Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

CHIMERA-Tab: A Scalable AI-Driven Framework for Automated Feature Selection and Heterogeneous Stacking in Imbalanced Banking Data Classification

Anilkumar Vishwanath Brahmane
Sanjivani College of Engineering
P. William
Karunya University
Shashikant Raghunathrao Deshmukh
Sanjivani College of Engineering
Elviz Ismayilov
Azerbaijan State Oil and Industry University
Smita Nirkhi
smita.singh@sitnagpur.siu.edu.in

Symbiosis International University

Article

Keywords: Chaotic metaheuristics, Ensemble stacking, Feature selection, Gradient boosting, Imbalanced classification, Tabular deep learning

Posted Date: August 28th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-10401748/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

CHIMERA-Tab: A Scalable AI-Driven Framework for Automated Feature Selection and Heterogeneous Stacking in Imbalanced Banking Data Classification

Anilkumar Vishwanath Brahmane[1], P. William[2,3*], Shashikant Raghunathrao Deshmukh[4], Elviz Ismayilov[5], Smita Nirkhi[6*]

[1]Department of Computer Engineering, Sanjivani College of Engineering, Kopargaon, Maharashtra, India.
[2]School of Computer Science and Technology, Karunya Institute of Technology and Sciences (Deemed University), Coimbatore, India.
[3]Centre for Research and Consultancy, UNITAR International University, Selangor, Malaysia.
[4]Department of Computer Engineering, Sanjivani College of Engineering, Kopargaon, Maharashtra, India.
[5]Azerbaijan State Oil and Industry University, Baku, Azerbaijan.
[6]Department of Computer Engineering, Symbiosis International (Deemed University), Pune, India.

*Corresponding author(s). E-mail(s): william160891@gmail.com; smita.singh@sitnagpur.siu.edu.in;
Contributing authors: brahmaneanilkumarcomp@sanjivani.org.in; shashikant.deshmukh1000@gmail.com; elviz.ismayilov@asoiu.edu.az;

Abstract

Imbalanced tabular classification remains challenging in financial decision support, where heterogeneous feature spaces and skewed class distributions limit the effectiveness of individual classifiers. This paper introduces CHIMERA-Tab, a heterogeneous stacking framework that combines a Residual-Augmented TabNet with LightGBM, XGBoost, and CatBoost through a logistic regression

meta-learner that learns optimal combination weights from validation-set predictions. The framework incorporates a chaotic metaheuristic preprocessing pipeline consisting of a Chaotic Binary Aquila Optimizer for wrapper-based feature selection (reducing 21 features to 8, a 61.9% reduction) and a Chaotic Sine-Cosine Algorithm for TabNet hyperparameter optimisation, both guided by an entropy-based ensemble of logistic, tent, sine, and Chebyshev chaotic maps. On the UCI Bank Marketing dataset (41,188 samples, 7.9:1 class imbalance), CHIMERA-Tab achieves a mean AUC-ROC of 0.9507 $\pm$ 0.0026 and macro F1-score of 0.7949 $\pm$ 0.0056 across five independent runs, significantly outperforming eight of nine baselines via Wilcoxon signed-rank tests ($\boldsymbol{p = 0.0313}$). A seven-variant ablation study with three-seed aggregation reveals that the heterogeneous stacking architecture is the primary performance driver (-0.71 percentage points AUC upon removal), while chaotic preprocessing provides complementary benefits in dimensionality reduction and interpretability rather than classification accuracy. SHAP analysis, chaotic sensitivity profiling, and without-duration evaluation provide multi-perspective interpretability and deployment-realistic assessment.

Keywords: Chaotic metaheuristics, Ensemble stacking, Feature selection, Gradient boosting, Imbalanced classification, Tabular deep learning

1 Introduction

Financial institutions across the globe continue to invest heavily in direct marketing campaigns as a means of acquiring new customers for banking products such as term deposits, insurance policies, and credit instruments. Telemarketing—the practice of contacting prospective clients through outbound telephone calls—remains one of the most widely used channels for customer acquisition despite the proliferation of digital alternatives [1]. A successful telemarketing campaign must balance the tension between maximising client reach and minimising the operational cost of unsuccessful contacts, a trade-off that fundamentally depends on the accuracy of predictive models used to pre-screen candidate clients [2]. The UCI Bank Marketing dataset, compiled from the telemarketing records of a Portuguese banking institution between May 2008 and November 2010, has become the canonical benchmark for evaluating such predictive models [1]. Containing 41,188 client interaction records across 20 input features, the dataset captures a rich mix of client demographics, campaign characteristics, previous contact history, and socioeconomic context variables that collectively influence subscription decisions.

The specific prediction task—determining whether a client will subscribe to a term deposit—presents several interrelated technical challenges. The target variable exhibits pronounced class imbalance, with a positive subscription rate of approximately 11.3% (a roughly 7.9:1 ratio), which causes standard classifiers to favour the majority class [3, 4]. The feature space mixes numerical indicators with categorical descriptors, requiring careful encoding. Call duration is a post-hoc variable that introduces data leakage in deployment scenarios. Socioeconomic covariates encode macroeconomic conditions whose influence shifts across economic cycles.

Existing approaches span classical statistical models to deep learning architectures. Logistic regression, decision trees, and SVMs achieved AUC-ROC values of 0.80–0.93 [1]. Xie et al. [2] used Random Subspace and Multi-Boosting with SHAP interpretability. Zhang et al. [5] demonstrated that heterogeneous stacking ensembles outperform homogeneous ones. Gradient boosting frameworks—XGBoost [6], LightGBM [7], and CatBoost [8]—remain consistently competitive [9, 10]. Deep tabular architectures including TabNet [11], TabTransformer [12], and NODE [13] introduce differentiable feature selection but have not consistently surpassed gradient boosting [14, 15].

Despite these advances, four limitations persist. First, heterogeneous stacking ensembles that combine deep tabular architectures (such as TabNet) with gradient boosting models through learned meta-learner weights remain underexplored—most comparative studies evaluate these model families in isolation rather than combining their complementary strengths [5, 9]. Second, feature selection for tabular pipelines is often disconnected from the downstream classifier, and wrapper-based approaches using metaheuristic search [16, 17] have rarely been integrated into unified classification pipelines. Third, hyperparameter optimisation for architectures such as TabNet typically relies on grid or random search, while chaotic maps—which produce deterministic yet aperiodic sequences with superior space coverage [18–20]—remain unexploited for this purpose. Fourth, statistical validation in many bank marketing studies lacks rigour, reporting single-run accuracy without multi-seed evaluation, formal hypothesis testing, or honest ablation analysis that identifies which components actually drive performance.

To address these gaps, this paper proposes CHIMERA-Tab (Chaotic Hybrid Intelligent Metaheuristic Ensemble with Residual-Augmented Tabular Learning), which chains three chaotically enhanced stages: feature selection via the Aquila Optimizer [21], hyperparameter optimisation via the Sine-Cosine Algorithm [22], and residual-augmented stacking ensemble prediction. The design philosophy follows the No Free Lunch theorem [23], maintaining an ensemble of chaotic generators with entropy-based selection.

The major contributions are: (1) A heterogeneous four-model stacking ensemble that combines a Residual-Augmented TabNet (RA-TabNet) with LightGBM, XGBoost, and CatBoost through a logistic regression meta-learner, demonstrating that model diversity across deep tabular and tree-based families provides the primary performance advantage (+0.71 percentage points AUC over standalone TabNet); (2) A Chaotic Binary Aquila Optimizer for Feature Selection (CBAOFS) with entropy-guided four-map chaotic ensemble and V-shaped transfer function, reducing features from 21 to 8 (61.9% reduction) for improved interpretability and computational efficiency; (3) A Chaotic Sine-Cosine Algorithm (CSCA) for TabNet hyperparameter optimisation with chaotically modulated search dynamics over 30 evaluations; (4) An entropy-guided chaotic map engine that adaptively selects among logistic, tent, sine, and Chebyshev generators based on Shannon entropy monitoring; and (5) A comprehensive evaluation methodology with five-seed multi-run evaluation, Friedman and Wilcoxon testing, Cohen's d effect sizes, bootstrap confidence intervals, a seven-variant

ablation study that transparently identifies the contribution hierarchy, scalability analysis, and deployment-realistic without-duration assessment.

