e-ISSN 2549-7472
Edumatic: Jurnal Pendidikan Informatika
Volume 10, Issue 3, **540-549.**
https://e-journal.hamzanwadi.ac.id/index.php/edumatic
DOI: 10.29408/edumatic.v10i3.36418

Algorithm-Dependent Effects of PSO-Based Hyperparameter Optimization in Chronic Kidney Disease Classification

Muhammad Hamdi [1,*], Galih Hendro Martono [1], Dadang Priyanto [1]

[1] Universitas Bumigora, Indonesia
[*] Corresponding author: Muhammad Hamdi, Universitas Bumigora, Indonesia
✉ hamdi@hamzanwadi.ac.id

Copyright: © 2026 by the authors

Received: 27 July 2026 | Revised: 7 September2026 | Accepted: 5 November 2026 | Published: 27 November 2026

Abstract
Classifying Chronic Kidney Disease (CKD) requires accurate, timely predictions, and machine learning performance is highly sensitive to hyperparameter settings. This study investigates whether Particle Swarm Optimization (PSO) improves classification performance consistently across different algorithm families. Using a public Kaggle CKD dataset consisting of 2,304 records and 9 clinical attributes, seven classifiers Random Forest, Decision Tree, Logistic Regression, Support Vector Machine, K-Nearest Neighbor, Naïve Bayes, and Artificial Neural Network were evaluated before and after PSO-based hyperparameter optimization. We applied an identical preprocessing pipeline across three data-splitting scenarios (70:30, 80:20, and 90:10) and evaluated accuracy, precision, recall, F1-score, and ROC-AUC. The results demonstrate that PSO does not provide uniform improvements across classifiers. Random Forest, Decision Tree, KNN, and Naïve Bayes showed consistent performance improvements, whereas SVM performance declined across all scenarios. ANN changed the precision-recall trade-off without substantial overall improvement. Random Forest and Decision Tree achieved near-perfect performance after optimization, which should be interpreted cautiously because of the risk of overfitting. These findings indicate that PSO effectiveness depends on the algorithm, emphasizing the need for tailored optimization strategies rather than uniform application across machine learning models.

Keywords: algorithm-dependent effect; chronic kidney disease; hyperparameter optimization; machine learning classification; particle swarm optimization

To cite this article: Hamdi, M., Martono, G. H., & Priyanto, D. (2026). Algorithm-Dependent Effects of PSO-Based Hyperparameter Optimization in Chronic Kidney Disease Classification. *Edumatic: Jurnal Pendidikan Informatika*, *10*(3), 540–549. https://doi.org/10.29408/edumatic.v10i3.36418

INTRODUCTION

Machine learning model performance depends not only on the learning algorithm but also on the interaction among model structure, hyperparameter configuration, and the optimization strategy used to tune it (Bischl et al., 2023; Probst et al., 2019). Algorithms differ in how sensitive they are to hyperparameters: tree-based models depend on depth and the number of estimators, SVM on regularisation and kernel parameters, KNN on neighborhood size, and ANN on architecture and learning-related parameters. This matters most in clinical classification, where datasets are limited, overfitting is a risk, and errors have unequal consequences. High accuracy does not guarantee generalization, as Ganie et al. (2023) reported an AdaBoost model with 100% training accuracy but 98.47% test accuracy. Chronic kidney disease (CKD) is a suitable case because it can be classified from routine clinical and laboratory

Edumatic: Jurnal Pendidikan Informatika is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License

variables such as glomerular filtration rate (Stevens et al., 2024) and carries a substantial public health burden (Bikbov et al., 2020; Hidayangsih et al., 2023). The central question, however, is methodological rather than clinical: can a given hyperparameter optimization strategy improve performance consistently across algorithms with different structures while preserving a balance among evaluation metrics?

Published CKD studies show no consensus on the best algorithm: Islam et al. (2023) reported Random Forest (RF) as superior, Swain et al. (2023) reported SVM, Ganie et al. (2023) and Iftikhar et al. (2026) reported boosting or ensemble methods. Similar variation appears in other healthcare prediction tasks, where SVM led in stunting prediction (Febriyanti et al., 2025) but Logistic Regression slightly outperformed RF and XGBoost in thyroid cancer recurrence (Ais & Sanjaya, 2025). These inconsistencies likely reflect are than intrinsic algorithm capability alone. Datasets, preprocessing, validation strategies, and hyperparameter configurations also differ across studies (Uddin & Lu, 2024), and because these factors vary simultaneously, the independent contribution of hyperparameter optimization cannot be isolated from published results. This gap motivates controlled experiments in which baseline and optimized models are evaluated under identical conditions.

