Scientific Journal of Informatics
Vol. 12, No. 4, November 2025

p-ISSN 2407-7658 http://journal.unnes.ac.id/nju/index.php/sji e-ISSN 2460-0040

Optimization of Logistic Regression Algorithm Using Grey Wolf Optimizer for Credit Card Fraud Detection

Wiyanda Puspita[1*], M. Faris Al Hakim[2]
[1,2]Department of Computer Science, Universitas Negeri Semarang, Indonesia

Abstract.
Purpose: The advancement of digital technology has significantly changed the financial transaction system, but has also led to an increase in cybercrime, especially credit card fraud. This crime poses a significant financial threat, with reported losses reaching hundreds of millions of dollars annually. This study aims to improve the effectiveness of fraud detection using the Logistic Regression (LR) algorithm, which although widely used in binary classification, is still vulnerable to challenges with imbalanced data. The goal is to optimize LR using the Grey Wolf Optimizer (GWO) to improve accuracy and reliability.
Methods: This research implements a Logistic Regression (LR) model whose hyperparameters are optimized using Grey Wolf Optimizer (GWO) algorithm. The model was trained and tested on a public Kaggle dataset containing 284,807 credit card transactions. Data preprocessing includes handling outliers using Interquartile Range (IQR) method and handling class imbalance using KMeansSMOTE. Evaluation metrics include accuracy, precision, recall, f1-score, and specificity based on confusion matrix.
Result: The baseline LR model achieved 99.92% accuracy, 75.18% precision, 74.73% recall, 75.45% F1-score, and 99.96% specificity. After GWO optimization, the model improved to 99.94% accuracy, 85.96% precision, 83.08% recall, 84.01% F1-score, and 99.97% specificity, showing a significant performance boost. This represents a notable improvement in key metrics for fraud detection, with an increase of 14.3% in precision, 11.2% in recall, and 11.3% in the F1-score, demonstrating a more robust model.
Novelty: This study proposed the application of the Grey Wolf Optimizer (GWO) for hyperparameter tuning of a Logistic Regression model in the context of fraud detection. Unlike conventional optimization techniques that can be computationally expensive, our GWO-based approach offers an efficient and effective method for discovering optimal model settings. The optimized model not only outperforms the baseline LR but also presents a scalable and powerful solution for financial institutions to improve the accuracy of their fraud detection systems.

Keywords: Fraud detection, Grey wolf optimizer, Logistic regression, Optimization
Received June 2025 / **Revised** December 2025 / **Accepted** December 2025

INTRODUCTION

The advent of digital technology has precipitated a paradigm shift within the financial sector, giving rise to innovations such as big data, artificial intelligence, blockchain, and the internet of things [1], [2], [3]. This transformation modifies the transaction system from conventional methods to modern cashless payment systems such as debit cards, credit cards, and QRIS, which offer convenience and security in transactions [4], [5], [6]. However, concomitant with this progress, significant challenges emerge in the domain of digital security, particularly the proliferation of fraudulent activities such as credit card fraud [7], [8], [9], [10]. According to a report by the Federal Trade Commission (FTC), the United States experienced 114,694 cases of credit card fraud in 2023, resulting in financial losses amounting to USD 246.5 million. On a global scale, the financial losses incurred as a result of credit card fraud amounted to USD 34 billion in 2022, and it is anticipated that this figure will rise to USD 43 billion by 2026.

In response to these threats, the financial industry has begun to adopt technologies such as machine learning to detect suspicious transactions in real-time [11], [12], [13]. A multitude of algorithms have been utilized in this context, including those such as Graph Neural Network, Graph-Based Anomaly Detection, and Support Vector Machine [14], [15], [16]. However, it should be noted that the majority of these methods possess certain drawbacks in terms of computational efficiency and adaptability to evolving fraud patterns. Consequently, this research focuses on the Logistic Regression (LR) algorithm, prized for its computational efficiency and interpretability in binary classification tasks like fraud detection [17], [18], [19]. However, LR exhibits limitations in processing imbalanced data, necessitating the implementation of an optimization approach.

[*] Corresponding author.
Email addresses: wiyandapuspita5730@students.unnes.ac.id (Puspita)[*], farishakim@mail.unnes.ac.id (Hakim)

DOI: 10.15294/sji.v12i4.26807

In order to enhance the efficacy of LR, this research incorporates the Grey Wolf Optimizer (GWO) optimization algorithm to make precise adjustments to the model parameters [20], [21]. The GWO, which draws inspiration from the social behavior of grey wolves, has been demonstrated to be effective in optimizing LR weights and biases while preventing overfitting. Known for its excellent balance between the exploration and exploitation phases of the search process, GWO is adept at navigating complex solution spaces to find global optima, often outperforming other popular metaheuristics like Particle Swarm Optimization (PSO) or Genetic Algorithms (GA) which can be prone to premature convergence in certain scenarios [22]. The objective of this research is to develop a more accurate and adaptive credit card transaction fraud detection model through a combination of LR and GWO, a method that has not been widely explored in previous studies. It is anticipated that this approach will enhance the reliability and relevance of fraud detection systems in addressing security challenges in the digital era.

Research related to the detection of fraud in credit card transactions has been carried out with various approaches and methods. Research by Itoo & Satwinder [23] compared Logistic Regression (LR), Naïve Bayes, and KNN algorithms using Random Under-Sampling (RUS) to overcome data imbalance. However, this approach has the potential to remove important information from the majority class, thereby decreasing the model's accuracy. Khalid et al. [24] implemented an ensemble approach that incorporated machine learning (ML) algorithms, including logistic regression (LR), k-nearest neighbors (KNN), random forest, bagging, and AdaBoost, in conjunction with resampling techniques such as under-sampling and synthetic minority over-sampling technique (SMOTE). This approach was further augmented by feature selection, leading to an LR accuracy of 94.4%. However, it should be noted that the study did not involve the application of optimization methods. Concurrently, Dang et al. [25] employed the LR algorithm in conjunction with the SMOTE and ADASYN approaches. Their findings indicated that the maximum accuracy attained by LR was 97.53% following resampling. However, this level of accuracy was achieved without the implementation of performance enhancement strategies through hyperparameter optimization.