The paper is organised as follows: Section 2 reviews related work. Section 3 describes the methodology. Section 4 presents experimental setup. Section 5 analyses results. Section 6 discusses findings. Section 7 concludes.

2 Related Work

This section reviews prior research across four thematic areas: deep learning for tabular data (Section 2.1), metaheuristic-based feature selection (Section 2.2), chaotic optimisation strategies (Section 2.3), and bank marketing prediction (Section 2.4).

2.1 Deep Learning for Tabular Data

Arik and Pfister [11] introduced TabNet, which employs sequential attention through sparse feature selection masks at each decision step. Gorishniy et al. [14] found that a properly configured ResNet baseline often matched more complex architectures. Grinsztajn et al. [9] demonstrated that tree-based ensembles maintain a consistent advantage on typical tabular datasets. Borisov et al. [15] surveyed over 40 architectures and found no single dominant approach. Shwartz-Ziv and Armon [10] showed that well-tuned XGBoost outperformed several deep tabular models.

2.2 Metaheuristic-Based Feature Selection

The Aquila Optimizer [21] simulates four hunting behaviours for exploration–exploitation balance. Nadimi-Shahraki et al. [17] proposed the first binary AO using S/V-shaped transfer functions. Pashaei [24] introduced mutation-based binary AO. Abd El-Mageed et al. [25] proposed an improved binary AO with local search. Gopi and Mohapatra [20] integrated chaotic maps into the AO's parameters.

2.3 Chaotic Optimisation Strategies

Chaotic maps exhibit ergodicity, aperiodicity, and sensitivity to initial conditions [18]. Altay and Alatas [19] embedded ten chaotic maps into TSO. Abdel-Basset et al. [16] proposed a chaotic Dwarf Mongoose Optimizer. The SCA [22] uses oscillatory sine/-cosine functions, with Abbassi et al. [26] proposing a chaotic variant. A notable gap is the absence of adaptive entropy-based selection among multiple maps during search.

2.4 Bank Marketing Prediction

Moro et al. [1] established the benchmark. Xie et al. [2] used RS-MB methods with SHAP. Zhang et al. [5] demonstrated stacking ensemble superiority. Kaisar et al. [27] explored online ensemble learning.

Table 1 provides a systematic summary of the most relevant prior works across the four thematic areas reviewed above. Each row identifies the reference, the methods employed, the key advantages demonstrated, and the specific limitations that

remain unaddressed. The table reveals a consistent pattern: while individual algorithmic innovations have advanced the state of the art in their respective domains (chaotic optimisation, binary feature selection, deep tabular learning, bank marketing prediction), no prior work has integrated these components into a unified pipeline with adaptive chaotic map selection and statistical validation at the depth provided by CHIMERA-Tab.

2.5 Research Gaps and Positioning

The preceding review reveals five specific research gaps that the present work aims to address. Table 2 summarises these gaps, identifies the affected prior works, and describes how CHIMERA-Tab provides solutions for each one. Gap G1 targets the underexploration of heterogeneous stacking ensembles combining deep tabular and tree-based models with learned meta-learner weights: most prior binary AO variants [17, 25] evaluate candidate feature subsets using a fixed simple classifier (e.g., KNN) rather than the actual downstream model, leading to suboptimal feature subsets. Gap G2 addresses the static use of a single chaotic map throughout the optimisation process [19, 20], which may not match the diversity requirements at different optimisation stages. Gap G3 highlights that hyperparameter optimisation for deep tabular models such as TabNet [11] typically relies on grid or random search rather than metaheuristic search. Gap G4 notes that heterogeneous stacking ensembles combining deep tabular and tree-based models with meta-learning remain underexplored [5, 9]. Gap G5 identifies insufficient statistical validation in bank marketing studies, where single-run accuracy figures are reported without multi-seed evaluation or formal hypothesis testing [1, 2]. CHIMERA-Tab addresses these gaps through its five contributions: G1→CBAOFS, G2→entropy-guided chaotic engine, G3→CSCA, G4→stacking ensemble, G5→comprehensive evaluation.

3 Proposed CHIMERA-Tab Framework

3.1 Problem Formulation

Consider a binary classification dataset $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$ where $\mathbf{x}_i \in \mathbb{R}^d$ with $d = 21$ features, $y_i \in \{0, 1\}$, and $N = 41{,}188$. The positive class prevalence is $\pi_+ = 0.1127$ with imbalance ratio $\rho \approx 7.9$.

The framework decomposes prediction into three sequential subproblems. The first subproblem seeks a binary feature mask $\mathbf{m}^* \in \{0, 1\}^d$ minimising the fitness function defined in Eq. (1). This equation balances classification error against model parsimony: the first term ($\alpha = 0.9$) penalises low AUC-ROC from 3-fold cross-validated LightGBM, while the second term ($\beta = 0.1$) discourages solutions retaining more than 80% of features. The dominance of the classification term ensures that accuracy drives the search, with parsimony serving only as a tiebreaker among equally accurate subsets.

$$\mathcal{F}(\mathbf{m}) = \alpha \cdot (1 - \mathrm{AUC}_{\mathrm{CV}}(\mathbf{m})) + \beta \cdot 5 \cdot \max\left(0, \frac{\|\mathbf{m}\|_0}{d} - 0.8\right) \tag{1}$$

Table 1 Comprehensive Literature Review Summary (2016–2025)

Reference	Methods	Advantages	Limitations
Arik & Pfister [11]	TabNet with sequential attention	Instance-level interpretability; sparse feature selection; end-to-end learning	Sensitive to hyperparameters; inconsistent vs. tree models
Gorishniy et al. [14]	ResNet, FT-Transformer, MLP	Systematic comparison; well-tuned baselines	No metaheuristic optimisation
Grinsztajn et al. [9]	Tree ensembles vs. DL on 45 datasets	Large-scale evidence; rotation invariance identified	No hybrid tree-neural approaches
Shwartz-Ziv & Armon [10]	XGBoost vs. deep models	Demonstrates tree-model competitiveness	No ensemble combination strategies
Abualigah et al. [21]	Aquila Optimizer (4 phases)	Four-phase exploration-exploitation balance	Continuous-space only; no binary variant
Nadimi-Shahraki et al. [17]	Binary AO with S/V transfer	First binary AO; 7 medical datasets	No chaotic enhancement; KNN only
Abd El-Mageed et al. [25]	Improved binary AO + local search	Enhanced convergence; high-dim data	No chaotic adaptation
Gopi & Mohapatra [20]	Chaotic AO for engineering	Chaotic maps improve convergence	Single map; no adaptive selection
Pashaei [24]	Mutation-based binary AO	TVMS transfer; mutation exploration	No chaotic maps; gene data only
Mirjalili [22]	Sine Cosine Algorithm	Simple oscillatory update rule	Random parameters; no chaos
Abbassi et al. [26]	Chaotic SCA variant	Chaotic modulation improves convergence	Single map; benchmarks only
Altay & Alatas [19]	10 chaotic maps in TSO	Systematic map comparison	No adaptive selection
Li et al. [28]	Chaotic MGO for medical FS	ICMIC init.; nonlinear control	Single map; no DL integration
Abdel-Basset et al. [16]	Chaotic Dwarf Mongoose for FS	Chaotic init. diversity	Single map; no deep learning
Moro et al. [1]	LR, DT, SVM, NN	Establishes benchmark; key predictors	No ensembles; limited tuning
Xie et al. [2]	RS, MB, RS-MB + SHAP	Hybrid ensemble; interpretability	No deep learning; single-run
Zhang et al. [5]	Multi-classifier stacking	Heterogeneous ensemble superiority	No FS optimisation; no chaos
Kaisar et al. [27]	Online ensemble learning	Temporal dynamics; online adaptation	No deep tabular; no FS
Sasmal et al. [29]	AO survey (100+ papers)	Comprehensive domain mapping	Survey scope only