Hyperparameter optimization is therefore better treated as an experimental factor whose effect may differ across algorithms than as a procedure for maximizing accuracy. Available strategies range from manual tuning, grid search, and random search (Bergstra & Bengio, 2012) to Bayesian optimization and population-based metaheuristics such as genetic algorithms and particle swarm optimization (PSO) (Shami et al., 2022). The availability of these methods does not imply that every classifier benefits equally. Hyperparameter importance varies across algorithms (Probst et al., 2019), consistent with the no-free-lunch principle. The effect of optimization likely depends on the interaction among the search mechanism, the dimensionality and bounds of the search space, parameter sensitivity, and the classifier's decision structure. The same optimizer may thus yield large gains for one classifier, negligible change for another, or even degraded performance under a given objective. Whether optimization effects are consistent or algorithm-dependent is a distinct empirical question that reporting the best accuracy alone cannot answer.

We use PSO here as an experimental instrument, not because we presume it superior to other optimizers (Kennedy & Eberhart, 1995). Applying one common search procedure to all classifiers reduces variation attributable to different optimization methods, so that observed differences better reflect classifier characteristics. This underpins the design's validity. PSO has been applied in CKD-related studies with Decision Tree (Fitri & Baita, 2025), SVM (Purwaningsih & Nurelasari, 2024), ANN (Huang & Chou, 2019), Random Tree, and echo state networks (Anbazhagan & Rangaswamy, 2025). Reported gains include Decision Tree from 94.2% to 98.75%, SVM from 95.1% to 98.2%, and ANN from 98% to 99%. Each study, however, examined a single model under its own dataset and protocol, so it remains unclear whether the improvement is a general property of PSO or results from a match between PSO and that particular classifier. This question is especially relevant when accuracy is the only objective, because higher accuracy does not necessarily mean higher recall, F1-score, or ROC-AUC, and in CKD these error profiles carry different clinical implications.

Against this backdrop, this study investigates whether the effectiveness of Particle Swarm Optimization (PSO)-based hyperparameter optimization varies across seven classification algorithms: Random Forest, Decision Tree, Logistic Regression, K-Nearest Neighbors (KNN), Naïve Bayes, Support Vector Machine (SVM), and Artificial Neural Network (ANN), and whether these variations are associated with classifier characteristics and hyperparameter configurations. Baseline and PSO-optimized models are evaluated using a Chronic Kidney Disease (CKD) dataset comprising 2,304 records, with consistent preprocessing, a common PSO framework, and three data-splitting scenarios. Performance is

assessed using accuracy, precision, recall, F1-score, and the Area Under the Receiver Operating Characteristic Curve (ROC-AUC), complemented by delta analysis relative to baseline models and confusion matrix analysis. The main contribution is empirical evidence of the algorithm-dependent effects of PSO-based optimization, showing that it may improve, marginally change, or degrade performance depending on the classifier being optimized. This study does not claim universal PSO effectiveness or inherent classifier superiority. However, because the analysis is limited to a single dataset, further validation on independent datasets and diverse clinical populations is needed before broader generalization.

METHOD

This study used a controlled comparative experimental design, evaluating each classifier under default (baseline) and PSO-optimized configurations with identical preprocessing, PSO settings, objective functions, and evaluation metrics (Figure 1) to ensure performance differences reflected classifier characteristics rather than procedural variation. The study used kidney_disease_dataset2026.csv, a publicly available secondary dataset from Kaggle comprising 2,304 anonymized patient records with nine clinical attributes: age, creatinine, blood urea nitrogen (BUN), diabetes, hypertension, glomerular filtration rate (GFR), urine output, dialysis requirement, and binary CKD status. The dataset contains no missing values and is nearly balanced (50.87% CKD and 49.13% non-CKD). As the data are publicly available and anonymized, no additional ethical clearance was required. However, we cannot verify their provenance or representativeness of clinical populations, and GFR, creatinine, and BUN may serve as proxies for CKD status because of their direct clinical association with the disease. Consider these limitations when interpreting the findings, which focus on relative algorithmic comparisons rather than estimates of clinical performance.

