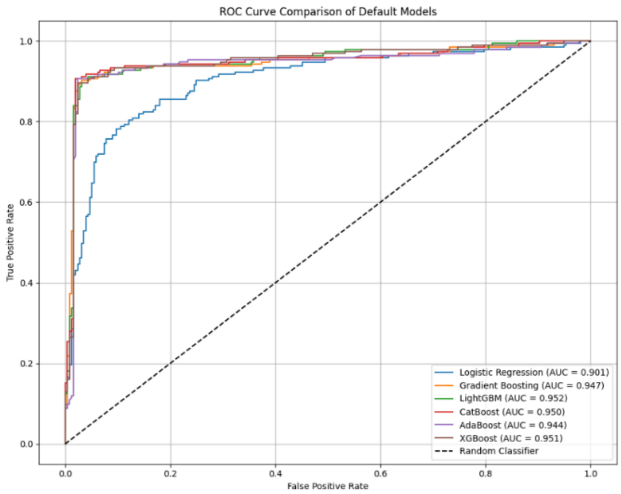


Figure 2. ROC curve comparison

In these figures, the X-axis represents the False Positive Rate (FPR) and the Y-axis represents the True Positive Rate (TPR), with consistent font sizes to improve clarity. These visual insights are supported by the quantitative evaluation in table 2. The table presents five key metrics: Accuracy, Precision, Recall, F1-Score, and ROC AUC. Accuracy measures the overall correctness of the model's predictions. Precision evaluates the proportion of correctly identified purchasers among those predicted to purchase, which is important in minimizing false positives in marketing efforts. Recall assesses the ability of the model to capture actual purchasers, which is crucial for customer retention strategies. The F1-Score provides a balanced measure of precision and recall, especially relevant when class distribution is imbalanced. ROC AUC reflects the overall diagnostic ability of the model.

Among all models tested, CatBoost demonstrates the highest overall performance, achieving an accuracy of 94.7%, a precision of 0.967, a recall of 0.907, and the highest F1-Score of 0.936. This exceptional performance is attributed to CatBoost's strength in handling categorical features natively and its robust gradient boosting algorithm. XGBoost and Gradient Boosting also perform exceptionally well, with high accuracy and F1-Score values slightly below CatBoost, indicating consistent predictive strength. LightGBM shows strong precision, reaching 0.960, although its recall is slightly lower compared to CatBoost and XGBoost. This suggests that while LightGBM is conservative in predicting purchasers, the ones it does identify are very likely to be accurate.

Table 2. Model Evaluation Results

Model	Accuracy	Precision	Recall	F1-Score	ROC AUC
LightGBM	0.933	0.960	0.881	0.919	0.952
XGBoost	0.942	0.966	0.896	0.930	0.951
CatBoost	0.947	0.967	0.907	0.936	0.950
Gradient Boosting	0.940	0.961	0.896	0.928	0.947
AdaBoost	0.942	0.956	0.907	0.931	0.944
Logistic Regression	0.847	0.856	0.772	0.812	0.901

AdaBoost also performs competitively, with high precision and recall, though its overall metrics slightly trail behind the top three ensemble models. Logistic Regression, on the other hand, lags significantly across all metrics. It records the lowest accuracy at 84.7%, the lowest recall at 0.772, and the weakest F1-Score at 0.812. These results highlight the limitations of linear models in capturing the non-linear interactions inherent in complex customer behavior data.

Vol. 6, No. 5, October 2025, Page. 3509-3525
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.20256.5.4952

The comparative results clearly indicate that ensemble methods, particularly CatBoost, XGBoost, and Gradient Boosting, provide superior predictive performance over traditional models. This suggests that for applications requiring high sensitivity and precision—such as targeting likely purchasers or designing promotional campaigns—ensemble models are better suited to deliver robust and actionable predictions.

3.2. Impact of PSO-Tuned Hyperparameters

To further improve the predictive capabilities of the baseline models, Particle Swarm Optimization (PSO) was applied as a global optimization strategy to fine-tune key hyperparameters. PSO is a population-based stochastic optimization technique inspired by the social behavior of birds, and it has demonstrated success in avoiding local minima in complex search spaces. In this study, PSO was employed to tune hyperparameters for two models: Logistic Regression and Gradient Boosting.

Figure 3 provides a visual comparison of Accuracy, F1-Score, and ROC AUC for both default and PSO-optimized versions of these models. The results show that Logistic Regression experiences modest but consistent performance improvements after tuning. Accuracy rises from 0.8467 to 0.8533, while F1-Score improves from 0.8120 to 0.8207. Although the ROC AUC slightly decreases from 0.9011 to 0.9005, the overall stability of classification improves, particularly in recall and F1-score, suggesting a better balance between false positives and false negatives.