Another study by Mniai et al. [26] employed a range of algorithms, including LR, and optimized the SVDD model using the PSLPSO method. Despite the evident enhancement in the outcomes pertaining to SVDD, the utilization of LR merely resulted in an 88% accuracy rate. This is attributable to the implementation of under-sampling, a process with the potential to diminish classification performance. Furthermore, Y. Tang & Liang [27] proposed Federated Graph Learning and compared several algorithms, including CLR and FCNN. The former algorithm obtained 96.49% accuracy. However, the cosine similarity technique employed is not sufficiently sophisticated to manage imbalanced data, and it lacks integration with advanced optimization methodologies.

The critical research gap, therefore, is the lack of a holistic approach that moves beyond data preprocessing to fundamentally enhance the model itself. While resampling adjusts the data to fit an algorithm, hyperparameter optimization adjusts the algorithm to better fit the complexities of the original data—a potentially more powerful and robust strategy. The application of a sophisticated metaheuristic algorithm like GWO to optimize an LR model for this specific problem represents a significant, underexplored opportunity. This study aims to fill this gap by proposing an integrated LR-GWO model. We hypothesize that this combination will yield a more effective and reliable solution for fraud detection than methods that rely solely on resampling techniques or non-optimized models.

METHODS

This study employs a quantitative approach to detect fraudulent transactions in a credit card activity dataset. The methodology is based on a computational experiment, training and testing machine learning algorithms. Specifically, this research focuses on the Logistic Regression (LR) algorithm as the main classification algorithm and the Grey Wolf Optimizer as the optimization algorithm. Figure 1 below presents a comprehensive representation of the research flowchart.

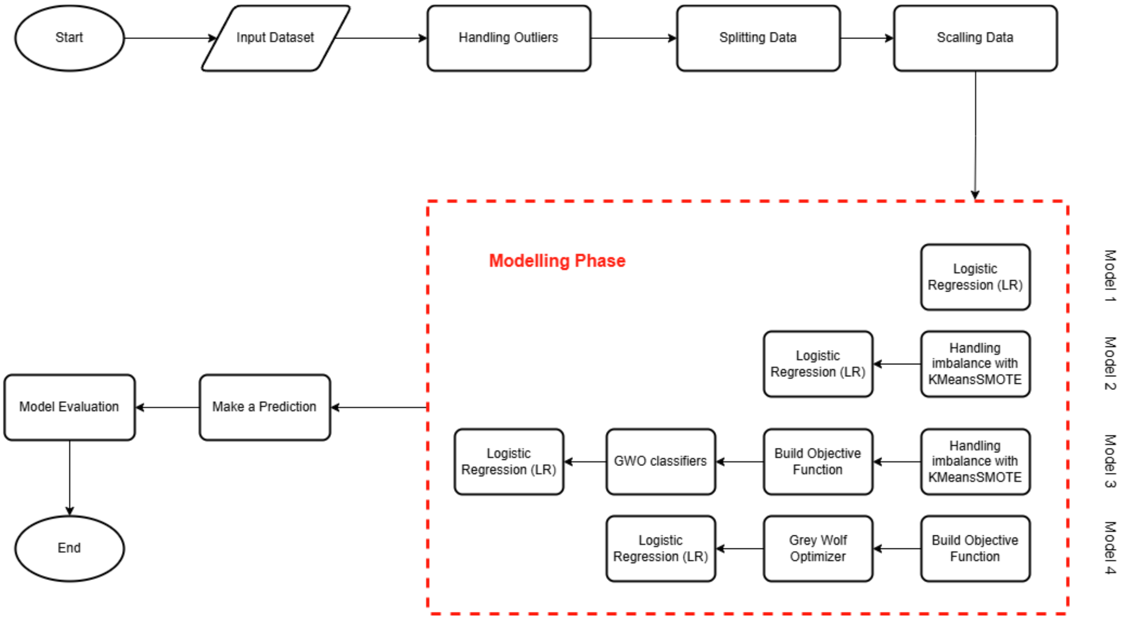

Figure 1. Research flowchart

Dataset

This research uses a public dataset search technique conducted through the Kaggle website for data collection. The obtained dataset is named Credit Card Fraud Detection. It contains data from credit card transactions made in September 2013. It contains 284,807 numeric records and 31 features. The target variable in this study is a feature class containing a binary column of 0 (non-fraud) or 1 (fraud). The Machine Learning Group (http://mlg.ulb.ac.be) at ULB (Université Libre de Bruxelles) collected and analyzed this dataset. It is a public dataset that has been widely used in previous fraud detection studies [23], [24], [25], [26], [27].

Preprocessing

The preprocessing phase commenced with missing value identification through comprehensive column-wise dataset analysis. Once identified, it will be handled such as deleting data that contains null values or filling the data using mean, median, or mode values. Subsequent steps involved outlier detection using the Interquartile Range method by calculating the difference between third and first quartiles. Data points exceeding one and a half times the IQR from either quartile were identified as outliers and subsequently treated using winsorization technique to preserve the original distribution characteristics.

The procedure continued with dataset partitioning through stratified sampling to maintain minority class representation. A 70-30 ratio was employed for training and testing sets respectively, with a fixed random state to ensure reproducible results [26]. This splitting was conducted after data cleaning but prior to feature transformation. The final stage involved standardization of numerical features using StandardScaler. The scaling process was implemented separately on training and testing data post-splitting, with transformation parameters calculated exclusively from the training set to prevent information leakage into the test data.

Handling Imbalance

Class imbalance refers to a scenario where the proportion between majority (normal transactions) and minority (fraud) classes is severely skewed. In our dataset, the minority class constitutes merely 0.172% of total instances, representing an extreme imbalance that may lead to model bias toward the majority class. While standard techniques like Random Over-sampling can lead to overfitting and basic SMOTE is known to sometimes generate noisy samples in overlapping class regions, a more robust strategy was deemed necessary.

