Enhancing Banking Data Classification Through Hybrid L2 Regularisation and Early Stopping in Artificial Neural Networks

Khairul Nizam Abd Halim[1, 3], Abdul Syukor Mohamad Jaya[2]*, Fauziah Kasmin[2], Azlan Abdul Aziz[3]
Fakulti Teknologi Maklumat dan Komunikasi, Universiti Teknikal Malaysia Melaka, Melaka, Malaysia[1]
Fakulti Kecerdasan Buatan dan Keselamatan Siber, Universiti Teknikal Malaysia Melaka, Melaka, Malaysia[2]
Fakulti Sains Komputer dan Matematik, Universiti Teknologi MARA, Malaysia[3]

***Abstract*—The demand for robust data-driven classification (DDC) techniques remains critical in banking applications, where accurate and efficient decision-making is paramount. Artificial Neural Networks (ANNs), particularly Multi-Layer Perceptrons (MLPs), are widely used due to their strong learning capabilities. However, their performance often depends on effective hyperparameter tuning and regularisation strategies to avoid overfitting. This study aims to enhance the efficiency of the MLP training process by introducing a hybrid approach that integrates L2 regularisation with Early Stopping (ES) into the hyperparameter tuning procedure. The key contribution lies in embedding both techniques within a grid search framework, thereby streamlining the search for optimal hyperparameters. The proposed method was evaluated using three real-world banking datasets: two related to loan subscription (16 and 20 features) and one concerning credit card default payment (23 features). Experimental results demonstrate that the hybrid approach reduces hyperparameter tuning time by over 90% while achieving high classification performance. Notably, the Receiver Operating Characteristic - Area Under the Curve (ROC-AUC) scores of 93.89% and 91.21% were achieved on the loan datasets, and 73.28% on the credit card dataset, surpassing previous benchmarks. These findings highlight the potential of the L2ES hybrid method to improve both the accuracy and computational efficiency of DDC in financial applications.**

Keywords—Artificial neural networks; L2 regularisation; early stopping; banking; classification

I. Introduction

The financial sector's digital transformation has intensified the demand for accurate and efficient data analysis techniques. Financial institutions continuously seek innovative strategies to enhance operational efficiency, particularly in loan processing, risk assessment, and fraud detection. In this context, data-driven classification (DDC) methods have emerged as essential tools for analysing large volumes of banking data to support informed decision-making. Among these, machine learning (ML)-based DDC techniques stand out because they can learn from complex, nonlinear data patterns and improve predictions over time. Their application spans multiple domains, including medicine, agriculture, and manufacturing, with particularly significant adoption in banking for tasks such as loan subscription analysis and credit card default prediction.

Artificial neural networks (ANNs), particularly the multi-layer perceptron (MLP), have gained recognition as robust classifiers within the DDC domain. Numerous studies have demonstrated the superior performance of MLPs in banking-related classification tasks. For example, MLPs have been effectively used to model customer decisions on loan subscriptions and to predict credit card default risks, as summarised in Tables I and II of this study. The robustness and adaptability of MLPs are further evidenced in domains such as bioinformatics. AbdElsalam et al. [1], for instance, significantly improved MLP performance in gene classification tasks by integrating the ADASYN technique to address class imbalance—an approach yielding high sensitivity and accuracy. These developments reaffirm that MLPs, when appropriately optimised, remain highly effective across various domains, including finance.

However, overfitting remains a critical challenge in MLPs, particularly when working with complex or imbalanced datasets. Regularisation techniques, such as L2 regularisation, have been widely adopted to mitigate this issue. Prior research by Aldelemy et al. [2] and Grosicki [3] demonstrates the efficacy of L2 regularisation in MLP-based classification for banking loan datasets. L2 regularisation has also been successfully applied to other ML models, including logistic regression and XGBoost, across both loan and credit card datasets [4]–[6]. Despite these applications, current implementations often overlook the considerable time and effort required for hyperparameter tuning in neural network models—an essential yet burdensome aspect of model optimisation.

This study identifies a gap in the literature: while L2 regularisation has been implemented to reduce overfitting in MLPs, limited attention has been given to streamlining the hyperparameter tuning process. Most existing approaches apply L2 within fixed or manually guided tuning routines, without exploring strategies to expedite this phase effectively. This oversight hinders the deployment of efficient ML systems in time-sensitive financial applications.

To address this, this paper proposes a hybrid approach that combines L2 regularisation with early stopping (ES) within a grid search framework. This technique aims to retain the regularisation benefits of L2 while simultaneously accelerating the hyperparameter tuning process via early termination of

*Corresponding Author.

underperforming configurations. By integrating ES into the grid search routine, the proposed method significantly lowers computational overhead and facilitates faster convergence to optimal settings, thus making the model development process more efficient.

Unlike conventional methods that treat L2 regularisation and hyperparameter optimisation as independent steps, our hybrid strategy embeds both into a unified workflow. This integration is particularly suitable for banking datasets, where rapid model development is critical. The proposed approach not only curbs overfitting but also enhances the efficiency of hyperparameter exploration—an aspect often neglected in previous studies.