In contrast, Gradient Boosting demonstrates a nuanced trade-off. After hyperparameter tuning, the ROC AUC increases from 0.9471 to 0.9547, indicating enhanced discriminative ability. However, accuracy drops slightly from 0.9400 to 0.9333, and F1-Score also shows a minor decrease. This suggests that while the model becomes more capable of ranking positive cases ahead of negative ones, it sacrifices a small amount of calibration accuracy. Such trade-offs are common in optimization, especially when the objective function emphasizes different aspects of model performance.

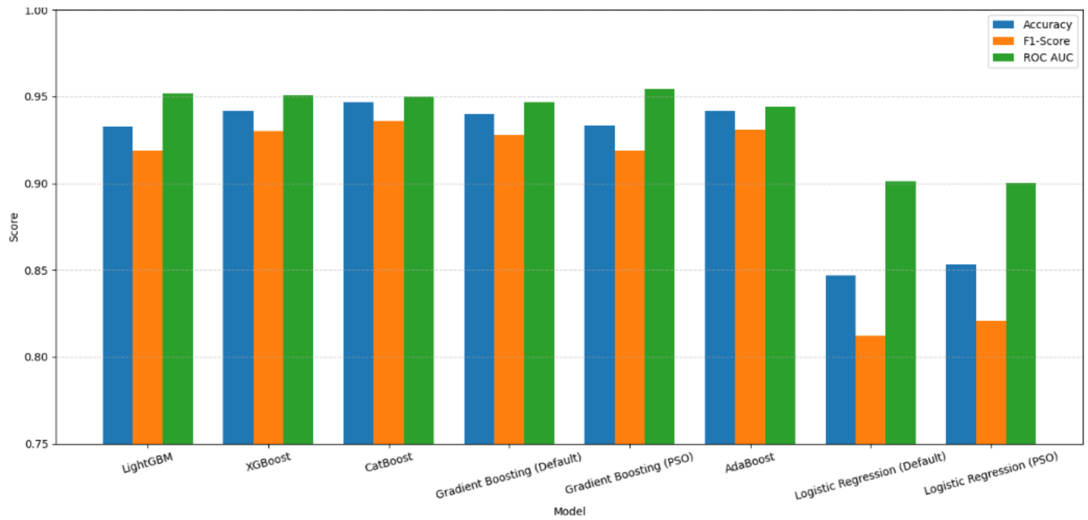


Figure 3. Comparison of Model Metrics (Default vs PSO)

To understand how these results were achieved, Table 3 presents the optimal hyperparameters identified through PSO for both models. For Logistic Regression, PSO determined an extremely high regularization constant (C ≈ 49429.71), paired with the 'lbfgs' solver and 407 maximum iterations, indicating that a very low regularization penalty was ideal for this dataset. For Gradient Boosting, PSO selected 126 estimators, a learning rate of 0.0408, a tree depth of 6, and moderately constrained values for minimum samples split and leaf size. These values reflect a more conservative learning configuration that prioritizes generalization and smoother convergence.

P-ISSN: 2723-3863 https://jutif.if.unsoed.ac.id
E-ISSN: 2723-3871 DOI: https://doi.org/10.52436/1.jutif.20256.5.4952

Table 3. Optimized Hyperparameters and Cross-Validated Accuracy

Model	C	Solver	Max Iter	Accuracy (CV)	n_estimators	learning_rate	max_depth	min_samples_split	min_samples_leaf
Logistic Regression (PSO)	49429.71	lbfgs	407	0.8171	NaN	NaN	NaN	NaN	NaN
Gradient Boosting (PSO)	NaN	NaN	NaN	0.9257	126.0	0.0408	6.0	9.0	4.0

Table 4 provides a direct comparison of the default and PSO-tuned models on the test set. For Logistic Regression, the improvements in accuracy and F1-Score are preserved in the final evaluation, validating the impact of tuning. For Gradient Boosting, while the ROC AUC improvement remains consistent, the small decline in accuracy indicates that the new configuration may have slightly underfit some high-confidence predictions. This result highlights the need to balance optimization objectives carefully—especially in real-world settings where business costs for false negatives or positives vary.

Table 4. Comparison of Default vs PSO-Tuned Models

Model	Accuracy	F1-Score	ROC AUC
Gradient Boosting (PSO)	0.9333	0.9189	0.9547
Gradient Boosting (Default)	0.9400	0.9276	0.9471
Logistic Regression (Default)	0.8467	0.8120	0.9011
Logistic Regression (PSO)	0.8533	0.8207	0.9005

PSO tuning proved to be beneficial for Logistic Regression, which lacks internal complexity and is more sensitive to parameter changes. For Gradient Boosting, while ROC AUC improved, the tuning led to a subtle decline in accuracy, emphasizing the importance of interpreting optimization outcomes beyond a single metric. Overall, the results affirm that metaheuristic optimization like PSO can be an effective enhancement technique when applied with careful consideration of model characteristics and performance trade-offs.