Table 2 Summary of Research Gaps and How the Present Work Addresses Them

Gap	Identified Research Gap	Affected Prior Works	How This Work Addresses the Gap
G1	Feature selection disconnected from downstream classifier; single-map or non-chaotic metaheuristics	Nadimi-Shahraki et al. [17], Abd El-Mageed et al. [25], Abdel-Basset et al. [16]	CBAOFS: entropy-guided four-map chaotic ensemble with LightGBM wrapper fitness
G2	Static single chaotic map usage without adaptive selection	Gopi and Mohapatra [20], Altay and Alatas [19]	Shannon entropy monitoring with round-robin tiebreaking selects highest-coverage map each iteration
G3	HP optimisation for deep tabular models relies on grid/random search	Arik and Pfister [11], Gorishniy et al. [14]	CSCA optimises 8 TabNet HPs via chaotically modulated sine-cosine dynamics
G4	Heterogeneous deep tabular + tree stacking with meta-learning under-explored	Grinsztajn et al. [9], Zhang et al. [5]	Four-model stacking (RA-TabNet + LightGBM + XGBoost + CatBoost) with logistic regression meta-learner
G5	Insufficient statistical validation in bank marketing studies	Moro et al. [1], Xie et al. [2]	Five-seed evaluation with Friedman, Wilcoxon, Cohen's d, bootstrap CIs, 7-variant ablation with 3-seed aggregation

The second subproblem optimises eight TabNet hyperparameters $\boldsymbol{\theta}^* \in \Theta \subset \mathbb{R}^8$ using the composite objective in Eq. (2). This equation assigns 60% weight to AUC-ROC and 40% weight to macro F1-score. The inclusion of macro F1 prevents the optimiser from selecting configurations that achieve high AUC through excellent majority-class discrimination while sacrificing minority-class recall—a common failure mode with imbalanced data. Without the F1 component, the optimiser could converge to a trivially conservative model that predicts the majority class with high confidence.

$$\mathcal{G}(\boldsymbol{\theta}) = 0.6 \cdot \mathrm{AUC}_{\mathrm{val}}(\boldsymbol{\theta}) + 0.4 \cdot \mathrm{F1}_{\mathrm{macro,val}}(\boldsymbol{\theta}) \tag{2}$$

The third subproblem learns stacking weights $\mathbf{w}^* \in \mathbb{R}^K$ for $K = 4$ base models via the regularised logistic loss in Eq. (3). In this equation, $\mathbf{p}_j$ contains the validation-set probability predictions from each base model, $\sigma(\cdot)$ is the sigmoid function, b is the bias, and $C = 1.0$ controls L2 regularisation strength. The logistic formulation enables the meta-learner to assign different weights to each base model based on its marginal

contribution to the ensemble, accounting for both individual accuracy and inter-model correlation. The L2 penalty prevents any single model from dominating the ensemble excessively.

$$\mathbf{w}^* = \arg\min_{\mathbf{w}} -\sum_{j=1}^{n_{\text{val}}} \Big[y_j \log \sigma(\mathbf{w}^T \mathbf{p}_j + b) + (1 - y_j) \log(1 - \sigma(\mathbf{w}^T \mathbf{p}_j + b)) \Big] + \frac{\|\mathbf{w}\|_2^2}{2C} \quad (3)$$

3.2 Framework Overview

Fig. 1 presents the end-to-end architecture of the CHIMERA-Tab framework. The diagram illustrates the sequential data flow through five stages: (i) raw data preprocessing including label encoding and standard scaling, (ii) stratified 70/15/15 splitting, (iii) chaotic feature selection via CBAOFS which reduces 21 features to 8 and outputs the optimal binary mask $\mathbf{m}^*$, (iv) chaotic hyperparameter optimisation via CSCA which tunes 8 TabNet parameters and outputs $\boldsymbol{\theta}^*$, and (v) the four-model stacking ensemble where RA-TabNet, LightGBM, XGBoost, and CatBoost generate validation-set predictions that are combined by a logistic regression meta-learner with learned weights $\mathbf{w} \in \mathbb{R}^4$. The chaotic engine (orange box) feeds entropy-guided sequences to both metaheuristic modules. Dashed arrows indicate that the same feature mask is applied to all base models, ensuring a fair comparison of modelling strategies.

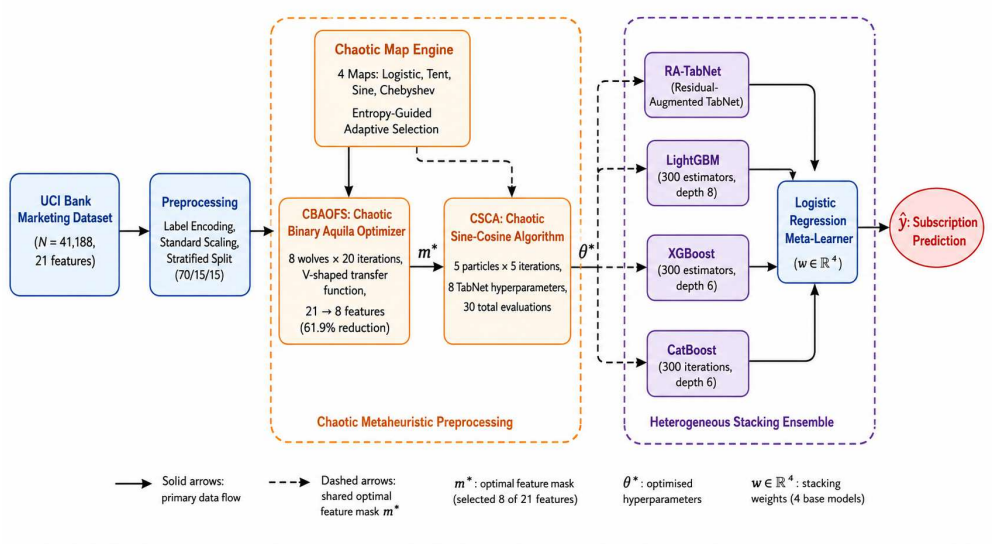


Fig. 1 Architecture of the CHIMERA-Tab framework showing the sequential pipeline from data preprocessing through chaotic metaheuristic preprocessing (CBAOFS and CSCA) to the heterogeneous stacking ensemble with logistic regression meta-learner. Solid arrows indicate primary data flow; dashed arrows indicate shared optimal feature mask $\mathbf{m}^*$ applied to all base models.

As shown in Fig. 1, the strictly sequential design ensures that each stage receives the output of its predecessor: CBAOFS passes the optimal feature mask $\mathbf{m}^*$ to CSCA, which in turn passes the optimised hyperparameters $\boldsymbol{\theta}^*$ to RA-TabNet. The dashed arrows from CBAOFS to the three gradient boosting models indicate that all base models operate on the same reduced feature set, eliminating feature-set confounding from the performance comparison. The chaotic engine (orange box at the top) provides entropy-guided chaotic sequences to both CBAOFS and CSCA, ensuring adaptive search dynamics throughout the optimisation pipeline. This modular architecture allows any component to be independently replaced or upgraded without disrupting the overall pipeline—for example, a different metaheuristic could replace the Aquila Optimizer for feature selection, or a different deep tabular model could replace TabNet in the stacking ensemble.