TABLE I. LIST OF RESEARCH ON THE APPLICATION OF MLPS FOR PREDICTING LOAN SUBSCRIPTION THROUGH BANK TELEMARKETING CAMPAIGNS

Research	ROC-AUC	Other Metrics	Features
Moro *et al.*, [7]	0.79	-	22
Marinakos and Daskalaki [8]	0.87	Accuracy 79.11%, Precision 74.99%, Recall 81.44%, F1 77.86%	16
Farooqi and Iqbal [9]	0.89	Accuracy 89.83%, Precision 55.50%, Recall 49.00%, F1 52%	20
Ghatasheh *et al.*, [10]	-	Accuracy 84.18%, TP 61.4%	16
Panigrahi and Patnaik [11]	-	Accuracy 90.02%	16
Mokrane [12]	-	Accuracy 98.93%, F1 0.95	20
Dutta and Bandyopadhyay [13]	-	Accuracy 88.32%	16
Masturoh *et al.*, [14]	0.90	Accuracy 94.27%	20
Aldelemy and Raed A. Abd-Alhameed [2]	0.92	-	16

TABLE II. LIST OF RESEARCH ON THE APPLICATION OF MLPS FOR PREDICTING CREDIT CARD DEFAULT PAYMENTS

Research	ROC-AUC	Other Metrics	Features
Singh and Aggarwal [15]	-	Recall 0.849, Precision 0.866	23
Liu [16]	-	Accuracy 82.27%, F1 0.46	23
Vishwakarma *et al.*, [17]	0.5000	-	23
de Campos Souza and Torres [18]	0.6506	-	23
Almajid [19]	0.7184	Accuracy 76.50%	23
Jiang *et al.*, [20]	-	Accuracy 77.35%	23
Shazly and Khodadadi [21]	-	Accuracy 89.45%, Recall 0.9967, Precision 0.6667, F1 0.9288	23
Idrees *et al.*, [22]	-	Accuracy 81.96%, Recall 0.820, Precision 0.803	23
Yash *et al.*, [23]	-	Accuracy 80.50%	23

Accordingly, this research evaluates the performance of the hybrid L2ES (L2 + Early Stopping) approach in banking data classification. The primary objectives are to (i) assess the effectiveness of integrating L2 regularisation within MLP-based classifiers, (ii) incorporate L2 into the grid search procedure for hyperparameter tuning, and (iii) demonstrate how the combination of L2 and ES within grid search can enhance training efficiency without compromising model accuracy.

The remainder of this paper is structured as follows: Section II introduces the proposed L2ES methodology, detailing the experimental setup and the grid search configuration. Section III presents the empirical results and offers a comparative analysis with conventional approaches. Section IV concludes the paper and discusses directions for future research.

II. METHODOLOGY

A. Materials

The datasets utilized in this study are derived from the banking sector, specifically focusing on loan and default credit card datasets. These datasets are publicly accessible through the UCI Machine Learning Repository. Table III presents a detailed overview of the loan subscription dataset, as documented by Moro et al. [24]. This dataset encompasses 16 features and one label, the latter serving as the target variable. The dataset comprises 45,211 records featuring a mix of data types, including integers, binary and categorical. The binary target variable is imbalanced: 'yes' labels account for 5,289 records, while 'no' labels constitute 39,922.

TABLE III. OVERVIEW OF THE LOAN SUBSCRIPTION DATASET DOCUMENTED BY MORO ET AL. [24]

Num.	Variable Name	Role	Type
1	age	Feature	Integer
2	job	Feature	Categorical
3	marital	Feature	Categorical
4	education	Feature	Categorical
5	default	Feature	Binary
6	balance	Feature	Integer
7	housing	Feature	Binary
8	loan	Feature	Binary
9	contact	Feature	Categorical
10	day_of_week	Feature	Date
11	month	Feature	Date
12	duration	Feature	Integer
13	campaign	Feature	Integer
14	pdays	Feature	Integer
15	previous	Feature	Integer
16	poutcome	Feature	Categorical
17	y	Target	Binary

Table IV outlines the characteristics of another loan subscription dataset, also studied by Moro et al. [7]. This contains 20 features and one label. It includes a mix of integer, float, and categorical data types for its features. Like the first, it exhibits a binary target with an imbalanced distribution among its 41,188 records: 4,640 are labelled 'yes', and 36,548 are labelled 'no'.

Table V details the dataset concerning bank credit card default payments sourced from Yeh and Lien [25]. This dataset comprises 23 features and one label, encompassing 30,000 records. The features are of integer data type, while the label is binary. The dataset exhibits label imbalance, with 6,636 records labelled '1' (indicating default) and 23,364 records labelled '0' (indicating no default).

TABLE IV. CHARACTERISTICS OF THE SECOND LOAN SUBSCRIPTION DATASET STUDIED BY MORO ET AL. [7]

Num.	Variable Name	Role	Type
1	age	Feature	Integer
2	job	Feature	Categorical
3	marital	Feature	Categorical
4	education	Feature	Categorical
5	credit	Feature	Categorical
6	housing	Feature	Categorical
7	loan	Feature	Categorical
8	contact	Feature	Categorical
9	month	Feature	Categorical
10	day of week	Feature	Categorical
11	duration	Feature	Integer
12	campaign	Feature	Integer
13	pdays	Feature	Integer
14	previous	Feature	Integer
15	poutcome	Feature	Categorical
16	emp.var.rate	Feature	Float
17	cons.price.idx	Feature	Float
18	cons.conf.idx	Feature	Float
19	euribor3m	Feature	Float
20	nr.employed	Feature	Integer
21	y	Target	Binary