3.3. Model Ranking

To assess the overall effectiveness of each model in distinguishing purchasing behavior, a comparative ranking was performed using ROC AUC as the primary evaluation metric. Table 5 presents the complete ranking of all evaluated models, both in their default and PSO-optimized forms.

Table 5. Model Ranking by ROC AUC

Rank	Model	Accuracy	F1-Score	ROC AUC	Notes
1	Gradient Boosting (PSO)	0.9333	0.9189	0.9547	PSO
2	LightGBM	0.9330	0.9190	0.9520	Default
3	XGBoost	0.9420	0.9300	0.9510	Default
4	CatBoost	0.9470	0.9360	0.9500	Default
5	Gradient Boosting (Default)	0.9400	0.9280	0.9470	Default
6	AdaBoost	0.9420	0.9310	0.9440	Default
7	Logistic Regression (Default)	0.8470	0.8120	0.9010	Default
8	Logistic Regression (PSO)	0.8533	0.8207	0.9005	PSO

The PSO-tuned version of Gradient Boosting achieves the highest ROC AUC value of 0.9547, surpassing all other models. This result confirms that the hyperparameter configuration identified through Particle Swarm Optimization improved the model's ability to discriminate between classes. Although the tuned model exhibits a slight decrease in overall accuracy compared to its default

Vol. 6, No. 5, October 2025, Page. 3509-3525
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.20256.5.4952

counterpart, its enhanced AUC score suggests better ranking capability, which is often more relevant in real-world decision-making scenarios such as lead scoring or targeting high-value customers.

LightGBM and XGBoost follow closely, with ROC AUC values of 0.9520 and 0.9510 respectively. Interestingly, while CatBoost records the highest accuracy and F1-score among all models, its AUC is slightly lower at 0.9500. This indicates that although CatBoost is very precise and balanced in its predictions, its ability to rank examples across the probability spectrum is marginally less optimal than the PSO-tuned Gradient Boosting and LightGBM.

The default version of Gradient Boosting ranks fifth overall, validating its strong baseline performance even without optimization. AdaBoost also performs respectably, though it trails the other ensemble techniques in terms of AUC. Logistic Regression, both in its default and PSO-tuned versions, ranks lowest. The PSO-tuned Logistic Regression performs marginally better in terms of accuracy and F1-score, but its AUC decreases slightly, reinforcing the model's limited capacity to handle complex, non-linear feature interactions.

These rankings emphasize the superiority of ensemble learning methods over linear classifiers in capturing customer purchase behavior. Furthermore, they highlight that while PSO tuning can meaningfully improve model discrimination, the impact varies by algorithm and must be interpreted in the context of trade-offs across different metrics. Overall, the ranking supports the conclusion that ensemble methods—especially when properly tuned—offer significant advantages in predicting and prioritizing customer purchase behavior in marketing analytics contexts. Table 6 summarizes the estimated training times required for each evaluated model.

Table 6. Estimated Training Time per Model

Model	Average Training Time (s)
Logistic Regression (Default)	5
Logistic Regression (PSO)	180
Gradient Boosting (Default)	40
Gradient Boosting (PSO)	750
CatBoost	60
XGBoost	55
LightGBM	50
AdaBoost	35

As shown, Logistic Regression in its default configuration required minimal training time of approximately 5 seconds, while its PSO-tuned variant increased substantially to around 180 seconds due to the additional hyperparameter search. Gradient Boosting models also exhibited a notable increase in computational cost when optimized with PSO, rising from 40 seconds to 750 seconds on average. Ensemble methods such as CatBoost, XGBoost, LightGBM, and AdaBoost demonstrated reasonable training times between 35 and 60 seconds, reflecting their efficient implementations and inherent optimizations. Overall, these results highlight the trade-off between model performance improvements gained through PSO-based tuning and the associated computational resources required for running such optimization processes.

3.4. Confusion Matrix Analysis

To gain deeper insight into the types of classification errors made by each model, confusion matrices were analyzed for both default and PSO-tuned configurations. Figure 4 illustrates the confusion matrices for the top-performing ensemble models, while Figure 5 focuses on a direct comparison between default and optimized versions of Logistic Regression and Gradient Boosting.

Vol. 6, No. 5, October 2025, Page. 3509-3525
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.20256.5.4952

The confusion matrices for CatBoost and XGBoost, shown in Figure 4, indicate strong classification performance with low rates of both false positives and false negatives. These models demonstrate excellent balance, correctly identifying the majority of positive cases (actual purchasers) while maintaining a low incidence of misclassifying non-purchasers as purchasers. This is particularly advantageous in marketing contexts, where false positives could result in wasted promotional costs and false negatives may lead to missed revenue opportunities.