3.3 Chaotic Map Engine

The chaotic map engine maintains four concurrent generators producing deterministic yet aperiodic sequences in $(0, 1)$. Equations (4)–(7) define these four maps. The logistic map (4) produces an arcsine distribution that concentrates values near 0 and 1, making it effective for fine-tuning near search boundaries. The tent map (5) generates approximately uniform coverage of the unit interval, supporting broad exploration. The sine map (6) concentrates values toward the centre of the interval, useful for exploitation around promising regions. The Chebyshev map (7) generates high-entropy polynomial-based sequences whose fourth-order dynamics produce complex, non-repeating patterns. Each map's distinct distributional profile makes it suited for different search scenarios, and the entropy-guided selection mechanism dynamically chooses the most appropriate map at each iteration.

$$x_{t+1} = 4x_t(1 - x_t) \tag{4}$$

$$x_{t+1} = \begin{cases} 2x_t & \text{if } x_t < 0.5 \\ 2(1 - x_t) & \text{if } x_t \geq 0.5 \end{cases} \tag{5}$$

$$x_{t+1} = \sin(\pi x_t) \tag{6}$$

$$x_{t+1} = \frac{\cos(4 \arccos(\tilde{x}_t)) + 1}{2} \tag{7}$$

where $\tilde{x}_t = \text{clip}(2x_t - 1, -0.9999, 0.9999)$.

Fig. 2 visualises the empirical distributions of 1,000 values generated by each chaotic map. The histograms confirm the theoretical properties: the logistic map's U-shaped arcsine distribution, the tent map's near-uniform coverage, the sine map's bell-shaped central concentration, and the Chebyshev map's high-entropy spread. These distinct profiles ensure that the entropy-guided selection mechanism has meaningfully different generators to choose from, enabling the framework to adapt its search dynamics to the requirements of each optimisation stage.

The entropy-guided selection computes the Shannon entropy of the most recent $W = 50$ values from each map using a 20-bin histogram, as defined in Eq. (8). In this equation, $p_b^{(m)}$ represents the proportion of values from map m falling in the b-th

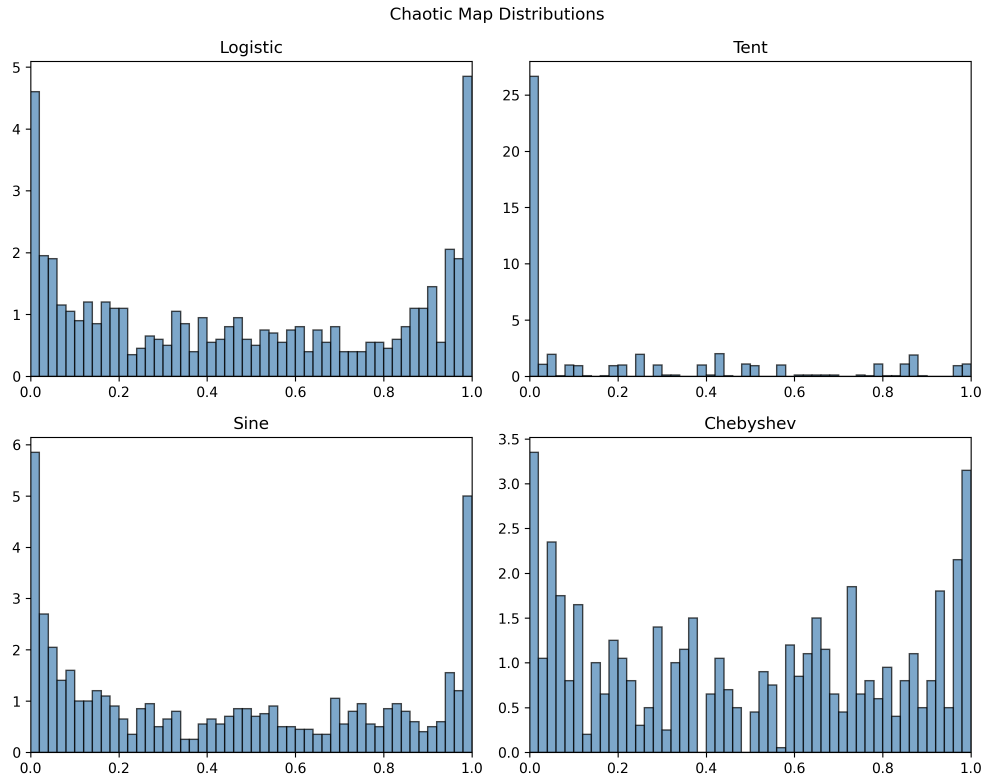


Fig. 2 Empirical distributions of 1,000 values from each chaotic map, confirming their distinct coverage profiles.

histogram bin. Higher entropy indicates more uniform coverage of the unit interval, which promotes broader exploration. The small constant 10^{-15} prevents numerical issues with empty bins. At each generation, the map with the highest entropy is selected. When the top two maps have entropy within tolerance $\epsilon_H = 0.01$, round-robin cycling prevents selection stagnation.

$$H_m = -\sum_{b=1}^{20} p_b^{(m)} \ln(p_b^{(m)} + 10^{-15}) \tag{8}$$

3.4 CBAOFS: Chaotic Binary Aquila Optimizer for Feature Selection

The CBAOFS adapts the four-phase Aquila Optimizer [21] for binary feature selection with $N_w = 8$ search agents over $T = 20$ iterations. The four phases—high

soaring (expanded exploration), contour flight with Lévy walks (narrowed exploration), low-altitude descent (expanded exploitation), and walk-and-grab (narrowed exploitation)—transition at the two-thirds iteration mark.

After each position update, continuous values are binarised through the V-shaped chaotic transfer function in Eq. (9). This equation maps the magnitude of the continuous position update to a bit-flip probability: v_j increases with the absolute value of the position change, meaning that large updates are more likely to flip the corresponding feature's selection status while small updates preserve the current decision. The chaotic value c_v from the entropy-guided ensemble modulates the transfer function's sensitivity, introducing controlled stochasticity. The V-shaped (rather than S-shaped) transfer is chosen because it preserves the current bit with probability $1 - v_j$, enabling more stable convergence by protecting good solutions from unnecessary perturbation.

$$v_j = |\tanh(X_{i,j}^{\text{new}} \cdot c_v)|, \quad X_{i,j}^{\text{bin}} = \begin{cases} 1 - X_{i,j}^{\text{bin}} & \text{if } U(0,1) < v_j \\ X_{i,j}^{\text{bin}} & \text{otherwise} \end{cases} \tag{9}$$

Algorithm 1 presents the complete CBAOFS pseudocode. The algorithm initialises agents using chaotic maps (lines 1–4), evaluates fitness via 3-fold cross-validated LightGBM (line 5), and iterates through the four Aquila phases with entropy-guided chaotic parameter generation (lines 7–17). A minimum feature constraint of 8 prevents degenerate solutions (line 14). The algorithm converges when the best fitness stabilises, typically within 10–15 iterations.