TABLE V. DATASET ON BANK CREDIT CARD DEFAULT PAYMENTS SOURCED FROM YEH AND LIEN [25]

Num.	Variable Name	Role	Type
1	X1: limit	Feature	Integer
2	X2: gender	Feature	Integer
3	X3: education	Feature	Integer
4	X4: marital	Feature	Integer
5	X5: age	Feature	Integer
6	X6: past payment	Feature	Integer
7	X7: past payment	Feature	Integer
8	X8: past payment	Feature	Integer
9	X9: past payment	Feature	Integer
10	X10: past payment	Feature	Integer
11	X11: past payment	Feature	Integer
12	X12: bill statement	Feature	Integer
13	X13: bill statement	Feature	Integer
14	X14: bill statement	Feature	Integer
15	X15: bill statement	Feature	Integer
16	X16: bill statement	Feature	Integer
17	X17: bill statement	Feature	Integer
18	X18: previous payment	Feature	Integer
19	X19: previous payment	Feature	Integer
20	X20: previous payment	Feature	Integer
21	X21: previous payment	Feature	Integer
22	X22: previous payment	Feature	Integer
23	X23: previous payment	Feature	Integer
24	y	Target	Binary

B. *Preparation*

The experimental setup adhered to past and present research's standard data-driven classification methodologies. Specifically, this study adopted the methodology illustrated in Fig. 1, adapted from A. Almajid [19]. All experiments were conducted within the Google Colab server environment, with relevant specifications for 2023 to 2025. The MLP model development commenced with a data cleaning phase, employing mode imputation to address missing data. Subsequently, the dataset was split using an 80:20 ratio into training and test sets, with the training set used to construct the MLP model. Before model creation, the training set transformed one-hot encoding for categorical features and standard scalar scaling for numerical features. The MLP classifier was selected as the machine learning algorithm to fulfil the primary objective of this paper, which is to augment the performance of the MLP classifier. The baseline model incorporated L2 regularisation to mitigate overfitting during the hyperparameter tuning process via Grid Search. Conversely, the innovative model combined L2 regularisation and ES to address overfitting concerns. The rationale behind this hybridisation was the anticipation of improved MLP classifier performance.

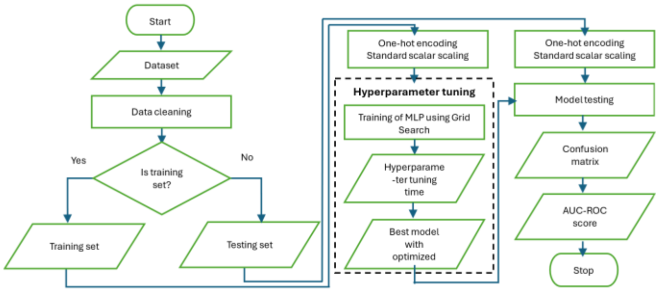

Fig. 1. Methodology for MLP model development and experimentation process.

Table VI outlines an algorithm for the hyperparameter tuning process that employs L2 regularisation in MLP models using Grid Search. This process is initiated by defining a hyperparameter grid, including the regularisation parameter, number of hidden layers, neuron range, activation function, and solver parameters.

A Grid Search Loop with 5-fold Cross-Validation follows, systematically exploring diverse hyperparameter combinations. Each iteration involves initialising and training an MLP model with L2 regularisation and monitoring training loss for convergence. The model's performance is assessed using the Receiver Operating Characteristic - Area Under the Curve (ROC-AUC) on held-out folds, with the average ROC-AUC calculated across all folds. Each iteration records and accumulates training time from the previous iteration. Ultimately, the hyperparameter combination yielding the highest average performance metric is selected, resulting in the best MLP model with optimised hyperparameters.

Table VII introduces an innovative algorithm for the hyperparameter tuning process in MLP models utilising L2 regularisation and ES through Grid Search. Initially, a hyperparameter grid specifies parameters such as regularisation strength, hidden layer configuration, and activation function.

TABLE VI. ALGORITHM FOR HYPERPARAMETER TUNING WITH L2 REGULARISATION IN MULTI-LAYER PERCEPTRON (MLP) USING GRID SEARCH

Input	: Training set
Output	: Best MLP model with optimized hyperparameters
Step 1	Define Hyperparameter Grid for MLP: − Set regularisation parameter λ for L2 regularisation to 0.0001. − Set number of hidden layers to 1. − Define a range for the number of neurons per hidden layer from 5 to 100. − Set activation function to ReLU. − Set solver to Adam. − Set batch size (minibatch) to the minimum of 200. − Set initial learning rate to 0.001. − Set maximum iterations to 5000. − Set shuffle sample to True. − Set exponential decay rate for estimates of the first moment vector in Adam to 0.999. − Set exponential decay rate for estimates of the second moment vector in Adam to 1e-8.
Step 2	Grid Search Loop with 5-Fold Cross-Validation: − For each combination of hyperparameters in the defined grid: − Initialize an MLP model with the specified architecture and hyperparameters. − Divide the training data into 5 folds for cross-validation. − For each fold: ○ Train the MLP model with L2 regularisation. ○ Monitor the training loss during training and stop when the training loss does not improve by more than 0.0001 for 10 consecutive passes over the training set. ○ Evaluate the model's performance on the held-out fold using ROC-AUC. − Calculate the average performance metric (i.e., ROC-AUC) across 5 folds of the combination of hyperparameters. − Record and accumulate training time from the previous loop.
Step 3	Select Best Model: − Select the combination of hyperparameters that resulted in the highest average performance metric.