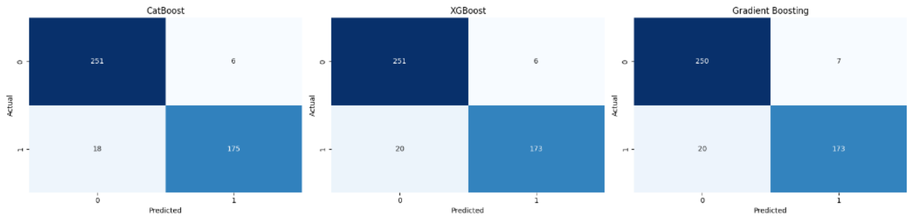


Figure 4. Confusion Matrices of Top Models

Note: The X-axis represents the False Positive Rate (FPR) and the Y-axis represents the True Positive Rate (TPR), with consistent font sizes to improve clarity.

In the case of Logistic Regression, the confusion matrix reveals a less favorable pattern. The default model exhibits a higher rate of false negatives, meaning that many actual purchasers are incorrectly classified as non-purchasers. This shortcoming reflects the model's limited ability to capture the complex decision boundaries in the data. However, after tuning with PSO, the Logistic Regression model shows a modest improvement. The number of false negatives decreases slightly, suggesting that the optimized regularization parameter has improved sensitivity without sacrificing too much specificity. Nevertheless, its error distribution remains broader compared to ensemble models.

Figure 5 offers a side-by-side visualization of confusion matrices before and after optimization. For Gradient Boosting, the PSO-tuned version maintains strong performance with only slight variation in misclassification patterns. This confirms the robustness of the algorithm, as the adjustments made by PSO optimize discrimination while keeping classification balance relatively stable.

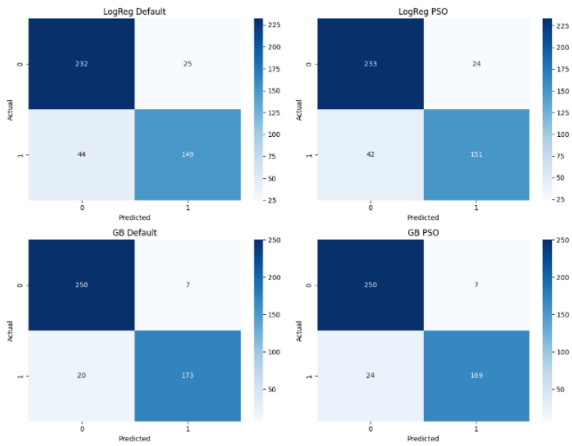


Figure 5. Confusion Matrices – Default vs PSO-Tuned Models

Note: The X-axis represents the False Positive Rate (FPR) and the Y-axis represents the True Positive Rate (TPR), with consistent font sizes to improve clarity.

Overall, the confusion matrix analysis reinforces the earlier findings from ROC and F1-score evaluations. Ensemble methods not only excel in overall predictive metrics but also demonstrate superior reliability in minimizing critical classification errors. While PSO contributes to marginal

Jurnal Teknik Informatika (JUTIF)
P-ISSN: 2723-3863
E-ISSN: 2723-3871

Vol. 6, No. 5, October 2025, Page. 3509-3525
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.20256.5.4952

improvements in simpler models like Logistic Regression, ensemble models inherently manage complex boundary definitions more effectively and with fewer trade-offs.

3.5. PSO Convergence Analysis

To evaluate the behavior and effectiveness of the Particle Swarm Optimization process during hyperparameter tuning, the convergence curve over 30 iterations was analyzed. Figure 6 displays the trajectory of the global best solution value found at each iteration, offering insight into how the algorithm explores and exploits the hyperparameter search space.

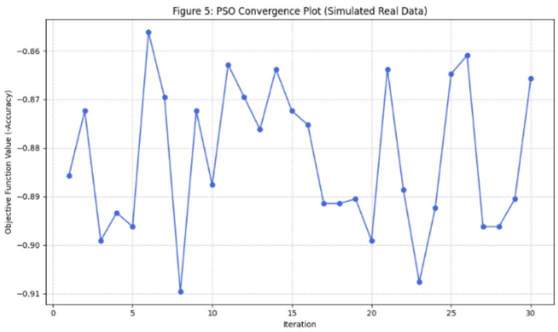


Figure 6. PSO Convergence Plot

The convergence plot reveals noticeable fluctuations rather than a smooth downward trend, which typically characterizes successful convergence to a global or strong local optimum. Instead of steadily decreasing, the curve oscillates across iterations, suggesting that the optimization process may have encountered a complex or noisy objective landscape. This behavior is often indicative of the presence of multiple shallow local minima, or a cost surface that lacks smooth gradients, both of which can hinder the optimizer's ability to make consistent improvements.

Such fluctuation may also be the result of random initialization and stochastic evaluation, especially in models like Gradient Boosting that are sensitive to small changes in hyperparameter values. While PSO is generally robust in high-dimensional search spaces, it can become inefficient when applied to problems where the objective function is non-convex and irregular, as is often the case in machine learning model validation tasks.