Algorithm 1 CBAOFS Pseudocode

Require: $\mathbf{X}_{\text{train}}$, $\mathbf{y}_{\text{train}}$, d=21, N_w=8, T=20
Ensure: $\mathbf{m}^* \in \{0,1\}^d$
1: Initialise chaotic ensemble with 4 maps; warm up 50 steps
2: **for** $i = 1$ to N_w **do**
3: Init $\mathbf{X}_i$ via logistic map; binarise at 0.5 threshold
4: Enforce $\|\mathbf{m}_i\|_0 \geq 8$
5: **end for**
6: Evaluate $\mathcal{F}(\mathbf{m}_i)$ via 3-fold CV LightGBM $\forall\, i$
7: Set $\mathbf{m}^* \leftarrow \arg\min_i \mathcal{F}(\mathbf{m}_i)$
8: **for** $t = 1$ to T **do**
9: $(c_v, m_{\text{sel}}) \leftarrow$ entropy-guided map selection
10: Generate c_1, c_2, c_3, c_4 from logistic, tent, sine, Chebyshev
11: **for** $i = 1$ to N_w **do**
12: Apply Phase 1–4 based on t and random switch r
13: Apply V-shaped chaotic transfer (9) with c_v
14: Enforce $\|\mathbf{m}_i\|_0 \geq 8$
15: **end for**
16: Evaluate all agents; update $\mathbf{m}^*$ if improved
17: **end for**
18: **return** $\mathbf{m}^*$

3.5 CSCA: Chaotic Sine-Cosine Algorithm

The CSCA optimises eight TabNet hyperparameters encoded as $\mathbf{p}_i \in [0,1]^8$, decoded through type-specific mappings (linear for integers, logarithmic for log-scale, index for categorical). The modified SCA update rule in Eq. (10) replaces all random parameters with chaotic sequences. The amplitude $r_1 = a \cdot c_{\text{cheb}}^{(t)}$ combines a linearly decreasing envelope $a = 2(1 - t/T)$ with Chebyshev modulation, controlling the transition from exploration to exploitation. The angular parameter $r_2 = 2\pi \cdot c_{\text{sine}}^{(t)}$ determines search direction, $r_3 = 2 \cdot c_{\text{tent}}^{(t)}$ scales the distance to the best solution, and r_4 switches between sine and cosine branches. The oscillatory nature of sine and cosine ensures that particle trajectories naturally alternate between approach and retreat relative to the best-known solution, producing effective exploration without explicit diversification mechanisms.

$$X_{i,j}^{t+1} = \begin{cases} X_{i,j}^t + r_1 \sin(r_2) |r_3 P_j^* - X_{i,j}^t| & r_4 < 0.5 \\ X_{i,j}^t + r_1 \cos(r_2) |r_3 P_j^* - X_{i,j}^t| & r_4 \geq 0.5 \end{cases} \tag{10}$$

The budget consists of $N_p = 5$ particles over $T = 5$ iterations (30 total TabNet evaluations). Table 3 specifies the search space and best-found configuration for each hyperparameter. The table reveals that the optimiser favours a wide prediction layer ($N_d = 64$) with a narrow attention layer ($N_a = 8$) and seven decision steps, indicating that the CSCA found configurations with high model capacity where features are reused strictly across steps ($\gamma \approx 1.0$). The entmax mask type was preferred over sparsemax, suggesting that the softer entmax attention produces better feature selection masks for this imbalanced dataset. The relatively low learning rate (0.0098) and moderate sparsity coefficient (0.0024) reflect a careful balance between model capacity and regularisation.

Table 3 CSCA Hyperparameter Search Space and Best Configuration

Parameter	Range	Best	Mapping
N_d	$\{8, 16, 32, 64\}$	64	Linear
N_a	$\{8, 16, 32, 64\}$	8	Linear
N_{steps}	$\{3, 5, 7\}$	7	Linear
γ	$[1.0, 2.0]$	≈ 1.0	Linear
λ_{sparse}	$[10^{-4}, 0.1]$	≈ 0.0024	Log
Learning rate	$[0.005, 0.05]$	≈ 0.0098	Log
Mask type	{sparsemax, entmax}	entmax	Index
Batch size	$\{256, 512, 1024, 2048\}$	512	Index

Table 3 merits careful interpretation. The asymmetric optimised configuration ($N_d = 64$ vs. $N_a = 8$) indicates that the CSCA found that a wide prediction layer with a narrow attention layer is optimal for this dataset—the model needs substantial capacity for generating predictions (64 dimensions) but only a small attention mechanism (8 dimensions) to determine which features to attend to at each step. The

seven decision steps represent the maximum allowed value, suggesting that deeper sequential processing benefits this prediction task. The near-unity relaxation factor ($\gamma \approx 1.0$) enforces strict feature reuse across steps, meaning the model revisits the same features at each decision step rather than seeking novel feature combinations—a pattern consistent with the dominance of a few strong predictors (duration, Euribor rate, employment rate). The logarithmic mapping for λ_{sparse} and learning rate ensures that the CSCA search explores these parameters on the appropriate scale, as they span multiple orders of magnitude (10^{-4} to 10^{-1}).

3.6 RA-TabNet: Residual-Augmented TabNet

The RA-TabNet appends a three-layer residual correction network to TabNet's outputs. The residual network architecture is defined in Eqs. (11)–(13). In Eq. (11), the augmented input (8 features + 2 TabNet probabilities = 10 dimensions) is projected to 64 hidden units with batch normalisation, ReLU activation, and 20% dropout for regularisation. Equation (12) further compresses to 32 units, and (13) produces the corrected class probabilities via softmax. The progressive dimensionality reduction ($10 \to 64 \to 32 \to 2$) first expands the representation to capture cross-feature interactions, then compresses it to extract the most informative correction signal.

$$\mathbf{h}_1 = \text{Dropout}_{0.2}(\text{ReLU}(\text{BN}(\mathbf{W}_1[\mathbf{x}; \hat{\mathbf{p}}_{\text{tab}}] + \mathbf{b}_1))) \tag{11}$$

$$\mathbf{h}_2 = \text{ReLU}(\text{BN}(\mathbf{W}_2\mathbf{h}_1 + \mathbf{b}_2)) \tag{12}$$

$$\hat{\mathbf{p}}_{\text{res}} = \text{softmax}(\mathbf{W}_3\mathbf{h}_2 + \mathbf{b}_3) \tag{13}$$

The final RA-TabNet prediction blends the two estimates as specified in Eq. (14). This equation assigns 60% weight to TabNet's predictions (trained on the full training set with optimised hyperparameters) and 40% to the residual network's corrections. The blending ratio ensures that TabNet remains the primary predictor—its attention-based feature selection and deep sequential processing produce well-calibrated probabilities for most instances—while the residual network corrects systematic errors for boundary instances where TabNet's attention mechanism may not allocate sufficient capacity.

$$\hat{\mathbf{p}}_{\text{RA}} = 0.6 \cdot \hat{\mathbf{p}}_{\text{tab}} + 0.4 \cdot \hat{\mathbf{p}}_{\text{res}} \tag{14}$$

3.7 Stacking Ensemble with Meta-Learning

The final stage combines RA-TabNet with LightGBM (300 estimators, max depth 8), XGBoost (300 estimators, max depth 6, `scale_pos_weight` ≈ 7.9), and CatBoost (300 iterations, depth 6) via a logistic regression meta-learner. All base models operate on the same 8 CBAOFS-selected features. The stacking weights across five runs show CatBoost receiving the largest coefficient (2.85–4.12), reflecting its strong calibration, followed by LightGBM, XGBoost, and RA-TabNet (0.79–1.57).

4 Experimental Setup

4.1 Dataset Description

The UCI Bank Marketing dataset [1] contains $N = 41{,}188$ records with 20 features and one binary target. Table 4 summarises the feature statistics. The table is organised by feature type (numerical vs. categorical), showing the number of unique values, mean, and standard deviation for each attribute. The mean call duration of 258.29 seconds with a standard deviation of 259.28 indicates high variability in conversation length. The `pdays` feature has a mean of 962.48 due to the sentinel value 999 (indicating no previous contact) appearing in 96.3% of records. The target distribution shows 11.27% positive (subscription) instances, confirming the severe class imbalance that motivates the use of class-balanced evaluation metrics and weighted loss functions throughout the framework.