A Grid Search Loop with 5-fold Cross-Validation is implemented after that. An MLP model is instantiated and trained on five folds of the training data for each hyperparameter combination, with 10% reserved for validation. Training halts using ES when the validation score fails to improve by at least 0.0001 for ten consecutive epochs. Model performance is assessed using ROC-AUC on held-out folds, with the average metric calculated across all folds. Each loop records and accumulates training time from the previous loop. Finally, the hyperparameter combination yielding the highest average performance metric is selected, resulting in the best performing MLP model with optimised hyperparameters.

TABLE VII. ALGORITHM FOR HYPERPARAMETER TUNING WITH L2 REGULARISATION AND EARLY STOPPING IN MULTI-LAYER PERCEPTRON (MLP) USING GRID SEARCH

Input	: Training set
Output	: Best MLP model with optimized hyperparameters
Step 1	Define Hyperparameter Grid for MLP: − Set regularisation parameter λ for L2 regularisation to 0.0001. − Set number of hidden layers to 1. − Define a range for the number of neurons per hidden layer from 5 to 100. − Set activation function to ReLU. − Set solver to Adam. − Set batch size (minibatch) to the minimum of 200. − Set initial learning rate to 0.001. − Set maximum iterations to 5000. − Set shuffle sample to True. − Set exponential decay rate for estimates of the first moment vector in Adam to 0.999. − Set exponential decay rate for estimates of the second moment vector in Adam to 1e-8.
Step 2	Grid Search Loop with 5-Fold Cross-Validation: − For each combination of hyperparameters in the defined grid: − Initialize an MLP model with the specified architecture and hyperparameters. − Divide the training data into 5 folds for cross-validation. − For each fold: ○ Set aside 10% (using the stratified approach) of training data as validation. ○ Train the MLP model with L2 regularisation. ○ Terminate training using the Early Stopping approach, i.e. when the validation score is not improving by at least 0.0001 for 10 consecutive epochs. ○ Evaluate the model's performance on the held-out fold using ROC-AUC. − Calculate the average performance metric (i.e., ROC-AUC) across 5 folds of the combination of hyperparameters. − Record and accumulate training time from the previous loop.
Step 3	Select Best Model: − Select the combination of hyperparameters that resulted in the highest average performance metric.

The Cross-Entropy Loss function was employed for binary classification in the MLP model, as depicted in Eq. (1). This loss function quantifies the discrepancy between predicted and actual class labels, penalising misclassifications with logarithmic terms.

$$Loss(x_i, y_i) = -[y_i\, log(\hat{y}_i) + (1 - y_i)\, log(1 - \hat{y}_i)] \quad (1)$$

During the hyperparameter tuning phase using Grid Search, each cycle involves adjusting parameters such as the number of neurons (N) within a range from 5 to 100, alongside L2 regularisation. Eq. (2) represents the L2 regularisation term, penalising large weights to prevent overfitting.

$$n = \frac{\lambda}{2} \sum_{i=1}^{N} w_i^2 \quad (2)$$

Eq. (3) illustrates the cost function combining the Cross-Entropy Loss with the L2 regularisation term, where lambda (λ) is set to 0.0001. This cost function aims to balance minimising classification error and controlling model complexity, enhancing the model's predictive accuracy and robustness. For the innovative model, the cost function includes L2 regularisation hybridised with ES, terminating training when the validation set score does not improve by at least 0.0001 for ten consecutive epochs.

$$Cost_{reg}(x_i, y_i) = Loss(x_i, y_i) + \; n = \frac{\lambda}{2}\sum_{i=1}^{N} w_i^2 \qquad (3)$$

C. Testing

In this study, the development of MLP models was driven by two primary objectives: reducing the hyperparameter tuning time to identify the optimal set of hyperparameters for maximising classification accuracy and enhancing the model's ability to distinguish between classes. These objectives were evaluated using various metrics, including tuning time, confusion matrix analysis, and the ROC-AUC score. Tuning time refers to the duration from the commencement of the Grid Search process to its completion. It aims to enhance the model by pinpointing the most effective hyperparameters for achieving maximal accuracy in classification tasks. This step is critical for refining the model's configuration to improve its predictive performance. Furthermore, the evaluation process entailed an analysis of the confusion matrix to gauge the model's accuracy in classifying instances correctly across different categories. Additionally, the ROC-AUC score provided more profound insights into the model's discriminative capacity between classes.

Accuracy is a foundational metric, offering a high-level overview of the model's correctness. It is derived from the confusion matrix, as shown in Eq. (4), and calculates the proportion of instances correctly classified, irrespective of their class. This metric is handy when all classes are equally important, and the dataset is balanced. However, in cases of class imbalance, accuracy alone may not fully capture the model's effectiveness, as the prevalence of the dominant class may skew it. Hence, while accuracy is a valuable initial metric for evaluation, it is often essential to supplement it with additional metrics to obtain a more comprehensive picture of model performance, particularly in the presence of uneven class distribution.

$$Accuracy = \frac{(TP+TN)}{(TP+FP+TN+FN)} \qquad (4)$$

with *TP* is True Positive, *TN* is True Negative, *FP* is False Positive, and *FN* is False Negative.