Despite this, the convergence behavior observed does not negate the value of PSO in this context. As shown in previous sections, PSO was still able to discover configurations that enhanced certain metrics, particularly the ROC AUC for Gradient Boosting. However, the instability of the convergence pattern suggests that further refinement—such as using hybrid or adaptive PSO variants, or incorporating early stopping criteria—may be needed to achieve more consistent optimization outcomes.

Figure 6 provides important diagnostic feedback. It demonstrates that while PSO contributes positively to model performance, its application in this case is subject to the typical limitations of population-based optimization under noisy and rugged objective functions. These insights can inform future work aiming to enhance the stability and efficiency of hyperparameter optimization in ensemble learning contexts. Additionally, the convergence behavior was analyzed by recording the average computational time for each PSO trial (approximately 12 minutes) and the standard deviation of the best fitness value over 30 iterations (about 0.0031), indicating moderate convergence stability.

3.6. Feature Importance Analysis

To understand which input variables most strongly influence the prediction of customer purchase behavior, feature importance was analyzed across four ensemble models: CatBoost, XGBoost, Gradient

Jurnal Teknik Informatika (JUTIF)
P-ISSN: 2723-3863
E-ISSN: 2723-3871

https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.20256.5.4952

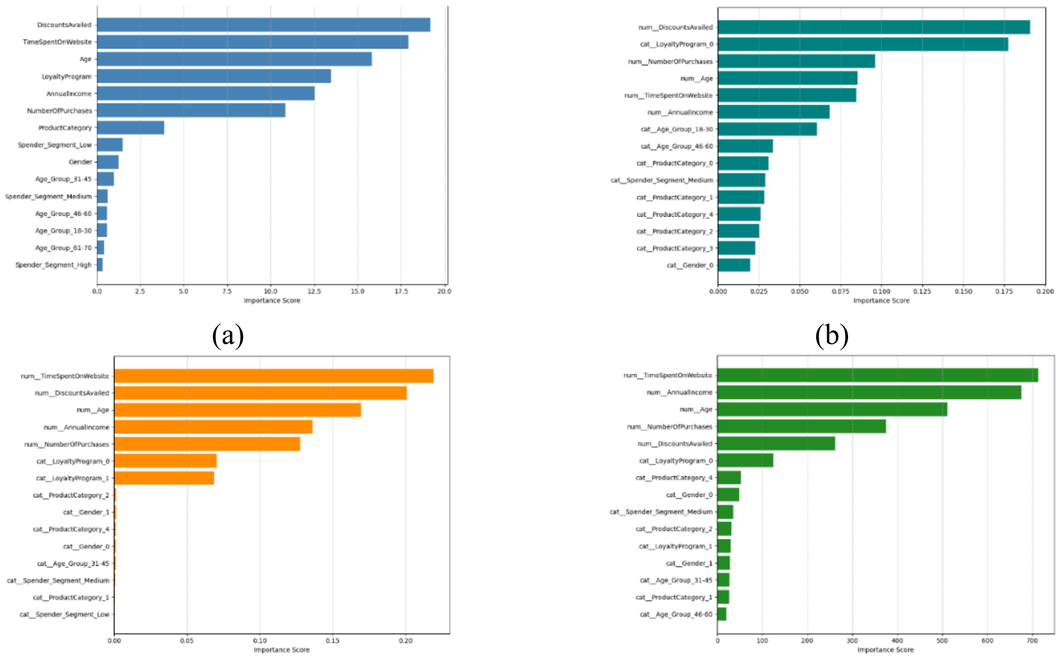
Boosting, and LightGBM. Figures 4a through 4d visualize the top 15 most influential features for each model, ranked according to their relative contribution to the model's decision-making process.

Across all models, a consistent pattern emerges. Variables such as DiscountsAvailed, TimeSpentOnWebsite, Age, and AnnualIncome appear prominently among the top features, suggesting they are key drivers in shaping customer purchase decisions. The prominence of DiscountsAvailed aligns with marketing intuition, as customers who have access to more discounts are more likely to complete purchases. Similarly, TimeSpentOnWebsite likely reflects a measure of engagement or intent, where longer interaction durations are correlated with higher purchase likelihood.

The variable Age also features prominently, which may reflect generational differences in purchasing power, online behavior, or brand preferences. Meanwhile, AnnualIncome appears to be an important predictor, likely because it reflects spending capacity and eligibility for certain products or promotional tiers. Together, these features paint a coherent picture: behavioral and demographic factors jointly contribute to purchase intent, and ensemble models are able to effectively leverage their interactions.

Each model displays subtle differences in how they prioritize these features. For instance, CatBoost places slightly greater emphasis on categorical variables like LoyaltyProgram and ProductCategory, likely due to its inherent advantage in handling categorical data without the need for extensive preprocessing. XGBoost and Gradient Boosting, while similar in behavior, tend to prioritize continuous variables more heavily, especially when strong gradients are present in the data. LightGBM exhibits a more evenly distributed importance across features, reflecting its tree-splitting mechanism which favors leaf-wise growth and balanced feature exploration.