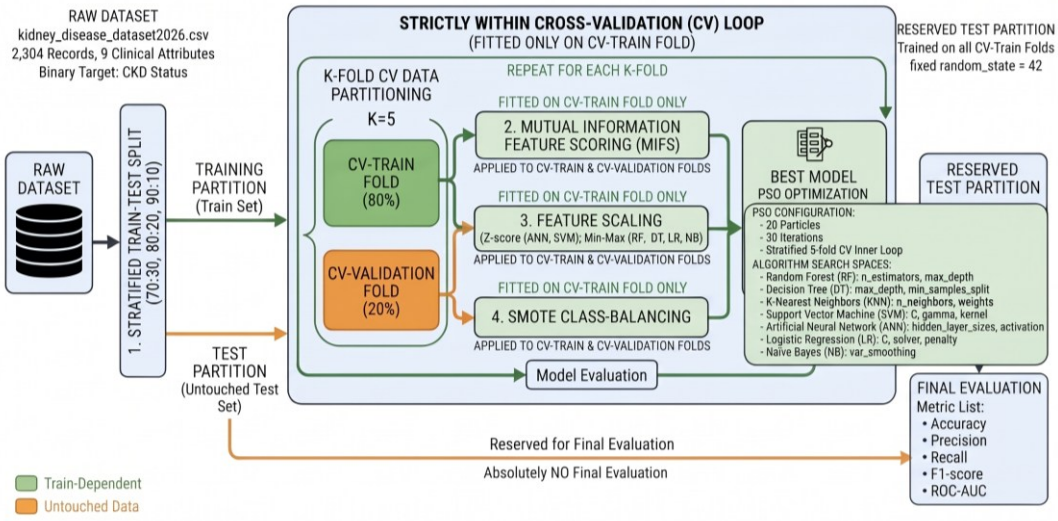

Figure 1. Experimental flow: leakage-free pre-processing, pso-based hyperparameter optimization, and evaluation on test data

To prevent information leakage between training and test data (Kapoor & Narayanan, 2023), we fitted all data-dependent procedures exclusively on the training partitions and then applied them to the corresponding test partitions. The preprocessing pipeline comprised stratified train-test splitting (70:30, 80:20, and 90:10), Mutual Information-based feature selection, feature scaling (z-score for ANN and SVM and Min-Max for the remaining classifiers), and SMOTE applied exclusively to training data. Given the nearly balanced class distribution, we expected SMOTE to have limited effect. We performed the same procedures

within each cross-validation fold to prevent validation-data leakage (Irfannandhy et al., 2024; Syaputa et al., 2025).

Hyperparameter optimization employed Global Best PSO (PySwarms) with 20 particles, 30 iterations, c1 = c2 = 1.5, and w = 0.8. Each particle represented a candidate hyperparameter configuration, with appropriate transformations for integer and categorical variables. Based on 5-fold cross-validation, the optimization process required up to 3,000 model fits per classifier per scenario. We applied identical PSO settings and used negative mean cross-validation accuracy as the fitness function across all seven classifiers. We tailored the search spaces to each algorithm, covering parameters related to model complexity, regularisation, neighborhood structure, and network architecture.

We selected optimized configurations through stratified 5-fold cross-validation within each training partition (inner loop) and then evaluated them on untouched test partitions (outer evaluation), ensuring test data did not influence model selection (Cawley & Talbot, 2010). We compared baseline and PSO-optimized models using accuracy, precision, recall, F1-score, and ROC-AUC, expressing performance differences as Δ = PSO − baseline. We also analyzed confusion matrices for the 80:20 scenario, with particular emphasis on recall and false negatives (FN), given the clinical implications of missed CKD cases.

To support reproducibility, we fixed random_state at 42 and fully documented the PSO configurations and classifier-specific search spaces. However, using a single random seed and test partition per scenario limited assessment of performance stability across different initializations and data splits. Accordingly, we interpreted small performance differences cautiously. In the 80:20 scenario (n = 461), each test observation corresponded to approximately 0.22 percentage points in accuracy. Future studies should employ multiple random seeds and report means, standard deviations, and pairwise statistical tests.