Precision offers a more detailed examination of the model's performance, concentrating specifically on positive predictions. It is defined as the ratio of correctly predicted positive instances to the total number of predicted positive instances, as illustrated in Eq. (5). Precision is paramount in scenarios with significant consequences or costs associated with false positives.

$$Precision = \frac{(TP)}{(TP+FP)} \qquad (5)$$

with *TP* is True Positive and is False Positive.

In contrast, recall emphasises the model's capacity to identify all positive instances within the dataset. It calculates the ratio of correctly predicted positive instances to the total actual positive instances, highlighting its critical role in situations where overlooking positive instances (false negatives), could have serious repercussions. Eq. (6) presents the formula for calculating recall derived from the confusion matrix.

$$Recall/Sensitivity = \frac{(TP)}{(TP+FN)} \qquad (6)$$

with *TP* is True Positive and is False Negative.

The F1 score, as presented in Eq. (7), amalgamates precision, and recall into a singular metric, offering a balanced evaluation of the model's performance. It computes the harmonic mean of precision and recall, providing a unified measure that accounts for false positives and false negatives, thereby preventing inflated scores when a significant discrepancy exists between precision and recall. The F1 score is particularly beneficial in scenarios involving class imbalance or when the costs associated with false positives and false negatives vary. By integrating precision and recall, the F1 score aids in guiding decision-making and model optimisation, ensuring a thorough classification performance assessment.

$$F1\,Score = \; 2 \times \frac{Precision \times Recall}{Precision + Recall} \qquad (7)$$

The ROC-AUC score is widely recognised as a crucial metric for evaluating the performance of binary classification models. It depicts the likelihood that a randomly selected positive instance will be ranked higher than a randomly selected negative instance. The ROC curve plots the true positive rate (sensitivity) against the false positive rate (1-specificity) across various threshold settings. A higher ROC-AUC score signifies superior discrimination between positive and negative classes, with a score of 1 denoting a perfect classifier and 0.5 indicating a model no better than random guessing. The ROC-AUC score's robustness to class imbalance and applicability across different threshold configurations make it an invaluable tool for evaluating binary classifiers' overall performance in practical scenarios.

III. Result and Discussion

Table VIII illustrates the performance comparison of the MLP model with and without the employment of L2 regularisation. Implementing L2 regularisation with a 0.05 alpha value in the MLP model significantly enhanced its performance compared to the model devoid of L2 regularisation. This enhancement was particularly evident across three banking datasets, where the influence of L2 regularisation on the MLP model's performance was substantial. For instance, in a loan dataset comprising 16 features, the ROC-AUC score increased from 0.9133 to 0.9244. Similarly, for a loan dataset with 20 features, the ROC-AUC score improved from 0.9285 to 0.9312, and for a banking dataset focusing on credit card applications, the ROC-AUC score rose from 0.7464 to 0.7696. As observed in this study, the emphasis on the ROC-AUC metric in the analysis underscores its utility in providing a nuanced interpretation of model performance, especially in datasets with imbalanced labels. Additionally, results derived from the confusion matrix indicated an improvement in the MLP model's performance. These findings corroborate the significance of L2 regularisation in mitigating overfitting when developing MLP models, aligning with insights from prior research.

TABLE VIII. COMPARATIVE PERFORMANCE OF MLP MODELS WITH AND WITHOUT L2 REGULARISATION ACROSS BANKING DATASETS

Performance Matrices	Bank Loan 16 features		Bank Loan 20 features		Bank credit card default payments	
	No L2	L2	No L2	L2	No L2	L2
ROC-AUC	0.9133	0.9244	0.9285	0.9312	0.7464	0.7696
Accuracy	0.8979	0.9027	0.9052	0.9048	0.8021	0.8145
Precision (Positive)	0.5884	0.6373	0.5812	0.5839	0.5618	0.6072
Recall (Positive)	0.4384	0.4017	0.5375	0.5114	0.3855	0.4017
F1 Score	0.5024	0.4928	0.5585	0.5452	0.4572	0.4835

Table IX showcases the impact of integrating L2 regularisation into a MLP model, specifically employing a hyperparameter value setting as outlined in Table IV for the hyperparameter tuning process. This model exhibited remarkable performance, consistent with earlier observations that L2 regularisation could bolster MLP model performance. Specifically, Table IX outlines the results concerning ROC-AUC and the confusion matrix metrics. The bank loan dataset, consisting of 16 features, achieved ROC-AUC of 0.9221. A separate dataset, also related to bank loans but encompassing 20 features, registered ROC-AUC of 0.9455, while the bank credit card dataset demonstrated the ROC-AUC of 0.7814.

TABLE IX. IMPACT OF L2 REGULARISATION ON HYPERPARAMETER TUNING TIME ACROSS BANKING DATASETS

Dataset	Tuning Time (Sec.)	ROC-AUC	Accuracy	F1 Score
Bank Loan 16 features	26,787.75	0.9221	0.9005	0.5233
Bank Loan 20 features	23,714.07	0.9455	0.9189	0.5917
Bank credit card default payments	23,087.06	0.7814	0.8260	0.4847

The application of L2 regularisation has proven to enhance the MLP model's efficacy in classification tasks. Nonetheless, identifying the optimal hyperparameter settings with L2 regularisation application has been revealed to be a time-intensive endeavour. The hyperparameter tuning phase for all three datasets necessitated upwards of twenty-three thousand seconds, raising concerns regarding the practicality of deploying such models in real-time applications that demand frequent updates with new data entries.