To address this, we employ KMeansSMOTE, an advanced SMOTE variant combining K-means clustering with oversampling [28], exclusively on the training data to prevent data leakage [29]. This method was specifically chosen over other variants due to its intelligent sample generation process. Instead of applying

SMOTE globally, KMeansSMOTE first identifies clusters of data points using K-Means. It then strategically generates synthetic samples only within clusters that have a safe and high concentration of minority class instances. The *sampling_strategy* parameter in KMeansSMOTE will be tuned to achieve an optimal ratio, targeting a minority class representation exceeding 40% of total samples while preserving the original data distribution [30]. This implementation leverages the *imbalanced-learn* library, with rigorous evaluation of its impact on model performance through precision-recall metrics.

Logistic Regression

For the binary classification task, this study employs Logistic Regression (LR), a standard statistical model effective for predicting binary outcomes [31]. The model's core function in this research is to calculate the probability of a transaction being fraudulent. It achieves this by mapping a linear combination of input features to a probability score between 0 and 1 using the sigmoid function, which is mathematically represented in Equation 1 [32]. This process generates an S-shaped curve, as illustrated in Figure 2.

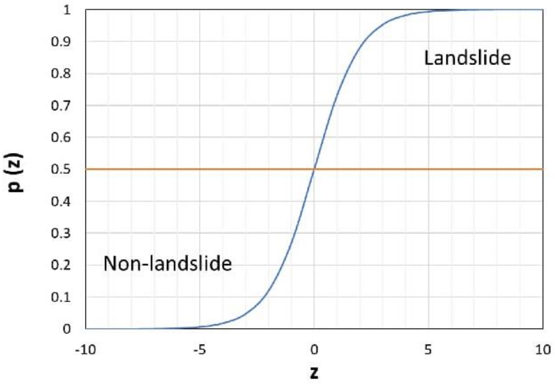

Figure 2. Sigmoid Function [33]

$$\sigma\,(z)\ =\ \frac{1}{1+e^{-z}} \tag{1}$$

Where:
$e =$ Euler’s number, usually the base of the natural logarithm $(\approx 2{,}718)$.
$z = \sum w_i\, x_i + bias$ or $z = \beta_0 + \beta_1 x_1 + \cdots + \beta_n x_n$.

The output of this function is interpreted as the predicted probability, which allows for the classification of each transaction as either fraudulent or non-fraudulent based on a defined threshold.

Grey Wolf Optimizer

The Grey Wolf Optimizer (GWO) is a metaheuristic algorithm inspired by the social hierarchy and hunting behavior of grey wolves [34]. In the GWO, the alpha, beta, and delta wolf leaders play a critical role in establishing efficient hierarchies and interactions to achieve optimal convergence. The alpha wolf is responsible for determining the optimal solution, while the beta and delta wolves collaborate to explore more extensive regions. GWO can be optimized to design predictive models that obtain the best parameters for involving relevant features. By utilizing the principle of wolf social behavior, GWO can help determine the optimal weights for features in Logistic Regression [20], [35].

The process begins with the random initialization of wolf positions ($\vec{X}_i$) and GWO parameters, including the number of agents (population size) and maximum iterations. The parameter $\vec{a}$ is initialized to 2, and the iteration counter (t) starts at 0. While the current iteration has not reached the maximum, the algorithm calculates the adaptive coefficients $\vec{A}$ and $\vec{C}$ using random vectors (*r₁*, *r₂*) as in Equation 2 and Equation 3 and the linearly updated $\vec{a}$ value as in Equation 4.

$$\vec{A} = 2.\vec{a}.\vec{r}_1 - \vec{a} \tag{2}$$

$$\vec{C} = 2.\vec{r}_2 \tag{3}$$

$$\vec{a} = 2 - \frac{2t}{t_{max}} \tag{4}$$

In each iteration, the fitness of each solution is evaluated based on the objective function. Objective function is a function to optimize the LR algorithm as listed in Equation 5. In this approach, the LR model is built and trained using the training data and the model weights are changed to adjust the resulting value of the wolf location in the vector.

$$fitness\ (t) = \frac{1}{1+E_i(t)} \tag{5}$$

The result of the objective function evaluation is then referred to as the solution. The three best solutions (α, β, δ) are selected, and their positions are stored. The positions of the remaining wolves (ω) are updated by following the hierarchy of α, β, and δ through three key steps:

1. Calculating the distance ($\vec{D}$) to each leader using Equation 6.
2. Updating temporary positions ($\vec{X}_1(t+1),\ \vec{X}_2(t+1),\ \vec{X}_3(t+1)$) based on Equation 7.
3. Determining the new position ($\vec{X}(t+1)$) as the average of the three temporary positions as in Equation 8.

If the fitness of the new solution improves, the α, β, and δ values are updated. This process repeats until the maximum iteration is reached, and the optimal solution (α) is returned as the output. The process highlights GWO's exploration and exploitation mechanism through dynamic and parameters, as well as its computational efficiency via selective storage of the best solutions.

$$\begin{cases} \vec{D}_\alpha = \left|\vec{C}_1 \cdot \vec{X}_\alpha(t) - \vec{X}(t)\right| \\ \vec{D}_\beta = \left|\vec{C}_2 \cdot \vec{X}_\beta(t) - \vec{X}(t)\right| \\ \vec{D}_\delta = \left|\vec{C}_3 \cdot \vec{X}_\delta(t) - \vec{X}(t)\right| \end{cases} \tag{6}$$

$$\begin{cases} \vec{X}_1(t+1) = \vec{X}_\alpha(t) - \vec{A}_1 \cdot \left(\vec{D}_\alpha\right) \\ \vec{X}_2(t+1) = \vec{X}_\beta(t) - \vec{A}_2 \cdot \left(\vec{D}_\beta\right) \\ \vec{X}_3(t+1) = \vec{X}_\delta(t) - \vec{A}_3 \cdot \left(\vec{D}_\delta\right) \end{cases} \tag{7}$$

$$\vec{X}(t+1) = \frac{\vec{X}_1(t+1)+\vec{X}_2(t+1)+\vec{X}_3(t+1)}{3} \tag{8}$$