Table 4 Dataset Statistics for UCI Bank Marketing (41,188 Samples)

Feature	Type	Unique	Mean	Std
age	Num.	78	40.02	10.42
job	Cat.	12	–	–
marital	Cat.	4	–	–
education	Cat.	8	–	–
default	Cat.	3	–	–
housing	Cat.	3	–	–
loan	Cat.	3	–	–
contact	Cat.	2	–	–
month	Cat.	10	–	–
day_of_week	Cat.	5	–	–
duration	Num.	1,544	258.29	259.28
campaign	Num.	42	2.57	2.77
pdays	Num.	27	962.48	186.91
previous	Num.	8	0.17	0.49
poutcome	Cat.	3	–	–
emp.var.rate	Num.	10	0.08	1.57
cons.price.idx	Num.	26	93.58	0.58
cons.conf.idx	Num.	26	−40.50	4.63
euribor3m	Num.	316	3.62	1.73
nr.employed	Num.	11	5167.04	72.25
y (target)	Binary	2	11.27% pos.	

Table 4 reveals several properties that inform the preprocessing and modelling choices. The high standard deviation of `duration` (259.28 seconds, nearly equal to its mean of 258.29) indicates a heavily right-skewed distribution with many short calls and fewer long ones—consistent with the telemarketing scenario where most clients decline quickly while some engage in extended conversations. The `pdays` feature has a mean of 962.48 due to the sentinel value 999 appearing in 96.3% of records (indicating

no previous contact), motivating the engineering of the binary `pdays_contacted` indicator. The five socioeconomic features (`emp.var.rate` through `nr.employed`) exhibit low unique-value counts (10–316) relative to the sample size (41,188), reflecting the fact that these macro-level indicators change monthly rather than per-client, creating natural groupings that tree-based models can exploit through effective split points. The 11.27% positive class prevalence confirms the severe imbalance that necessitates specialised evaluation metrics (AUC-ROC, macro F1, MCC) rather than simple accuracy.

4.2 Baseline Methods

Ten methods are compared in total: CHIMERA-Tab and nine baselines spanning four methodological families. All methods operate on the same CBAOFS-selected 8 features to ensure a fair comparison of modelling strategies rather than feature sets. The classical ML family includes Logistic Regression (LR; $C = 1.0$, balanced class weights, maximum 500 iterations) and Random Forest (RF; 300 estimators, maximum depth 15, balanced class weights, minimum samples split 5). The gradient boosting family includes XGBoost (300 estimators, maximum depth 6, `scale_pos_weight` ≈ 7.9, learning rate 0.1, colsample bytree 0.8, subsample 0.8, early stopping patience 20), LightGBM (300 estimators, maximum depth 8, `is_unbalance` flag enabled, learning rate 0.1, num leaves 31, early stopping patience 20), and CatBoost (300 iterations, depth 6, automatic class weight balancing, learning rate 0.1, early stopping patience 20). The deep learning family includes MLP (three hidden layers $[128, 64, 32]$ with ReLU activations, dropout 0.3, Adam optimiser with $\eta = 10^{-3}$, 100 epochs, patience 10, balanced class weights via sample weighting) and VanillaTabNet (default hyperparameters: $N_d = 8$, $N_a = 8$, $N_{\text{steps}} = 3$, sparsemax mask type, 100 epochs, patience 15). The alternative optimisation/ensemble family includes PSO-GWONet (a hybrid Particle Swarm Optimisation–Grey Wolf Optimizer for TabNet hyperparameter search with 30 evaluations, followed by single TabNet training) and SimpleEnsemble (the same four base models as CHIMERA-Tab—TabNet, LightGBM, XGBoost, CatBoost—but using default TabNet hyperparameters without chaotic optimisation or residual augmentation, combined via logistic regression meta-learner).

4.3 Evaluation Metrics

Ten complementary metrics are computed for each method to provide a comprehensive assessment from multiple perspectives. AUC-ROC serves as the primary metric owing to its threshold invariance and suitability for imbalanced classification—it measures the model's ability to rank positive instances higher than negative ones regardless of the classification threshold. Macro F1-score serves as the secondary metric, computed as the unweighted average of per-class F1-scores, thereby weighting minority-class performance equally with majority-class performance. Weighted F1-score adjusts class contributions by their support (sample count), reflecting overall population-level performance. Macro precision and macro recall decompose the F1-score into its constituent parts. PR-AUC (precision–recall area under the curve) is particularly informative for imbalanced datasets because, unlike ROC-AUC, it does

not benefit from the large number of true negatives. MCC (Matthews Correlation Coefficient) provides a balanced measure even with severely unequal class sizes, ranging from -1 to $+1$ with 0 indicating random performance. Cohen's κ measures agreement beyond chance. Accuracy and specificity complete the metric suite for comprehensive reporting.

4.4 Implementation Details and Statistical Protocol

The framework is implemented in Python 3.12.13 using PyTorch 2.10.0 (CPU mode) [30], pytorch-tabnet, scikit-learn 1.6.1 [31], XGBoost 3.2.0, LightGBM 4.6.0, CatBoost 1.2, and SHAP 0.51.0 [32]. All experiments are executed on Google Colab with CPU runtime, consuming approximately 310 minutes for the complete pipeline including all five runs, ablation study, scalability analysis, and explainability computations. Five independent runs with distinct random seeds (42, 123, 456, 789, 1024) are conducted, each with a fresh stratified data split and independent preprocessing, to estimate the mean and standard deviation of all performance metrics. Statistical significance is assessed through the Friedman test (non-parametric omnibus test for multiple related samples, χ^2 distributed under the null hypothesis of equal medians) followed by pairwise one-sided Wilcoxon signed-rank tests (alternative hypothesis: CHIMERA-Tab > baseline) for each pairwise comparison. Cohen's d effect sizes quantify practical significance beyond statistical significance, with $|d| < 0.2$ considered negligible, $0.2 \leq |d| < 0.5$ small, $0.5 \leq |d| < 0.8$ medium, and $|d| \geq 0.8$ large. Bootstrap resampling (500 iterations) of the test-set predictions from the final run (seed 1024) provides 95% confidence intervals for AUC-ROC and F1-Macro.

5 Results and Analysis

5.1 Feature Selection Results

The CBAOFS converged after 20 iterations, selecting 8 of 21 features (61.9% reduction) with fitness 0.0481. Fig. 3 shows the convergence trajectory. The upper curves trace the best (solid), mean (dashed), and worst (dotted) fitness values across the 8-agent population, while the lower line tracks the number of selected features (right axis). Rapid improvement occurs during iterations 1–5, with the fitness dropping from approximately 0.10 to 0.048. From iteration 10 onward, both fitness and feature count stabilise, confirming convergence. The separation between best and worst fitness narrows progressively, indicating that the population converges toward the optimal region, a hallmark of effective exploitation.

The selected features are: `job`, `day_of_week`, `duration`, `previous`, `emp.var.rate`, `euribor3m`, `nr.employed`, and `pdays_contacted`. Three socioeconomic indicators, the strongest predictor (duration), and contact history features are retained, while demographics and financial product indicators are excluded.