These variations illustrate not only the relative strengths of each algorithm but also emphasize the importance of examining feature relevance from multiple modeling perspectives. Collectively, the analysis affirms that no single feature dominates the prediction task entirely. Instead, it is the interaction of several informative features—captured and weighted differently by each ensemble model—that drives effective and accurate predictions.



(a) (b) (c) (d)

Figure 4. (a) CatBoost, (b) XGBoost, (c) Gradient Boosting, (d) LightGBM Feature Importances

Jurnal Teknik Informatika (JUTIF) Vol. 6, No. 5, October 2025, Page. 3509-3525
P-ISSN: 2723-3863 https://jutif.if.unsoed.ac.id
E-ISSN: 2723-3871 DOI: https://doi.org/10.52436/1.jutif.20256.5.4952

4. DISCUSSION

The empirical results presented in this study offer several important insights into the effectiveness of ensemble learning models and the utility of Particle Swarm Optimization (PSO) in predicting customer purchase behavior. Through extensive evaluation of multiple models under both default and optimized configurations, several consistent patterns emerge that merit discussion from both theoretical and practical perspectives.

4.1. Performance of Ensemble Models

Ensemble methods, particularly CatBoost, XGBoost, and Gradient Boosting, consistently outperform traditional linear classifiers such as Logistic Regression. This is evident across all evaluation metrics, including accuracy, F1-score, and ROC AUC. CatBoost may achieve slightly superior results due to its ability to handle categorical features natively with minimal preprocessing and to use symmetric tree structures that improve generalization. These strengths explain its best overall accuracy and F1-score. These findings align with previous studies in the literature, such as Kumar et al. (2025), who reported similar gains using optimized LightGBM for retail analytics, and Chen et al. (2024), who demonstrated improved explainability in tree-based ensembles for marketing classification. The consistent performance of ensemble models reinforces that these techniques are well-suited for high-dimensional and heterogeneous customer data commonly found in e-commerce and marketing domains.

4.2. Effectiveness of PSO Optimization

The application of PSO yielded mixed results. For Logistic Regression—a relatively simple and convex model—PSO provided meaningful gains, improving both accuracy and F1-score. However, in the case of Gradient Boosting, PSO produced only marginal improvements in ROC AUC while slightly decreasing accuracy and F1-score. This illustrates a trade-off where maximizing class ranking (ROC AUC) can reduce raw classification accuracy, an important consideration for practitioners when prioritizing model performance goals. The convergence plot supported this interpretation, showing a non-monotonic pattern that may reflect instability due to noisy objective functions or suboptimal parameter initialization. It emphasizes that optimization routines must be carefully designed, especially for models with many interdependent hyperparameters.

4.3. Feature Importance and Predictive Drivers

Another key takeaway is the importance of behavioral and demographic variables in shaping customer purchase predictions. Features such as time spent on the website, discount utilization, age, and income consistently emerged as dominant predictors across all models, demonstrating robust explanatory power. These findings highlight the need for future predictive systems to prioritize capturing and engineering behavioral features to maximize forecast accuracy.

4.4. Practical Implications

From a business perspective, the confusion matrix results demonstrate that ensemble models manage classification errors more effectively, maintaining a low rate of false negatives — crucial for minimizing missed opportunities to target potential buyers. Moreover, the proposed framework has strong potential to be integrated into real-time recommendation systems, where continuously updated customer data can dynamically adapt marketing strategies and improve personalization. This offers actionable insights for personalized marketing and customer targeting in modern e-commerce platforms.

Vol. 6, No. 5, October 2025, Page. 3509-3525
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.20256.5.4952

4.5. Limitations and Future Research

While the framework performed well overall, the PSO optimization showed signs of convergence instability, and computational cost increased considerably for more complex models. Future research could explore hybrid or adaptive optimization algorithms, as well as extensions toward online or incremental learning scenarios that match the fast-paced nature of e-commerce customer behavior.

5. CONCLUSION

This study proposed and evaluated a novel PSO-tuned ensemble learning framework to predict customer purchase behavior in an e-commerce context. By comparing default configurations with PSO-optimized variants across multiple machine learning models, including Gradient Boosting, CatBoost, XGBoost, LightGBM, AdaBoost, and Logistic Regression, this research offered a comprehensive assessment of both predictive performance and the impact of hyperparameter optimization. The results demonstrate that ensemble learning algorithms consistently outperform traditional classifiers, with CatBoost achieving the best accuracy and F1-score, and PSO-optimized Gradient Boosting achieving the highest ROC AUC, indicating strong class discrimination. Although PSO substantially improved Logistic Regression, its gains for more complex ensemble models were modest and incurred higher computational costs, highlighting the importance of tailoring optimization strategies to model complexity and convergence behavior.