Evaluation Metrics

The performance of the proposed fraud detection system was rigorously evaluated using a comprehensive set of metrics to assess different aspects of classification effectiveness. Confusion matrix is a matrix that measures model performance based on actual and predicted values [36]. In the case of binary classification, the confusion matrix is depicted as in Figure 3. The confusion matrix has 4 (four) entries, namely True Positive, True Negative, False Positive, and False Negative [37]. The confusion matrix served as the foundation for deriving key indicators including accuracy, precision, recall (sensitivity), specificity, and F1-score [23]. Accuracy as shown in Equation 9 measured the overall proportion of correct predictions, while precision quantified the reliability of fraud alerts by calculating the fraction of true fraud cases among all transactions flagged as fraudulent as given in Equation 10. Recall in Equation 11 evaluated the model's ability to identify actual fraud cases, and specificity in Equation 12 assessed its performance in correctly recognizing legitimate transactions. The F1-score as shown in Equation 13 provided a balanced measure by harmonizing precision and recall, which is particularly valuable for imbalanced datasets. Additionally, the AUC-ROC analysis was conducted to examine the model's discrimination ability across various classification thresholds.

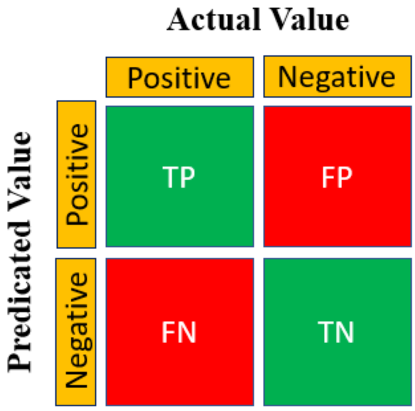

Figure 3. Confusion Matrix [37]

To ensure thorough evaluation, all metrics were carefully interpreted in the context of credit card fraud detection requirements. While high precision minimizes false alarms that could inconvenience customers, strong recall reduces financial losses by catching more fraudulent transactions. The AUC-ROC curve visualization helped demonstrate the optimal balance between sensitivity (recall) and specificity across different decision thresholds. This multi-metric approach enabled a nuanced understanding of model performance, addressing both the technical aspects of classification and the practical implications for fraud detection systems. The mathematical formulations of these metrics are presented below for reference.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{9}$$

$$Precision = \frac{TP}{TP+FP} \tag{10}$$

$$Recall = \frac{TP}{TP+FN} \tag{11}$$

$$Specificity = \frac{TN}{TP+FN} \tag{12}$$

$$F1 = \frac{2 \times Precision \times Recall}{Precision+Recall} \tag{13}$$

RESULT AND DISCUSSION

This section describes the results and discussion of the research conducted, namely fraud classification models using LR and GWO algorithms. The discussion begins with the result of pre-processing, model implementation, and evaluation of the algorithm model used to get optimal accuracy as the output of this research.

Preprocessing Result

The preprocessing stage consists of several processes, including identifying and handling outliers, splitting data, and scaling data. Outliers were identified using the Interquartile Range (IQR) method. Some features such as V27, V28, and Amount have a high number of outliers. Handling is done with the clipping technique, which trims extreme values to normal limits. The results of boxplot visualization before and after handling can be seen in Figure 4 and Figure 5.