Table X displays the performance of the MLP model employing a hybrid L2 regularisation and Early Stopping (L2ES) approach. Both techniques aim to mitigate overfitting, yet the experimental results indicate a significant reduction in hyperparameter tuning time with the hybrid L2ES approach. For the bank loan dataset with 16 features, tuning time was reduced by 91.44%, from 26,787.75 seconds to 2,424.88 seconds. Similarly, the dataset with 20 features saw a reduction of 90.95%, from 23,714.07 seconds to 2,030.71 seconds. The bank credit card dataset experienced a tuning time reduction of 93.57%, from 23,087.06 seconds to 1,485.15 seconds. The efficacy of the hybrid L2 and ES approach lies in its ability to shorten hyperparameter tuning time, primarily through the advantage offered by ES. This feature ceases the training process upon the absence of a minimum improvement of 0.0001 in the validation set score over ten successive epochs. This method is in stark contrast to the termination criteria of the baseline model, which discontinues training based on the absence of a notable improvement in the training loss over ten successive iterations, without consideration for the validation set score. Concurrently, L2 regularization is crucial in maintaining the classification model's accuracy by averting overfitting.

TABLE X. IMPACT OF HYBRID L2 REGULARISATION AND EARLY STOPPING ON HYPERPARAMETER TUNING TIME ACROSS BANKING DATASETS

Dataset	Tuning Time (Sec.)	ROC-AUC	Accuracy	F1 Score
Bank Loan 16 features	2,424.88	0.9121	0.8966	0.5351
Bank Loan 20 features	2,030.71	0.9389	0.9137	0.5854
Bank credit card default payments	1,485.15	0.7328	0.7877	0.4323

Fig. 2 presents a comparative analysis of hyperparameter tuning time between the MLP model using L2 regularisation and the proposed hybrid L2ES approach. The results are visually displayed using a bar chart, clearly illustrating a substantial reduction in tuning time across all datasets. Specifically, the hybrid L2ES method reduced tuning time by more than 90% compared to the L2-only approach, which was used as the baseline in this study. This represents a notable improvement, as such a reduction directly contributes to the practicality of implementing data-driven classification (DDC) models in real-world banking systems, particularly in scenarios requiring rapid model updates and continuous learning.

Fig. 3 compares the classification performance of the two approaches regarding ROC-AUC scores. The performance gap between the L2 and L2ES models is minimal, indicating that the significant gains in computational efficiency do not come at the cost of model accuracy. For the bank loan dataset with 16 features, the ROC-AUC difference is only 0.70%, while the 20-feature loan dataset shows a difference of 1.08%. The bank credit card dataset exhibits a slightly larger difference of 6.22%, which remains within an acceptable range of less than 10%. Although the L2ES approach yields marginally lower ROC-AUC scores in some cases, it consistently outperforms previous studies in overall performance benchmarks, particularly in hyperparameter tuning efficiency. These findings support the feasibility and value of the proposed hybrid approach for time-sensitive banking applications.

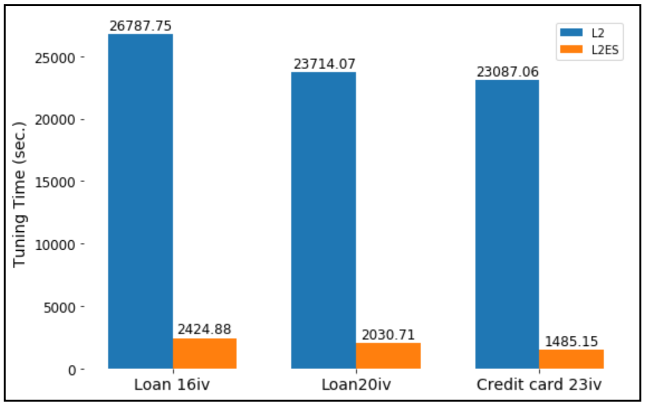

Fig. 2. Comparison of hyperparameter tuning time: L2 regularisation vs hybrid L2ES approach,

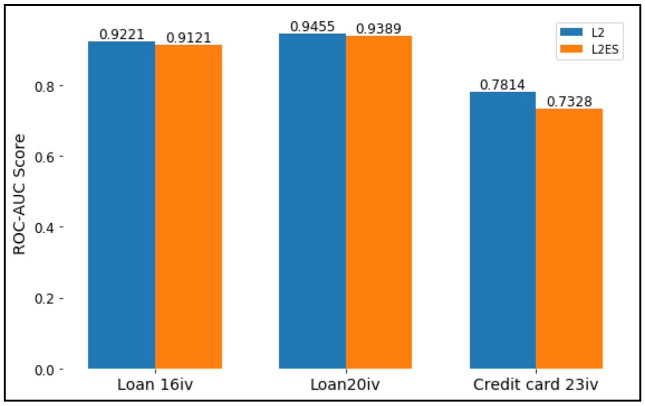

Fig. 3. ROC-AUC performance comparison: L2 regularisation vs hybrid L2ES approach across datasets.