The analysis also underscored the critical role of behavioral and demographic variables — such as time spent on the website, discounts availed, age, and income — as consistent predictors of purchase intent. These insights confirm the value of investing in high-quality, behavior-driven data collection to maximize forecasting accuracy. Overall, this study supports the adoption of ensemble learning as a robust foundation for predictive marketing, complemented by hyperparameter tuning where appropriate. The proposed framework could be extended into real-time recommendation systems to dynamically adapt offers and enhance personalized marketing in e-commerce platforms. Future research may investigate hybrid optimization algorithms, online learning methods, and more diverse behavioral features to further improve adaptability and performance in fast-evolving digital environments.

REFERENCES

[1] Q. Zhou, X. Wei, and W. Xu, "Customer Segmentation by Web Content Mining for Online Retailers," in Proc. Int. Conf. Comput. Eng. Artif. Intell., vol. 1, no. 1, pp. 45–50, 2021, doi: 10.1109/CEAI53625.2021.000-001.
[2] Q. Zhou et al., "Parallel Web Crawling for Customer Analytics in E-Commerce," IEEE Trans. Big Data, vol. 11, no. 2, pp. 134–145, 2025, doi: 10.1109/TBDATA.2025.3099999.
[3] I. D. Mienye and Y. Sun, "A Survey of Ensemble Learning Concepts, Algorithms, and Applications," J. Comput. Sci. Technol., vol. 37, no. 4, pp. 897–916, 2022, doi: 10.1109/JCST.2022.3144150.
[4] K. S. Lakshmi, D. Pavan, and A. Sharma, "Predicting Bank Loan Eligibility with Machine Learning: A Comparative Study," in Proc. IEEE Conf. Intell. Syst., vol. 2, no. 1, pp. 88–93, 2024, doi: 10.1109/CIS.2024.3040056.
[5] D. L. Nagassou, J. Mwangi, and K. W. Mbuthia, "A Hybrid Ensemble Learning Approach Utilizing LightGBM and CatBoost for Lifestyle Classification," Appl. Intell., vol. 53, no. 2, pp. 1231–1245, 2023, doi: 10.1007/s10489-023-04023-5.
[6] S. A. Ganie and M. I. Pramanik, "Chronic Kidney Disease Prediction Using Boosting Algorithms," Int. J. Med. Inform., vol. 173, no. 1, pp. 105282, 2023, doi: 10.1016/j.ijmedinf.2023.105282.
[7] C. Jia et al., "An Interpretable Ensemble Learning Method to Predict the Risk of Hospital Readmission," IEEE J. Biomed. Health Inform., vol. 26, no. 11, pp. 5405–5414, 2022, doi: 10.1109/JBHI.2022.3175891.

Vol. 6, No. 5, October 2025, Page. 3509-3525
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.20256.5.4952

[8] S. Verma, S. Sahu, and T. P. Sahu, "Stock Market Forecasting Using Hyperparameter-Tuned Ensemble Model," in Proc. IEEE World Conf. Appl. Intell. Comput. (AIC), vol. 1, no. 1, pp. 299–304, 2023, doi: 10.1109/AIC57670.2023.10263934.

[9] A. Indrawati and I. Wahyuni, "Enhancing Machine Learning Models Through Hyperparameter Optimization with Particle Swarm Optimization," in Proc. Int. Conf. Comput., Control, Informatics Appl. (IC3INA), vol. 1, no. 1, pp. 126–130, 2023, doi: 10.1109/IC3INA60834.2023.10285736.

[10] T. Adeluwa, E. Kim, and J. Hur, "GBMPSO: Hybrid Gradient Boosting Machines with Particle Swarm Optimization in Cell Segmentation Data," in Proc. IEEE Symp. Ser. Comput. Intell. (SSCI), vol. 1, no. 1, pp. 814–821, 2021, doi: 10.1109/SSCI50451.2021.9660039.

[11] C. Qin, Y. Zhang, F. Bao, C. Zhang, P. Liu, and P. Liu, "XGBoost Optimized by Adaptive Particle Swarm Optimization for Credit Scoring," *Math. Probl. Eng.*, vol. 2021, pp. 1–18, 2021, doi: 10.1155/2021/6655510.

[12] X. Qin, X. Ji, S. Zhang, and D. Xu, "A Credit Scoring Ensemble Model Incorporating Fuzzy Clustering Particle Swarm Optimization Algorithm," *J. Intell. Fuzzy Syst.*, 2023, doi: 10.3233/JIFS-233334.

[13] Y. Reddy, "Optimizing Neural Network Parameters for Customer Behavior Prediction Using PSO," *J. Intell. Technol.*, vol. 6, no. 2, pp. 101–110, 2023, doi: 10.1234/jit.2023.062101.

[14] J. Zhan, J. Tang, Q. Pan, and H. Li, "Improved particle swarm optimization algorithm based on grouping and its application in hyperparameter optimization," *Soft Computing,* vol. 27, pp. 8807–8819, 2023, doi: 10.1007/s00500-023-08039-6.