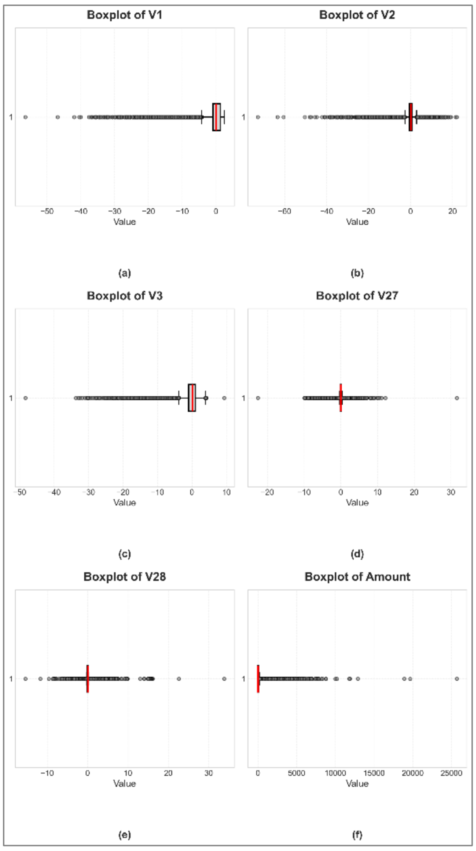

Figure 4. Boxplot of each feature before handling

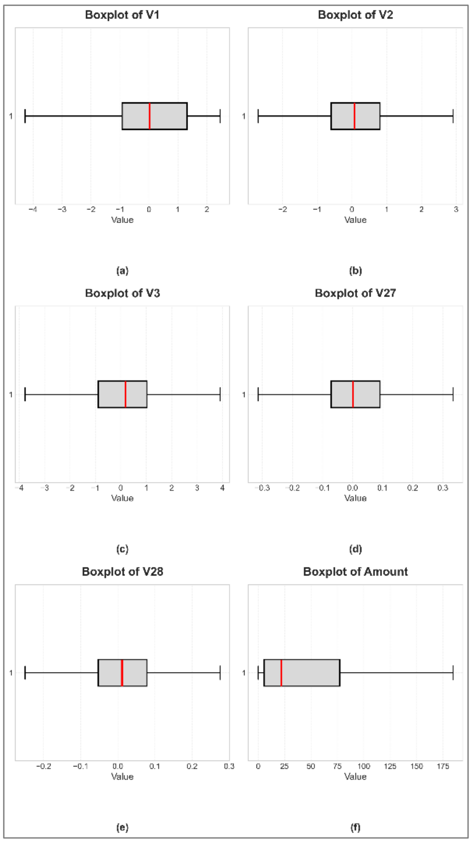

Figure 5. Boxplot of each feature after handling

After handling outliers, the dataset is divided into training data and test data with a ratio of 70:30 [26] using stratified sampling to maintain class proportions. To overcome class imbalance, the KMeansSMOTE method was applied to the training data to avoid data leakage [29]. After resampling, the proportion of minority classes increased to 41.18%, close to being balanced with the majority class (58.82%). In addition to data leakage, another consideration is the originality of the testing data. If resampling is also applied to the testing data, the evaluation results are no longer objective because the data created is potentially not in accordance with the original [38]. Bar chart of target variables before and after resampling can be seen in Figure 6.

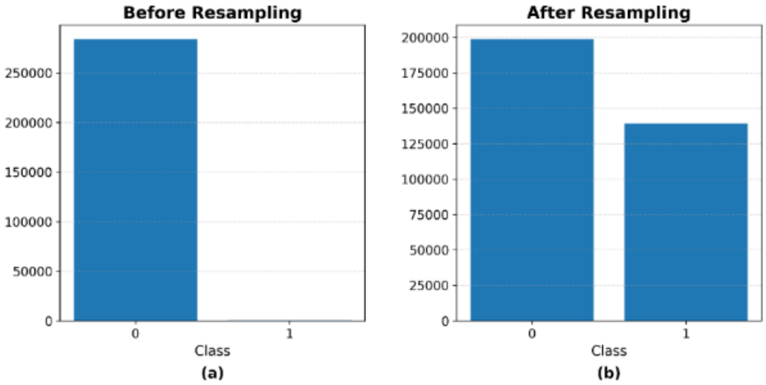

Figure 6. (a) Class Distribution Before Resampling and (b) Class Distribution After Resampling

Model Implementation

In this study, a series of experiments were conducted on 4 (four) classification models to evaluate the performance of the proposed approach. The 4 models evaluated in this study were Logistic Regression (LR) as the model 1, LR with KMeansSMOTE as the model 2, a combination of LR, KMeansSMOTE, and GWO as the model 3, and LR with GWO as the model 4. The evaluation process involved the utilization of 5-fold cross-validation, with a particular emphasis on the f1-score metric, to assess the performance of the models in the initial, third, fourth, and fifth experiments. The outcomes of the cross-validation process for each model are presented in Table 1.

Table 1. Cross-Validation Result

Model	Fold-1	Fold-2	Fold-3	Fold-4	Fold-5	Mean
LR	0,7938	0,6984	0,704	0,6861	0,7943	0,7233
LR + KMeansSMOTE	0,9793	0,9795	0,9780	0,9791	0,9795	0,979
LR + KMeansSMOTE + GWO	0,7791	0,9816	0,9825	0,9818	0,982	0,9414
LR + GWO	0.8281	0,75	0,7969	0,845	0,826	0,8092

The fourth and fifth experiments employed GWO as an optimization algorithm. The model was evaluated using the objective function delineated in Equation 5, which yielded solutions designated as α, β, and δ. Solution α is regarded as the optimal solution, solution β is considered the second-best solution, and solution δ is identified as the third-best solution. The outcomes of α in both objective functions are presented in Table 2. The optimal weight is defined as the value of α, except for the final value in the list, while the optimal bias is defined as the final value in α. Subsequently, the LR model is configured using the optimal weight and bias.

Table 2. GWO Result

Features	Alpha (α)	Beta (β)	Delta (δ)
Time	0	0	0
V1	0.0021	0.0021	0.0021
V2	0.0631	0.0627	0.0626
V3	-0.0316	-0.0312	-0.0314
…	...	...	...
V27	0.0057	0.0057	0.0057
V28	0.0933	0.0928	0.0934
Amount	0	0	0
Bias	-0.9908	-0.9909	-0.9949

Model Evaluation

Evaluation is conducted to determine the performance of the model built. Models are evaluated using relevant metrics such as confusion matrix, accuracy, precision, recall or sensitivity, specificity, f1-score, and AUC. In this section, we will evaluate 5 models that have been built in the previous stage. The confusion matrix results can be seen in Table 3.

Table 3. Confusion Matrix Result

Model	TP	FP	TN	FN
LR	103	34	85273	33
LR+KMeansSMOTE	126	2028	83279	10
LR+KMeansSMOTE+GWO	124	873	84434	12
LR+GWO	113	20	85287	23

In addition to employing the confusion matrix as a primary metric, this research is evaluated using a range of other evaluation metrics, including accuracy, precision, recall, f1-score, and specificity. The results of the evaluation metrics in this study are presented in Table 4. The results of the evaluation demonstrate that the LR+GWO model exhibits the most balanced performance, with an accuracy of 99.94%, a precision of 85.96%, a recall of 83.08%, an f1-score of 84.01%, and a specificity of 99.97%. A comparison of the f1-score produced by the Logistic Regression (LR) and GWO models reveals a notable enhancement in performance following the implementation of GWO optimization. Specifically, the LR model yielded an f1-score of 75.45%. Concurrently, the implementation of KMeansSMOTE led to a substantial enhancement in recall, with a percentage exceeding 91%. However, this approach concomitantly resulted in a significant decline in precision, culminating in a substantial deterioration of the f1-score to 11% and 21.88%, respectively. This finding indicates that while the resampling method is effective in identifying a greater number of fraud cases, it also leads to a significant number of misclassifications. Consequently, the integration of LR with GWO was identified as the most efficacious and consistent approach for accurately detecting fraudulent transactions.