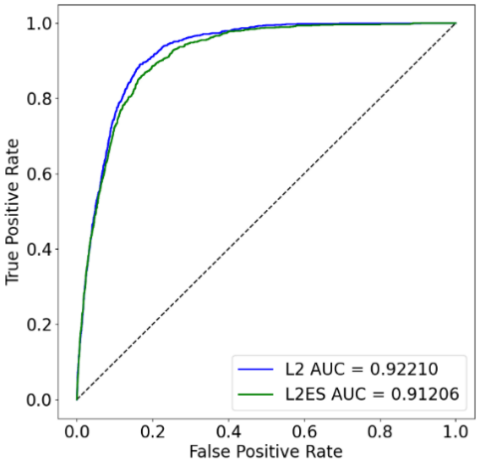

Fig. 4. ROC-AUC comparison for bank loan dataset with 16 features: L2 vs. hybrid L2ES approach.

Fig. 4 and Fig. 5 present the ROC-AUC curves for the bank loan datasets with 16 and 20 features, respectively, providing an alternative visual comparison between the L2 and hybrid L2ES approaches. Both graphs clearly show that the performance of the MLP model under the two conditions is very similar, with no significant differences observed in the shape or position of the curves.

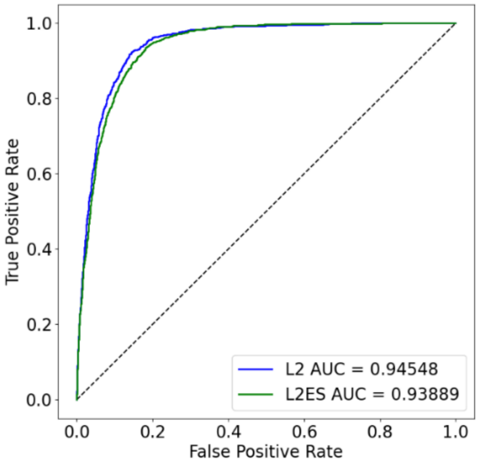

Fig. 5. ROC-AUC comparison for bank loan dataset with 20 features: L2 vs. hybrid L2ES approach.

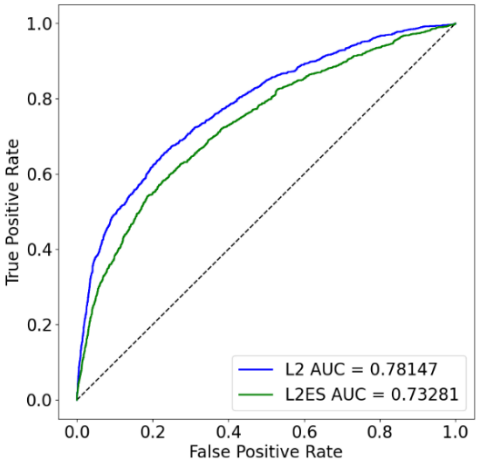

Fig. 6. ROC-AUC comparison for bank credit card dataset with 23 features: L2 vs. hybrid L2ES approach.

Fig. 6 displays the ROC-AUC curve for the bank credit card dataset. In this case, the hybrid L2ES approach shows a slightly lower curve than the model using L2 regularisation alone, indicating a marginal reduction in performance. However, this difference is relatively minor and does not significantly affect the overall effectiveness of the hybrid method. Notably, the L2ES approach outperforms previous studies, achieving a ROC-AUC score of 0.7328, compared to the values reported in Table II, where previous models did not exceed a ROC-AUC of 0.7200.

IV. Conclusion

This study evaluated the effectiveness of hybridising ES with L2 regularisation to reduce the time required for hyperparameter tuning—conducted via grid search—in developing MLP models, while preserving classification performance. The proposed L2ES approach significantly reduced tuning time without compromising model accuracy, as demonstrated by consistently high ROC-AUC scores comparable to those obtained using L2 regularisation alone. These results indicate that the L2ES technique offers a practical solution for streamlining hyperparameter tuning in

MLP models, thereby enhancing the efficiency of developing data-driven classification models.

Nevertheless, this study is not without limitations. The evaluation was limited to a set of structured banking datasets, and further validation is needed across diverse data domains and model architectures. Additionally, while grid search was used for tuning, other optimisation methods such as random search or Bayesian optimisation could offer further improvements.

Future research will aim to extend the proposed hybrid approach to other neural network models and investigate its compatibility with alternative tuning strategies. Enhancing generalisability, automating the hybridisation process, and applying it to unstructured or real-time financial data are promising directions for continued research.

ACKNOWLEDGMENT

The authors gratefully acknowledge the financial support from Universiti Teknikal Malaysia Melaka (UTeM) and the voluntary collaboration between research teams at UTeM and Universiti Teknologi MARA (UiTM), driven by a shared commitment to advancing data-driven classification methods.

REFERENCES

[1] A. AbdElsalam, M. Abdallah and H. Refaat, "Predicting Human Essential Genes Using Deep Learning: MLP with Adaptive Data Balancing" International Journal of Advanced Computer Science and Applications, vol. 16, no. 4, 2025.

[2] A. Aldelemy, and R. A. Abd-Alhameed, "Binary Classification of Customer's Online Purchasing Behavior Using Machine Learning." Journal of Techniques, vol. 5, no. 2, pp. 163-86, 2023.

[3] M. Grosicki. "Application of artificial neural networks to prediction of success of bank telemarketing campaign." 11-th European Conference of Young Researchers and Scientists, no. 3, pp. 6-8. 2015.