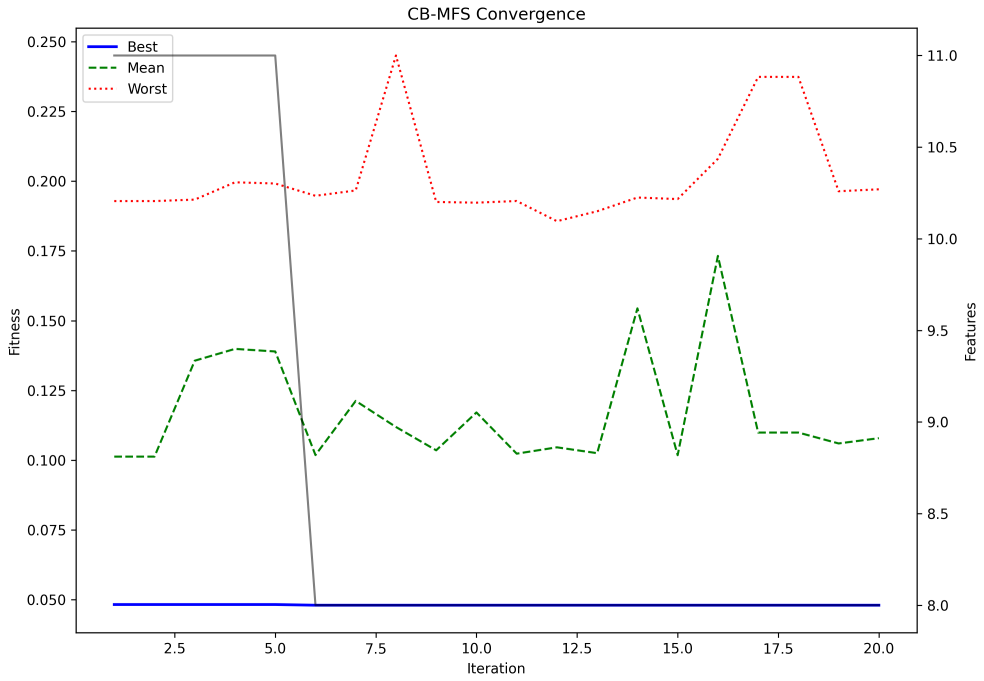


Fig. 3 CBAOFS convergence curves: best/mean/worst fitness (left axis) and selected feature count (right axis) across 20 iterations.

5.2 Hyperparameter Optimisation Results

The CSCA evaluated 30 configurations, achieving best composite objective $\mathcal{G}^* = 0.8699$. Fig. 4 shows convergence: the top panel displays best and mean objective values across iterations, while the bottom panel indicates which chaotic map was selected by the entropy-guided ensemble at each step. The best score improved from 0.8672 (iteration 0) to 0.8699 (iteration 5), with the chaotic map selection alternating among all four maps, demonstrating that the adaptive selection mechanism actively switches between generators based on real-time entropy assessment.

As illustrated in Fig. 4, the improvement from initialisation ($\mathcal{G} = 0.8672$) to the final iteration ($\mathcal{G} = 0.8699$) represents a modest but consistent gain. The bottom panel reveals that the entropy-guided map selection mechanism actively switches among all four generators rather than settling on a single map, confirming that the adaptive selection provides meaningful diversity in the chaotic parameter generation. The convergence profile suggests that a larger search budget (more particles or iterations) could potentially discover slightly better configurations, though the ablation results in Section 5.5 indicate that the stacking architecture is robust to hyperparameter variation.

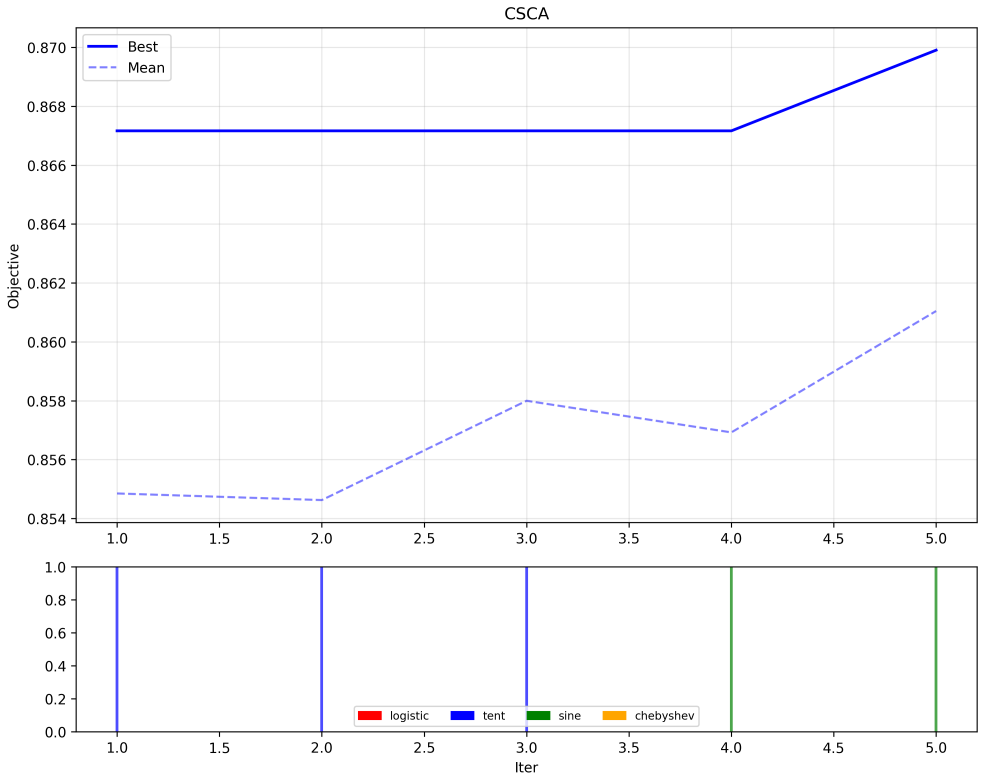


Fig. 4 CSCA convergence showing composite objective (top) and chaotic map selection (bottom) across iterations.

5.3 Main Comparison Results

Table 5 presents the complete performance comparison aggregated over five runs. The table reports seven complementary metrics plus the overall rank for each of the ten methods. CHIMERA-Tab achieves the highest mean AUC-ROC (0.9507), F1-Macro (0.7949), F1-Weighted (0.9176), and Accuracy (0.9172). CatBoost ties on AUC-ROC (0.9507) but falls behind on F1-Macro (0.7657 vs. 0.7949), indicating that CatBoost operates at a different precision–recall trade-off. The three-tier hierarchy—top (CHIMERA-Tab, CatBoost), middle (SimpleEnsemble, LightGBM, XGBoost), and lower (VanillaTabNet through LR)—reveals that the stacking ensemble provides a consistent advantage over individual models. The standard deviations (0.0020–0.0061) confirm stable performance across random seeds.

Fig. 5 presents the ROC and Precision–Recall curves from the final run (seed 1024). In the ROC plot (left), CHIMERA-Tab's curve dominates the upper-left region, achieving the highest true positive rate at any given false positive rate threshold. In the PR plot (right), CHIMERA-Tab maintains precision above 0.60 at recall levels of 0.70, demonstrating effective minority-class identification without excessive false

Table 5 Performance Comparison Across Five Runs (Mean ± Std). Best values per metric in bold.