Table 4. Evaluation Metrics Results

Model	Accuracy	Precision	Recall	F1-score	Specificity
LR	99,92%	75,18%	74,73%	75,45%	99,96%
LR+KMeansSMOTE	97,61%	5,84%	**92,64%**	11%	97,62%
LR+KMeansSMOTE+GWO	98,96%	12,43%	91,17%	21,88%	98,97%
LR+GWO	**99,94%**	**85,96%**	83,08%	**84,01%**	**99,97%**

Discussion

The Logistic Regression (LR) algorithm in this study proved to be able to classify credit card transactions in general quite well. This can be seen from the accuracy value that reached 99.92% and specificity of 99.96%, indicating a high ability to recognize non-fraud transactions. However, LR's performance in detecting fraud transactions as a minority class is still moderate, with a precision of 75.18%, recall of 74.73%, and f1-score of 75.45%. This shows that although the model is quite accurate overall, it is not sensitive enough in identifying fraud cases that are very small in number. The extreme imbalance of class distribution (only 0.17% of the data is classified as fraud) is the main challenge that causes the model to be biased towards the majority class [37].

To overcome this, the KMeansSMOTE method is applied which aims to balance the class distribution by performing structured oversampling of fraud data [38]. Although this technique managed to drastically increase recall to 92.64% in the LR+KMeansSMOTE model and 91.17% in LR+KMeansSMOTE+GWO, it had a negative impact on precision and f1-score. Precision dropped dramatically to 5.84% and f1-score to only 11% in the LR+KMeansSMOTE model. This is due to the large number of false positive predictions, i.e. non-fraud transactions that are classified as fraud. The main cause is the new data synthesis process by SMOTE which, while increasing the amount of fraud data, does not guarantee that the synthetic samples are truly representative of the original pattern. As a result, the model becomes too aggressive in recognizing the fraud class but loses the ability to distinguish between genuine and synthetic transactions, resulting in an overall decrease in prediction accuracy.

Conversely, the implementation of Grey Wolf Optimizer (GWO) to optimize the weights and bias in LR yielded substantial enhancements in all evaluation metrics. The LR+GWO model demonstrated an enhancement in precision to 85.96%, recall to 83.08%, and f1-score to 84.01%, with the highest accuracy of 99.94%. GWO facilitates the identification of optimal solutions through an adaptive search mechanism that emulates the social behavior of wolves in the pursuit of prey. This approach enhances the precision of the model in differentiating between fraud and non-fraud classes, while demonstrating robust performance in the presence of imbalanced data without generating a high number of false positives. Consequently, GWO was found to be a more effective approach than the resampling method in enhancing the performance of LR in detecting fraudulent transactions.

In the context of research, the novelty of a study cannot be considered in isolation; it must be situated within the broader research field. A critical evaluation of research results requires a comparison with previous studies, which is an essential step in the research process. By means of a comparative analysis, it will be possible to evaluate the advantages, uniqueness, and limitations of the research, as well as to gain insight into the extent to which the research is able to overcome the limitations or fill the gaps that exist in previous research. This research conducts a comparative study of the results obtained with previous studies that also use CCFD datasets, but apply different optimization techniques. The accuracy comparison between this study and previous studies is presented in Table 5.

Table 5. Accuracy Comparison with Previous Research

Research	Methods	Accuracy
[23]	Logistic Regression	95%
[24]	Logistic Regression	94,4%
[25]	Logistic Regression dan SMOTE	97.53%
[26]	Logistic Regression	88%
[27]	Centralized Logistic Regression	96,49%
Proposed Method	Logistic Regression dan Grey Wolf Optimizer	**99,94%**

The comparison results indicate that the proposed model consistently yields higher accuracy than previous methods. For instance, research by Itoo & Satwinder [23] using Logistic Regression with Random Under-Sampling technique only achieved 95% accuracy. Khalid et al. [24] combined undersampling, SMOTE, and feature selection with 94.4% accuracy, while the approach of Dang et al. [25] who used Logistic Regression and SMOTE obtained the highest result of 97.53%. In this study, the LR+GWO model achieved an accuracy of 99.94%, showing a significant advantage.

Another study by Mniai et al. [26] used the Logistic Regression method with an under-sampling and feature selection approach, but only produced an accuracy of 88%. Meanwhile, research by Y. Tang & Liang [27] applied the Centralized Logistic Regression (CLR) method with cosine similarity to handle class imbalance and achieved 96.49% accuracy. In comparison, the method proposed in this study, namely Logistic Regression optimized with Grey Wolf Optimizer (LR+GWO), managed to obtain an accuracy of 99.94%. These results show that the method used in this study is consistently superior and more effective in improving fraud detection accuracy than the approaches used in these studies.

The superiority of the model is evident not only in its accuracy but also in its balanced approach to precision and recall, which is crucial for fraud detection, a process susceptible to false negatives and false positives. Most of the extant research in this field has focused on enhancing accuracy and fraud detection. However, there is a paucity of research that addresses the misclassification of non-fraud transactions, which can have a direct impact on the user experience. The LR+GWO model demonstrates a notable capacity to preserve this equilibrium, as evidenced by its elevated f1-score and sustained specificity. GWO's metaheuristic approach is characterized by its independence from data manipulation techniques, such as resampling. Instead, it prioritizes the optimization of model parameters, leveraging the existing data structure for this purpose.

Therefore, based on empirical evaluation and comparison with previous research, it can be concluded that the LR+GWO method makes a real contribution in improving the effectiveness of credit card transaction fraud detection systems. This approach has been demonstrated to enhance statistical performance while concurrently providing a more stable and reliable solution for real-world systems characterized by highly imbalanced data distribution.