RESULTS AND DISCUSSION

Results

All results represent single estimates based on one random seed and test partition per scenario; therefore, we did not calculate confidence intervals. In the 80:20 scenario, the test set comprised 461 samples (235 positive and 226 negative), yielding a baseline accuracy of 50.98% for a classifier predicting all cases as positive.

Table 1. Accuracy (%) before (baseline) and after pso optimization in three data sharing scenarios and their changes (Δ, percentage points)

Model	70:30 Basic	70:30 PSO	Δ	80:20 Basic	80:20 PSO	Δ	90:10 Basic	90:10 PSO	Δ
Random Forest	100.00	100.00	0.00	97.83	100.00	+2.17	98.27	100.00	+1.73
Decision Tree	61.27	100.00	+38.73	83.95	100.00	+16.05	59.74	100.00	+40.26
Logistic Regression	83.96	84.25	+0.29	83.30	83.30	0.00	83.98	83.55	−0.43
KNN	80.64	83.67	+3.03	82.00	84.16	+2.16	84.85	89.18	+4.33
Naive Bayes	72.11	91.62	+19.51	73.32	91.54	+18.22	75.32	93.51	+18.19
SVM	72.69	50.87	−21.82	74.19	50.98	−23.21	77.49	51.08	−26.41
ANN	79.77	73.41	−6.36	85.68	84.60	−1.08	83.12	88.74	+5.62

Baseline accuracy varied considerably across classifiers, ranging from 59.74% (Decision Tree, 90:10) to 100% (Random Forest, 70:30). PSO produced algorithm-dependent changes rather than uniform improvements. Decision Tree showed the largest gains (Δ = +38.73, +16.05, and +40.26 percentage points across the three scenarios), achieving 100% accuracy in all cases. Naïve Bayes also improved consistently (Δ = +19.51, +18.22, and +18.19), while KNN recorded modest gains (Δ = +3.03, +2.16, and +4.33). Random Forest maintained or reached 100% accuracy, whereas Logistic Regression remained virtually unchanged (Δ = +0.29, 0.00, and −0.43). Conversely, SVM declined across all scenarios (Δ = −21.82, −23.21, and −26.41), reaching approximately 51% accuracy. ANN showed inconsistent changes, declining in the 70:30 and 80:20 scenarios (Δ = −6.36 and −1.08) but improving under 90:10 (Δ = +5.62).

Table 2. Evaluation metrics before (baseline) and after pso optimization in the 80:20 scenario and their changes (Δ)

Model	Condition	Accuracy	Precision	Recall	F1-score	ROC-AUC
Random Forest	Base	0.9783	1,0000	0.9574	0.9783	1,0000
	PSO	1,0000	1,0000	1,0000	1,0000	1,0000
	Δ	+0.0217	0.0000	+0.0426	+0.0217	0.0000
Decision Tree	Base	0.8395	0.9600	0.7149	0.8195	0.8923
	PSO	1,0000	1,0000	1,0000	1,0000	1,0000
	Δ	+0.1605	+0.0400	+0.2851	+0.1805	+0.1077
Logistic Regression	Base	0.8330	0.8435	0.8255	0.8344	0.9092
	PSO	0.8330	0.8435	0.8255	0.8344	0.9092
	Δ	0.0000	0.0000	0.0000	0.0000	0.0000
KNN	Base	0.8200	0.8762	0.7532	0.8101	0.8882
	PSO	0.8416	0.9010	0.7745	0.8330	0.9122
	Δ	+0.0216	+0.0248	+0.0213	+0.0229	+0.0240
Naive Bayes	Base	0.7332	1,0000	0.4766	0.6455	0.9760
	PSO	0.9154	0.9224	0.9106	0.9165	0.9705
	Δ	+0.1822	−0.0776	+0.4340	+0.2710	−0.0055
SVM	Base	0.7419	0.7266	0.7915	0.7576	0.8394
	PSO	0.5098	0.5098	1,0000	0.6753	0.4702
	Δ	−0.2321	−0.2168	+0.2085	−0.0823	−0.3692
ANN	Base	0.8568	0.9043	0.8043	0.8514	0.9187
	PSO	0.8460	0.9362	0.7489	0.8322	0.9154
	Δ	−0.0108	+0.0319	−0.0554	−0.0192	−0.0033