[4] A. Vitório and G. Marques. "Impact of imbalanced data on bank telemarketing calls outcome forecasting using machine learning." IEEE 2021 International Conference on Data Analytics for Business and Industry (ICDABI), pp. 380-384, 2021.

[5] T. C. Hsu, S. T. Liou, Y. P. Wang, and Y. S. Huang. "Enhanced recurrent neural network for combining static and dynamic features for credit card default prediction." ICASSP 2019 IEEE International Conference on Acoustics, pp. 1572-1576, 2019.

[6] A. Arram, M. Ayob, M. A. A. Albadr, A. Sulaiman, and D. Albashish. "Credit card score prediction using machine learning models: A new dataset." arXiv preprint arXiv:2310.02956, 2023.

[7] S. Moro, P. Cortez, and P. Rita. "A data-driven approach to predict the success of bank telemarketing." Decision Support Systems, vol. 62, 22-31, 2014.

[8] G. Marinakos, and S. Daskalaki. "Imbalanced customer classification for bank direct marketing." Journal of Marketing Analytics, no. 5, pp. 14-30, 2017.

[9] R. Farooqi and N. Iqbal. "Performance evaluation for competency of bank telemarketing prediction using data mining techniques." International Journal of Recent Technology and Engineering, vol. 8, no. 2, pp. 5666-5674, 2019.

[10] N. Ghatasheh, H. Faris, I. AlTaharwa, Y. Harb, and A. Harb. "Business analytics in telemarketing: Cost-sensitive analysis of bank campaigns using artificial neural networks." Applied Sciences, vol. 10, no. 7, pp. 2581, 2020.

[11] A. Panigrahi, and M. C. Patnaik. "Customer deposit prediction using neural network techniques." International Journal of Applied Engineering Research, vol. 15, no. 2, pp. 253-258, 2020.

[12] S. Mokrane. "Predicting the success of bank telemarketing using Artificial Neural Network." International Journal of Economics and Management Engineering, vol. 14, no. 1, pp. 1-4, 2020.

[13] S. Dutta and S. K. Bandyopadhyay. "Recommender System for Term Deposit Likelihood Prediction Using Cross-Validated Neural Network." South Asian Journal of Social Studies and Economics, vol. 11, no. 3, pp. 21-28, 2021.

[14] S. Masturoh, F. S. Nugraha, S. Nurlela, M. R. Ramadhan Saelan, D. U. Eka Saputri, and R. Nurfalah. "Telemarketing Bank Success Prediction Using Multilayer Perceptron (MLP) Algorithm with Resampling." Jurnal Pilar Nusa Mandiri, vol. 17, no. 1, pp. 19-24, 2021.

[15] R. Singh and R. R. Aggarwal. "Comparative evaluation of predictive modeling techniques on credit card data." International Journal of Computer Theory and Engineering, vol. 3, no. 5, pp. 598, 2021.

[16] R. Liu. "Machine learning approaches to predict default of credit card clients." Modern Economy, vol. 9, no. 11, pp. 1828-1838, 2018.

[17] Vishwakarma, S. Kumar, A. Rasool, and G. Hajela. "Machine Learning Algorithms for Prediction of Credit Card Defaulters—A Comparative Study." Proceedings of International Conference on Sustainable Expert Systems Expert Systems: ICSES 2020, pp. 141–149, 2021.

[18] S. de Campos, P. Vitor, and L. C. Bambirra Torres. "Extreme wavelet fast learning machine for evaluation of the default profile on financial transactions." Computational Economics, vol. 57, no. 4, pp. 1263-1285, 2021.

[19] A. Almajid. "Multilayer Perceptron Optimization on Imbalanced Data Using SVM-SMOTE and One-Hot Encoding for Credit Card Default Prediction." Journal of Advances in Information Systems and Technology, vol. 3, no. 2, pp. 67-74, 2022.

[20] J. Jiang, X. Meng, Y. Liu, and H. Wang. "An enhanced TSA-MLP model for identifying credit default problems." SAGE Open, vol. 12, no. 2, 2022.

[21] K. Shazly, and N. Khodadadi. "Credit card clients classification using hybrid guided wheel with particle swarm optimized for voting ensemble." Journal of Artificial Intelligence and Metaheuristics vol. 2, pp. 46-54, 2023.

[22] M. Q. Idrees, H. Naeem, M. Imran, A. Batool, & N. Tabassum. "Identifying Optimal Parameters and their Impact for Predicting Credit Card Defaulters Using Machine-Learning Algorithms." Lahore Garrison University Research Journal of Computer Science and Information Technology, vol. 6, no. 1, pp. 1-21, 2022.

[23] H. Yash, Affan, K. Saurav, and S. S. Dhanda. "Credit Card Default Prediction Using Machine Learning Models." Proc. - 2023 3rd Int. Conf. Innov. Sustain. Comput. Technol. CISCT 2023, pp. 1–5, 2023.

[24] S. Moro, R. Laureano, and P. Cortez. "Using data mining for bank direct marketing: An application of the crisp-dm methodology." The 2011 European Simulation and Modelling Conference, pp. 117–121, 2011.

[25] I-C. Yeh, and C-H. Lien. "The comparisons of data mining techniques for the predictive accuracy of probability of default of credit card clients." Expert systems with applications, vol. 36, no. 2, pp. 2473-2480, 2009.