CONCLUSION

This research successfully builds a credit card transaction fraud detection model using the Logistic Regression algorithm, which has been optimized by the Grey Wolf Optimizer (GWO). The evaluation results demonstrate the efficacy of Logistic Regression in differentiating fraudulent transactions from authentic ones, exhibiting commendable initial performance. The efficacy of GWO as an optimization method has been demonstrated to enhance various model evaluation metrics, including precision, recall, and f1-score, which are critical indicators in fraud detection scenarios. A key contribution of this work is its novel application of GWO for hyperparameter tuning of an LR model in this specific domain, an approach that has been notably underexplored in existing literature. The superior and stable performance of the LR-GWO model compared to the baseline and findings in related studies confirms that this integration is an effective, reliable, and promising solution for fraud detection systems.

Despite the positive outcomes, this study has several acknowledged limitations. First, the research relies on a publicly available Kaggle dataset; while standard for benchmarking, its age may not fully represent the most contemporary and evolving fraud patterns. Second, the scope was intentionally focused on optimizing a single, interpretable algorithm (Logistic Regression) and did not include a comparative performance analysis against more complex, computationally intensive models like deep neural networks. These factors should be considered when contextualizing the results.

Notwithstanding the encouraging outcomes, this research demonstrates potential for further development. Future work should explore alternative methods for handling class imbalance, such as cost-sensitive learning, which directly penalizes the misclassification of minority class instances. To further validate the choice of optimizer, a comprehensive benchmarking study is needed to compare GWO's performance against other metaheuristic algorithms like PSO, GA, or Differential Evolution. Finally, to enhance real-world relevance, it is crucial to validate the proposed model on newer, proprietary fraud datasets to ensure its robustness against current fraud tactics..

REFERENCES

[1] X. Yang, L. Kong, and S. Qu, “Technology in Society Evolution of technology cooperation networks based on networked evolutionary games model : An industrial heterogeneity perspective,” *Technol Soc*, vol. 78, no. December 2023, p. 102631, 2024, doi: 10.1016/j.techsoc.2024.102631.

[2] S. Elia, M. Giuffrida, M. M. Mariani, and S. Bresciani, “Resources and digital export : An RBV perspective on the role of digital technologies and capabilities in cross-border e-commerce,” *J Bus Res*, vol. 132, no. November 2020, pp. 158–169, 2021, doi: 10.1016/j.jbusres.2021.04.010.

[3] V. Sharma, K. Jangir, M. Gupta, and R. Rupeika-apoga, “International Journal of Information Does service quality matter in FinTech payment services ? An integrated SERVQUAL and TAM approach,” *International Journal of Information Management Data Insights*, vol. 4, no. 2, p. 100252, 2024, doi: 10.1016/j.jjimei.2024.100252.

[4] E. Isaia, N. Oggero, and D. Sandretto, “Journal of Behavioral and Experimental Finance Is financial literacy a protection tool from online fraud in the digital era ?,” *J Behav Exp Finance*, vol. 44, no. September, p. 100977, 2024, doi: 10.1016/j.jbef.2024.100977.

[5] W. Li, X. Liu, and L. Zhou, “The role of digital finance in enhancing export competitiveness: insights from China’s listed companies,” *Financ Res Lett*, vol. 71, no. September, p. 106376, 2025, doi: https://doi.org/10.1016/j.frl.2024.106376.

[6] Q. Luo and J. Wang, “Digital finance and green technology innovation : A dual path test based on market and government,” *Financ Res Lett*, vol. 70, no. 217, p. 106283, 2024, doi: 10.1016/j.frl.2024.106283.

[7] S. L. Fulford and S. D. Schuh, “Credit cards , credit utilization , and consumption ☆,” *J Monet Econ*, no. xxxx, p. 103619, 2024, doi: 10.1016/j.jmoneco.2024.103619.

[8] A. Gunawan, A. Farrah, and F. Fatikasari, “The Effect of Using Cashless (QRIS) on Daily Payment Transactions Using the Technology Acceptance Model,” *Procedia Comput Sci*, vol. 227, pp. 548–556, 2023, doi: 10.1016/j.procs.2023.10.557.

[9] C. A. Maher and T. A. Engle, “Knowing is half the battle_ Examining the association between acknowledgement of victimization and reporting of fraud,” *Journal of Economic Criminology*, vol. 5, no. August, p. 100092, 2024, doi: 10.1016/j.jeconc.2024.100092.

[10] W. Duan, N. Hu, and F. Xue, “The information content of financial statement fraud risk : An ensemble learning approach,” *Decis Support Syst*, vol. 182, no. March 2023, p. 114231, 2024, doi: 10.1016/j.dss.2024.114231.

[11] C. Gomes and Z. Jin, “Insurance fraud detection with unsupervised deep learning,” no. May, pp. 591–624, 2021, doi: 10.1111/jori.12359.

[12] T. P. Ogundunmade and A. A. Adepoju, “Modelling Credit Card Fraud Data using Machine Learning Algorithms,” pp. 43–49, 2024.

[13] A. Rb and S. K. Kr, “Credit card fraud detection using artificial neural network,” *Global Transitions Proceedings*, vol. 2, no. 1, pp. 35–41, 2021, doi: 10.1016/j.gltp.2021.01.006.

[14] D. Cheng, X. Wang, Y. Zhang, and L. Zhang, “Graph Neural Network for Fraud Detection via Spatial-Temporal Attention,” *IEEE Trans Knowl Data Eng*, vol. 34, no. 8, pp. 3800–3813, 2022, doi: 10.1109/TKDE.2020.3025588.

[15] T. Pourhabibi, K. Ong, B. H. Kam, and Y. Ling, “Fraud detection : A systematic literature review of graph-based anomaly detection approaches,” *Decis Support Syst*, vol. 133, no. August 2019, p. 113303, 2020, doi: 10.1016/j.dss.2020.113303.

[16] K. G. Al-hashedi and P. Magalingam, “Financial fraud detection applying data mining techniques : A comprehensive review from 2009 to 2019,” vol. 40, 2021, doi: 10.1016/j.cosrev.2021.100402.

[17] M. Devika and S. R. Kishan, “Credit Card Fraud Detection Using Logistic Regression,” *2022 Second International Conference on Advanced Technologies in Intelligent Control, Environment, Computing & Communication Engineering (ICATIECE)*, pp. 1–6, 2022, doi: 10.1109/ICATIECE56365.2022.10046976.