Further analysis of the 80:20 scenario (Table 2) revealed differences beyond accuracy. Decision Tree improved substantially, primarily through increased recall (0.7149 to 1.0000; Δ = +0.2851), alongside higher precision and ROC-AUC, both reaching 1.0000. Random Forest also achieved perfect scores across all five metrics, with recall increasing to 0.9574. Naïve Bayes accuracy increased (Δ = +0.1822), driven by higher recall (0.4766 to 0.9106; Δ = +0.4340), despite a precision decline (Δ = −0.0776). Its F1-score improved (Δ = +0.2710), while ROC-AUC decreased slightly (0.9760 to 0.9705), indicating that the improvement primarily reflected a change in classification sensitivity rather than discrimination ability. KNN

improved modestly across all metrics (accuracy Δ = +0.0216; ROC-AUC Δ = +0.0240), whereas Logistic Regression remained unchanged.

SVM exhibited the most detrimental changes: precision declined from 0.7266 to 0.5098 (Δ = −0.2168), while recall reached 1.0000 (Δ = +0.2085). However, F1-score decreased (Δ = −0.0823), and ROC-AUC fell from 0.8394 to 0.4702 (Δ = −0.3692), indicating discrimination below the random-classification threshold. For ANN, accuracy declined slightly (Δ = −0.0108), while precision increased (Δ = +0.0319) and recall decreased (Δ = −0.0554); ROC-AUC remained nearly unchanged (Δ = −0.0033). Thus, ANN primarily shifted the precision-recall balance rather than improving overall performance.

Table 3. Confusion matrix of 80:20 scenario before (base) and after pso optimization

Model	Condition	TP	FP	FN	TN	Recall	Specificity
Random Forest	Base	225	0	10	226	0.9574	1,0000
	PSO	235	0	0	226	1,0000	1,0000
Decision Tree	Base	168	7	67	219	0.7149	0.9690
	PSO	235	0	0	226	1,0000	1,0000
Logistic Regression	Base	194	36	41	190	0.8255	0.8407
	PSO	194	36	41	190	0.8255	0.8407
KNN	Base	177	25	58	201	0.7532	0.8894
	PSO	182	20	53	206	0.7745	0.9115
Naive Bayes	Base	112	0	123	226	0.4766	1,0000
	PSO	214	18	21	208	0.9106	0.9204
SVM	Base	186	70	49	156	0.7915	0.6903
	PSO	235	226	0	0	1,0000	0.0000
ANN	Base	189	20	46	206	0.8043	0.9115
	PSO	176	12	59	214	0.7489	0.9469

The TP, FP, FN, and TN counts in Table 3 are reconstructed from the accuracy, precision, and recall in Table 2 with a test data set of 461 samples (235 positive and 226 negative). For each model and condition, one integer matrix corresponds to all three metrics, and all are consistent with the reported F1-score. Specificity is calculated as TN/(TN + FP). False negatives (FN) decreased from 10 to 0 in Random Forest, 67 to 0 in Decision Tree, 123 to 21 in Naïve Bayes, and 58 to 53 in KNN, while Naïve Bayes false positives (FP) increased from 0 to 18. Logistic Regression remained unchanged (FN = 41). In contrast, ANN increased FN from 46 to 59 despite reducing FP from 20 to 12. SVM showed the most severe deterioration: although it detected all 235 positive cases (FN = 0), it misclassified all 226 negative cases as positive (FP = 226; TN = 0), resulting in zero specificity and an accuracy of 0.5098, equivalent to the positive-class proportion in the test set.

These outcomes suggest that PSO effects vary with classifier characteristics and their respective search spaces (Table 1). Decision Tree and Naïve Bayes showed the largest changes, reflecting the influence of parameters controlling tree complexity and decision sensitivity. Random Forest achieved 100% accuracy across all three scenarios despite its restricted search space, while KNN exhibited moderate changes and Logistic Regression produced identical predictions before and after optimization. Conversely, PSO failed to improve SVM and ANN, whose hyperparameters govern decision boundaries and network architecture, respectively.