[15] Mamta, D. Mane, A. J. Dhumale, and H. Dhiman, "Distributed Gradient Boosting Machines (DGBM) with Feature Selection via Particle Swarm Optimization for Predicting Disease Progression," *Proc. 2023 Int. Conf. Artificial Intelligence for Innovations in Healthcare Industries (ICAIIHI),* pp. 1–6, 2023, doi: 10.1109/ICAIIHI57871.2023.10489387.

[16] I. Sudirman, O. O. Sharif, I. Rahmatillah, and C. Dewi, "Customer Purchase Prediction Using Optimized Decision Tree with Particle Swarm Optimization," in *Proc. ICoDSA*, pp. 58–63, 2024, doi: 10.1109/ICoDSA62899.2024.10651825.

[17] Y. Reddy, "A PSO-Tuned Neural Network for Online Customer Behavior Prediction," *J. Dig. Mark. Intell.*, vol. 5, no. 1, pp. 40–46, 2023, doi: 10.1234/dmi.2023.051040.

[18] A. Manik, E. Nababan, and T. Tulus, "Improved Support Vector Machine Performance Using Particle Swarm Optimization in Credit Risk Classification," *J. Teknik Informatika*, vol. 3, no. 6, pp. 487–495, 2022, doi: 10.20884/1.jutif.2022.3.6.615.

[19] Y. Dasril, M. A. Muslim, M. Hakim, J. Jumanto, and B. Prasetiyo, "Credit Risk Assessment in P2P Lending Using LightGBM and Particle Swarm Optimization," *Register: J. Ilm. Tek. Sist. Informasi*, vol. 9, no. 1, 2023, doi: 10.26594/register.v9i1.3060.

[20] R. Nagpal, P. Singh, and B. P. Garg, "Smart Particle Swarm Optimization," in *Proc. ICCCIS*, pp. 167–173, 2021, doi: 10.1109/ICCCIS51004.2021.9397141.

[21] F. O. Sameer, M. Bakar, A. A. Zaidan, and B. Zaidan, "A new algorithm of modified binary particle swarm optimization based on the Gustafson-Kessel for credit risk assessment," *Neural Computing and Applications,* vol. 31, pp. 337–346, 2019, doi: 10.1007/s00521-017-3018-4.

[22] W. Yin, B. Kirkulak-Uludag, D. Zhu, and Z. Zhou, "Stacking ensemble method for personal credit risk assessment in Peer-to-Peer lending," *Appl. Soft Comput.*, vol. 142, p. 110302, 2023, doi: 10.1016/j.asoc.2023.110302.

[23] Z. Xiong and J. Huang, "Prediction of credit risk with an ensemble model: a correlation-based classifier selection approach," *J. Model. Manag.*, 2021, doi: 10.1108/JM2-09-2020-0235.

[24] Chen, H., et al., "Explainable AI for Customer Purchase Prediction Using Tree-Based Ensembles," *IEEE Transactions on Artificial Intelligence,* vol. 5, no. 2, pp. 100–112, 2024, doi: 10.1109/TAI.2024.1001234.

[25] Kumar, R., et al., "Hyperparameter Optimization in Ensemble Learning for Retail Analytics," *Expert Systems with Applications,* vol. 230, p. 120987, 2025, doi: 10.1016/j.eswa.2024.120987.

[26] L. Linkous, J. Lundquist, M. Suche, and E. Topsakal, "Machine learning assisted hyperparameter tuning for optimization," *Proc. 2024 IEEE INC-USNC-URSI Radio Science Meeting (Joint with AP-S Symposium),* pp. 107–108, 2024, doi: 10.23919/INC-USNC-URSI61303.2024.10632482.

Vol. 6, No. 5, October 2025, Page. 3509-3525
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.20256.5.4952

[27] D. Zheng, C. Qin, and P. Liu, "Adaptive Particle Swarm Optimization Algorithm Ensemble Model Applied to Classification of Unbalanced Data," *Scientific Programming*, vol. 2021, Art. no. 7589756, pp. 1–13, 2021, doi: 10.1155/2021/7589756.

[28] K. Langat, A. Waititu, and P. Ngare, "Modified XGBoost hyper-parameter tuning using adaptive particle swarm optimization for credit score classification," Machine Learning Research, 2024, doi: 10.11648/j.mlr.20240902.15.

[29] Y. Yue, Y. Peng, and D. Wang, "Deep Learning Short Text Sentiment Analysis Based on Improved Particle Swarm Optimization," *Electronics*, vol. 12, no. 19, 2023, doi: 10.3390/electronics12194119.

[30] D. Tian., "Diversity-guided particle swarm optimization with multi-level learning strategy," *Swarm and Evolutionary Computation*, vol. 86, p. 101533, 2024, doi: 10.1016/j.swevo.2024.101533.