[18] P. K. Kumar, P. S. Kiran, S. Kouashik, V. Koushik, K. K. Kumar, and A. K. Chaitanya, “Credit Card Fraud Detection Using Machine,” no. December, 2023.

[19] K. Shah, H. Patel, D. Sanghvi, and M. Shah, “A Comparative Analysis of Logistic Regression, Random Forest and KNN Models for the Text Classification,” *Augmented Human Research*, vol. 5, no. 1, 2020, doi: 10.1007/s41133-020-00032-0.

[20] N. M. H. Azlan, M. Zain, R. Sallehuddin, and Y. Yusoff, “Recent studies on optimisation method of Grey Wolf Optimiser (GWO): a review (2014 – 2017),” *Artif Intell Rev*, vol. 52, no. 4, pp. 2651–2683, 2019, doi: 10.1007/s10462-018-9634-2.

[21] X. Zhang, Y. Zhang, and Z. Ming, “Improved dynamic grey wolf optimizer *,” vol. 22, no. 6, pp. 877–890, 2021.

[22] H. Faris, I. Aljarah, M. A. Al-Betar, and S. Mirjalili, “Grey wolf optimizer: a review of recent variants and applications,” *Neural Comput Appl*, vol. 30, no. 2, pp. 413–435, 2018, doi: 10.1007/s00521-017-3272-5.

[23] F. Itoo and M. Satwinder, “̈ ve Bayes Comparison and analysis of logistic regression , Naı and KNN machine learning algorithms for credit card fraud detection,” *International Journal of Information Technology*, vol. 13, no. 4, pp. 1503–1511, 2021, doi: 10.1007/s41870-020-00430-y.

[24] A. R. Khalid, N. Owoh, O. Uthmani, M. Ashawa, J. Osamor, and J. Adejoh, “Enhancing Credit Card Fraud Detection: An Ensemble Machine Learning Approach,” *Big Data Cognitive Computing*, vol. 8, no. 6, pp. 1–27, 2024, doi: https://doi.org/10.3390/bdcc8010006.

[25] T. K. Dang, T. C. Tran, and L. M. Tuan, “Machine Learning Based on Resampling Approaches and Deep Reinforcement Learning for Credit Card Fraud Detection Systems,” *Applied Sciences*, vol. 11, p. 10004, 2021, doi: https://doi.org/10.3390/app112110004.

[26] A. Mniai, M. Tarik, and K. Jebari, “A Novel Framework for Credit Card Fraud Detection,” *IEEE Access*, vol. 11, no. September, pp. 112776–112786, 2023, doi: 10.1109/ACCESS.2023.3323842.

[27] Y. Tang and Y. Liang, “Credit card fraud detection based on federated graph learning,” *Expert Syst Appl*, vol. 256, no. July, p. 124979, 2024, doi: 10.1016/j.eswa.2024.124979.

[28] Q. Zeng, Z. Gong, S. Wu, C. Zhuang, and S. Li, “International Journal of Applied Earth Observation and Geoinformation Measuring cyclists ’ subjective perceptions of the street riding environment using K-means SMOTE-RF model and street view imagery,” *International Journal of Applied Earth Observation and Geoinformation*, vol. 128, no. February, p. 103739, 2024, doi: 10.1016/j.jag.2024.103739.

[29] A. Demircioğlu, “Applying oversampling before cross - validation will lead to high bias in radiomics,” *Sci Rep*, pp. 1–11, 2024, doi: 10.1038/s41598-024-62585-z.

[30] H. Al, M. Islam, E. Öykü, A. Mehtap, and K. Ulukök, “HCAB - SMOTE : A Hybrid Clustered Affinitive Borderline SMOTE Approach for Imbalanced Data Binary Classification,” *Arab J Sci Eng*, vol. 45, no. 4, pp. 3205–3222, 2020, doi: 10.1007/s13369-019-04336-1.

[31] J. Kwaku *et al.*, “A supervised machine learning algorithm for detecting and predicting fraud in credit card transactions,” *Decision Analytics Journal*, vol. 6, no. January, p. 100163, 2023, doi: 10.1016/j.dajour.2023.100163.

[32] X. Zou, “Logistic Regression Model Optimization and Case Analysis,” pp. 135–139, 2019.

[33] F. S. Tehrani, G. Santinelli, and M. Herrera, “A framework for predicting rainfall-induced landslides using machine learning methods,” *Geotechnical Engineering*, no. September, 2019, doi: 10.32075/17ECSMGE-2019-0521.

[34] S. Mirjalili, S. Mohammad, and A. Lewis, “Advances in Engineering Software Grey Wolf Optimizer,” *Advances in Engineering Software*, vol. 69, 2014.

[35] K. Meidani, A. Hemmasian, S. Mirjalili, and A. B. Farimani, “Adaptive grey wolf optimizer,” *Neural Comput Appl*, vol. 34, no. 10, pp. 7711–7731, 2022, doi: 10.1007/s00521-021-06885-9.

[36] M. S. Sandeep, K. Tiprak, S. Kaewunruen, P. Pheinsusom, and W. Pansuk, “Shear strength prediction of reinforced concrete beams using machine learning,” *Structures*, vol. 47, no. June 2022, pp. 1196–1211, 2023, doi: 10.1016/j.istruc.2022.11.140.

[37] A. E. Maxwell, T. A. Warner, and L. A. Guillén, “Accuracy assessment in convolutional neural network-based deep learning remote sensing studies—part 1: Literature review,” *Remote Sens (Basel)*, vol. 13, no. 13, 2021, doi: 10.3390/rs13132450.

[38] M. Khushi, K. Shaukat, T. M. Alam, X. Yang, and M. C. Reyes, “A Comparative Performance Analysis of Data Resampling Methods on Imbalance Medical Data,” vol. 9, 2021.