The dataset was nearly balanced (50.87% CKD-positive), indicating that class imbalance does not explain the observed performance differences. GFR, creatinine, and BUN also

exhibited the highest mutual information scores, suggesting that classification relies heavily on a limited number of informative clinical features. Therefore, the near-perfect tree-based results (FP = 0 and FN = 0 in the 80:20 test set) and 100% accuracy across scenarios should be interpreted cautiously, given the possibility of high class separability in this dataset. The Discussion section examines this issue further.

Discussion

Hyperparameter optimization does not produce uniform benefits across classification algorithms. PSO yielded substantial improvements in Decision Tree and Naïve Bayes, moderate gains in KNN, negligible changes in Random Forest and Logistic Regression, and performance deterioration in SVM. These findings support the no-free-lunch principle and previous evidence that hyperparameter importance varies across algorithms (Probst et al., 2019). The study provides controlled empirical evidence from CKD classification that the same optimizer can improve one classifier while degrading another. Thus, optimization effectiveness should be understood through the interaction among the optimization mechanism, classifier structure, search space, and objective function (Bischl et al., 2023).

Comparisons with previous studies further demonstrate this variability. The Decision Tree improvement is consistent with Fitri and Baita (2025), whereas SVM deterioration contrasts with the improvement reported by Purwaningsih and Nurelasari (2024). ANN also showed inconsistent responses compared with Huang and Chou (2019). Such differences may reflect variations in datasets, search spaces, objective functions, and validation protocols. Changes in algorithm rankings across prediction tasks (Ais & Sanjaya, 2025; Febriyanti et al., 2025) further highlight the influence of dataset characteristics (Uddin & Lu, 2024) and the importance of reporting performance changes rather than only the best-performing classifier.

These differences may reflect how hyperparameters shape model behavior. In tree-based models, parameters such as max_depth, min_samples_split, and min_samples_leaf directly control partition complexity. In KNN and Naïve Bayes, neighborhood configuration and variance smoothing influence prediction sensitivity. Notably, Naïve Bayes shifted from a conservative classification pattern (precision = 1.0000; recall = 0.4766) to substantially higher sensitivity (recall = 0.9106). Conversely, PSO-optimized SVM classified all test observations as positive (TP = 235, FP = 226, FN = 0, TN = 0), resulting in zero specificity and ROC-AUC of 0.4702. This outcome may indicate that the accuracy-based objective and selected C and γ search ranges were insufficient to preserve discriminatory performance. However, alternative search spaces and decision-boundary analyses are needed to establish the underlying mechanism.

From a clinical screening perspective, false negatives are particularly important because missed CKD cases may delay further assessment. PSO reduced FN from 123 to 21 in Naïve Bayes, while Random Forest and Decision Tree achieved zero FN. Conversely, ANN showed increased FN from 46 to 59 despite higher precision (0.9362). The SVM results further demonstrate that perfect recall is insufficient when specificity is zero. Therefore, classifier evaluation should consider recall, specificity, F1-score, ROC-AUC, and confusion matrices alongside accuracy. Explainability methods such as SHAP (Lundberg & Lee, 2017) may also support interpreting predictions before considering high-performing models for clinical decision support.

The near-perfect performance of Random Forest and Decision Tree requires cautious interpretation. GFR, BUN, and creatinine are closely associated with CKD diagnosis (Stevens et al., 2024) and may act as proxies for the target label, potentially increasing class separability. Consequently, high performance on this public dataset does not establish clinical readiness, particularly given the unverifiable data provenance and potential differences from real-world clinical records. Although performance consistency across three splitting scenarios provides

some evidence of split-related stability, it does not replace external validation using independent datasets and diverse clinical populations (Ganie et al., 2023).

Several limitations should be acknowledged. First, we used only one random seed (42) and one test partition per scenario, without statistical significance testing. Second, the uniform PSO configuration and accuracy-based objective may not suit all classifiers equally, so SVM and ANN deterioration does not necessarily indicate an inherent incompatibility with PSO. Third, we did not perform repeated nested cross-validation or external validation. Finally, we reconstructed confusion matrices from evaluation metrics rather than obtaining them directly from model predictions, and we limited ROC analysis to AUC values without full ROC curves. Future research should investigate classifier-specific or adaptive optimization, multi-objective functions incorporating recall, precision, and ROC-AUC, and explainable AI. Multiple random seeds, paired statistical tests, and independent external datasets are also needed to establish the robustness and generalisability of the findings.

CONCLUSION

This study concludes that the effect of PSO-based hyperparameter optimization on CKD classification depends on the optimized algorithm, so PSO cannot be considered a universally effective method for improving performance. On the public dataset and configuration used, Decision Tree and Naïve Bayes showed the largest improvements; KNN improved moderately; Random Forest reached 100% accuracy; Logistic Regression remained almost unchanged; SVM degraded to the point of losing its discriminatory power; and ANN responded inconsistently across scenarios. These findings are limited to a single dataset without external validation, so a near-perfect score on a tree-based model cannot be taken as evidence of generalizability. A generalizable methodological implication is that optimization success depends on the fit between the optimizer's search mechanism, the model's decision-boundary characteristics, and the available hyperparameter space. Therefore, reported improvements for one algorithm cannot be assumed to apply to others, and authors should report precision, recall, and ROC-AUC alongside accuracy. This contribution emphasizes the need for optimization strategies that are adaptive to classifier types, multi-objective objective functions that incorporate recall and ROC-AUC, and more robust validation through multiple seeds, statistical tests, and external datasets in future machine learning optimization research.

REFERENCES

Ais, S. R., & Sanjaya, U. P. (2025). Perbandingan algoritma Random Forest, XGBoost, dan Logistic Regression untuk prediksi risiko kekambuhan kanker tiroid. *Edumatic: Jurnal Pendidikan Informatika, 9*(1), 324-332. https://doi.org/10.29408/edumatic.v9i1.29664

Anbazhagan, T., & Rangaswamy, B. (2025). Early prediction of CKD from time series data using adaptive PSO optimized echo state networks. *Scientific Reports, 15*(1), 6966. https://doi.org/10.1038/s41598-025-91028-6

Bergstra, J., & Bengio, Y. (2012). Random search for hyper-parameter optimization. *Journal of Machine Learning Research, 13* , 281–305.

Bikbov, B., Purcell, C. A., Levey, A. S., Smith, M., Abdoli, A., Abebe, M., ... & Owolabi, M. O. (2020). Global, regional, and national burden of chronic kidney disease, 1990–2017: a systematic analysis for the Global Burden of Disease Study 2017. *The lancet*, *395*(10225), 709-733. https://doi.org/10.1016/S0140-6736(20)30045-3

Bischl, B., Binder, M., Lang, M., Pielok, T., Richter, J., Coors, S., ... & Lindauer, M. (2023). Hyperparameter optimization: Foundations, algorithms, best practices, and open challenges. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, *13*(2), e1484. https://doi.org/10.1002/widm.1484

Cawley, G. C., & Talbot, N. L. (2010). On over-fitting in model selection and subsequent selection bias in performance evaluation. *The Journal of Machine Learning Research, 11*, 2079-2107.

Febriyanti, N. R., Kusrini, K., & Hartanto, A. D. (2025). Analisis perbandingan algoritma SVM, Random Forest dan Logistic Regression untuk prediksi stunting balita. *Edumatic: Jurnal Pendidikan Informatika, 9*(1), 149-158. https://doi.org/10.29408/edumatic.v9i1.29407

Fitri, L. A., & Baita, A. (2025). Optimization of Decision Tree Algorithm for Chronic Kidney Disease Classification Based on Particle Swarm Optimization (PSO). *Journal of Applied Informatics and Computing, 9*(1), 178-186. https://doi.org/10.30871/jaic.v9i1.8940

Ganie, S. M., Dutta Pramanik, P. K., Mallik, S., & Zhao, Z. (2023). Chronic kidney disease prediction using boosting techniques based on clinical parameters. *Plos one, 18*(12), e0295234. https://doi.org/10.1371/journal.pone.0295234

Gao, Y. H., Huang, X. W., & Meng, Y. M. (2025). A systematic evaluation of data preprocessing and model optimization for machine learning algorithms: using sphalerite trace element data as an example. *Journal of Asian Earth Sciences, 292*, 106728. https://doi.org/10.1016/j.jseaes.2025.106728

Hidayangsih, P. S., Tjandrarini, D. H., Sukoco, N. E. W., Sitorus, N., Dharmayanti, I., & Ahmadi, F. (2023). Chronic kidney disease in Indonesia: evidence from a national health survey. *Osong Public Health and Research Perspectives, 14*(1), 23. https://doi.org/10.24171/j.phrp.2022.0290

Huang, M. L., & Chou, Y. C. (2019). Combining a gravitational search algorithm, particle swarm optimization, and fuzzy rules to improve the classification performance of a feed-forward neural network. *Computer methods and programs in biomedicine*, *180*, 105016. https://doi.org/10.1016/j.cmpb.2019.105016

Iftikhar, H., Hashem, A. F., Mohamud, L. A., Al-Moisheer, A. S., Medina, R. I. G., & López-Gonzales, J. L. (2026). An intelligent ensemble machine learning model for early detection of chronic kidney disease in aging populations. *Scientific Reports, 16*(1), 3021. https://doi.org/10.1038/s41598-025-32919-6

Irfannandhy, R., Handoko, L. B., & Ariyanto, N. (2024). Analisis Performa Model Random Forest dan CatBoost dengan Teknik SMOTE dalam Prediksi Risiko Diabetes. *Edumatic: Jurnal Pendidikan Informatika, 8*(2), 714-723. https://doi.org/10.29408/edumatic.v8i2.27990

Islam, M. A., Majumder, M. Z. H., & Hussein, M. A. (2023). Chronic kidney disease prediction based on machine learning algorithms. *Journal of pathology informatics, 14*, 100189. https://doi.org/10.1016/j.jpi.2023.100189

Kapoor, S., & Narayanan, A. (2023). Leakage and the reproducibility crisis in machine-learning-based science. *Patterns, 4*(9). https://doi.org/10.1016/j.patter.2023.100804

Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. *Proceedings of ICNN'95-international conference on neural networks* (Vol. 4, pp. 1942-1948). IEEE. https://doi.org/10.1109/ICNN.1995.488968

Lundberg, S. (2017). A unified approach to interpreting model predictions. *arXiv preprint arXiv:1705.07874*.

Probst, P., Boulesteix, A. L., & Bischl, B. (2019). Tunability: Importance of hyperparameters of machine learning algorithms. *Journal of Machine Learning Research, 20*(53), 1-32.

Purwaningsih, E., & Nurelasari, E. (2024). Peningkatan Akurasi Metode Support Vector Machine melalui Particle Swarm Optimization pada Penyakit Ginjal Kronis. *Inf. Manag. Educ. Prof*, *9*(1), 61.

Shami, T.M., El-Saleh, A.A., Alswaitti, M., Al-Tashi, Q., Summakieh, M.A., & Mirjalili, S. (2022). Particle swarm optimization: A comprehensive survey. *IEEE Access, 10*, 10031–10061. https://doi.org/10.1109/ACCESS.2022.3142859

Stevens, P. E., Ahmed, S. B., Carrero, J. J., Foster, B., Francis, A., Hall, R. K., ... & Levin, A. (2024). KDIGO 2024 clinical practice guideline for the evaluation and management of chronic kidney disease. *Kidney international, 105*(4), S117-S314. https://doi.org/10.1016/j.kint.2023.10.018

Swain, D., Mehta, U., Bhatt, A., Patel, H., Patel, K., Mehta, D., ... & Manika, S. (2023). A robust chronic kidney disease classifier using machine learning. *Electronics, 12*(1), 212. https://doi.org/10.3390/electronics12010212

Syaputa, M. R., Arifin, M., & Fithri, D. L. (2025). Klasifikasi Sentimen Ulasan E-Wallet menggunakan TF-IDF dan Random Forest dengan Penyeimbangan Data SMOTE. *Edumatic: Jurnal Pendidikan Informatika, 9*(2), 532-541. https://doi.org/10.29408/edumatic.v9i2.31084

Uddin, S., & Lu, H. (2024). Dataset meta-level and statistical features affect machine learning performance. *Scientific Reports, 1*4(1), 1670. https://doi.org/10.1038/s41598-024-51825-x

Wulandari, K. A., Nugraha, A., Luthfiarta, A., & Nisa, L. R. (2024). Peningkatan Akurasi Deteksi Dini Penyakit Parkinson melalui Pendekatan Ensemble Learning dan Seleksi Fitur Optimal. *Edumatic: Jurnal Pendidikan Informatika, 8*(2), 575-584. https://doi.org/10.29408/edumatic.v8i2.27788