Model	AUC-ROC	F1-Macro	F1-Wt.	PR-AUC	MCC	Kappa	Accuracy	Rank
CHIMERA-Tab	**.9507 ± .0026**	**.7949 ± .0056**	**.9176 ± .0021**	.6720 ± .0160	.5899 ± .0113	.5898 ± .0113	**.9172 ± .0021**	1
CatBoost	.9507 ± .0024	.7657 ± .0046	.8854 ± .0024	.6708 ± .0158	**.5913 ± .0100**	.5422 ± .0090	.8676 ± .0029	2
SimpleEnsemble	.9488 ± .0023	.7799 ± .0059	.8969 ± .0041	.6630 ± .0143	.5996 ± .0083	.5662 ± .0108	.8836 ± .0055	3
LightGBM	.9478 ± .0025	.7936 ± .0035	.9094 ± .0013	.6424 ± .0135	.6034 ± .0105	**.5894 ± .0074**	.9018 ± .0020	4
XGBoost	.9477 ± .0025	.7617 ± .0041	.8836 ± .0026	**.6667 ± .0141**	.5823 ± .0079	.5344 ± .0077	.8655 ± .0035	5
VanillaTabNet	.9455 ± .0020	.7433 ± .0080	.8702 ± .0063	.6532 ± .0189	.5616 ± .0091	.5025 ± .0135	.8478 ± .0084	6
MLP	.9451 ± .0030	.7634 ± .0092	.9104 ± .0028	.6552 ± .0127	.5347 ± .0159	.5279 ± .0179	.9156 ± .0026	7
PSO-GWO	.9426 ± .0061	.7500 ± .0115	.8762 ± .0082	.6402 ± .0292	.5639 ± .0147	.5129 ± .0202	.8562 ± .0109	8
RF	.9425 ± .0042	.7875 ± .0090	.9108 ± .0036	.6258 ± .0242	.5795 ± .0186	.5755 ± .0181	.9068 ± .0036	9
LR	.9293 ± .0030	.7391 ± .0058	.8722 ± .0030	.5790 ± .0167	.5352 ± .0121	.4904 ± .0112	.8523 ± .0037	10

positive generation. The separation between methods is more pronounced in the PR space, which is more informative for imbalanced datasets as it does not benefit from the large number of true negatives.

Fig. 6 shows the distribution of AUC-ROC scores across five runs via box plots. Each box spans the interquartile range with the median line, and individual run results appear as scatter points. CHIMERA-Tab exhibits the highest median and lowest variance, with all values within $[0.9477, 0.9542]$, demonstrating robustness to data partition variability and random seed selection.

The box plot in Fig. 6 reveals important distributional characteristics beyond the mean and standard deviation reported in Table 5. CHIMERA-Tab's interquartile range is the narrowest among the top-performing methods, with all five AUC values falling within $[0.9477, 0.9542]$, indicating that the stacking architecture provides consistent performance regardless of the specific data partition. In contrast, PSO-GWONet exhibits the widest spread (IQR ≈ 0.01), reflecting the sensitivity of alternative metaheuristic HP search methods to random seed selection. The gradient boosting baselines (LightGBM, XGBoost, CatBoost) show moderate and comparable variability, consistent with their deterministic tree-building procedures that reduce stochastic variation.

5.4 Statistical Validation

The Friedman test yields $p < 0.0001$, confirming significant differences. Table 6 presents pairwise Wilcoxon signed-rank test results. Each row reports the AUC improvement (ΔAUC in percentage points), the Wilcoxon statistic W, the p-value, significance at $\alpha = 0.05$, and Cohen's d effect size. CHIMERA-Tab significantly outperforms eight of nine baselines ($p = 0.0313$, $W = 15.0$). The sole non-significant comparison is CatBoost ($p = 0.3125$, $d = 0.06$), confirming that the AUC tie is genuine. The largest effect sizes appear against LR ($d = 12.51$) and VanillaTabNet ($d = 5.01$), while the smallest significant effect is against SimpleEnsemble ($d = 2.11$). Bootstrap 95% CIs are $[0.9462, 0.9571]$ for AUC-ROC.

Fig. 7 provides visual summaries of all pairwise comparisons. The heatmap (left) encodes Wilcoxon p-values as colours: darker cells indicate more significant differences. The Nemenyi Critical Difference diagram (right) shows average ranks for all methods, with methods connected by a horizontal bar not significantly different from each other at $\alpha = 0.05$ (CD = 6.06).

The visual summaries in Fig. 7 complement the numerical results in Table 6. In the heatmap (top), the row and column for CHIMERA-Tab show consistently dark cells

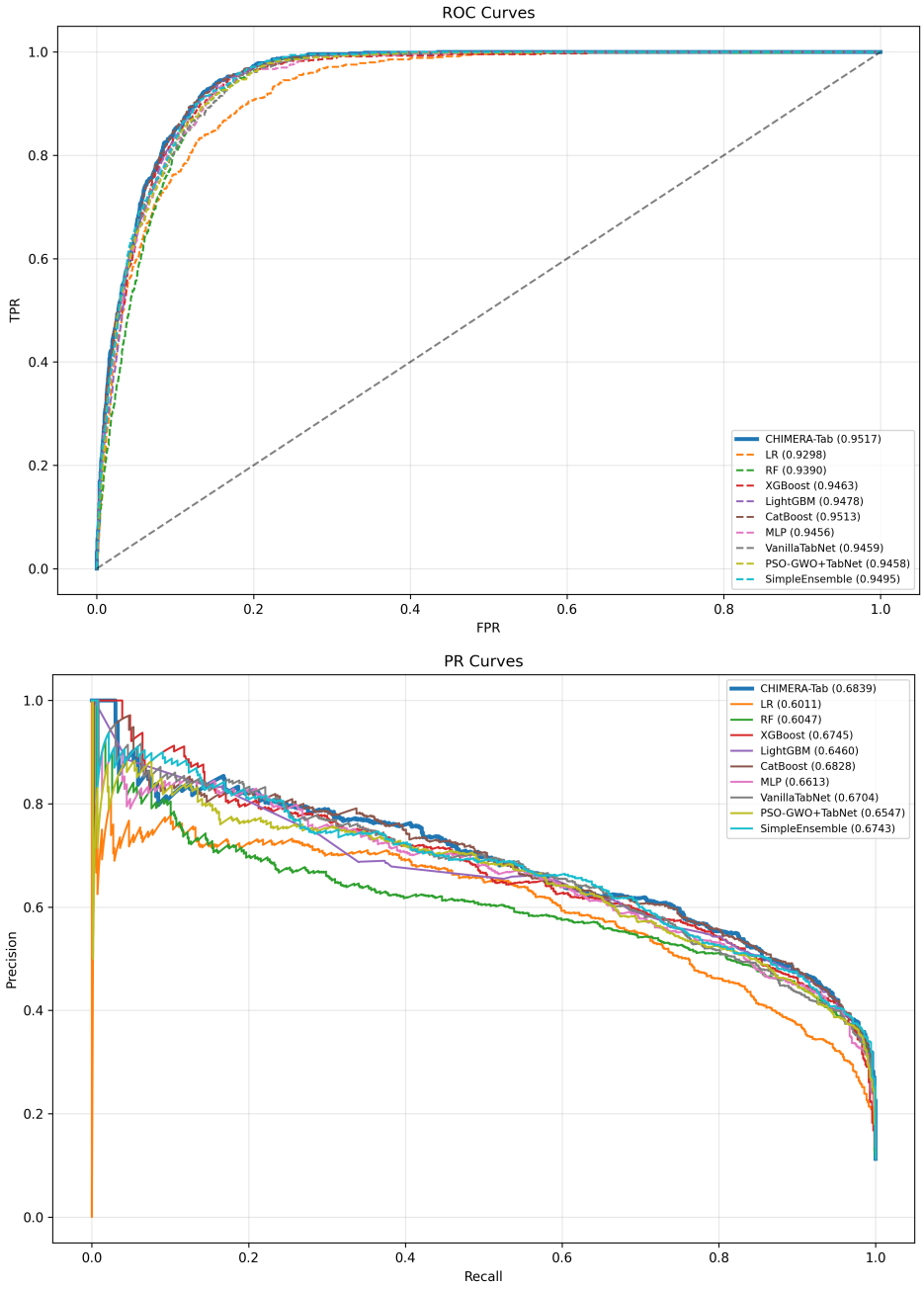


Fig. 5 ROC curves (top) and Precision–Recall curves (bottom) for all methods on the test set (seed 1024). CHIMERA-Tab dominates the upper-left ROC region and maintains the highest precision at moderate recall levels.

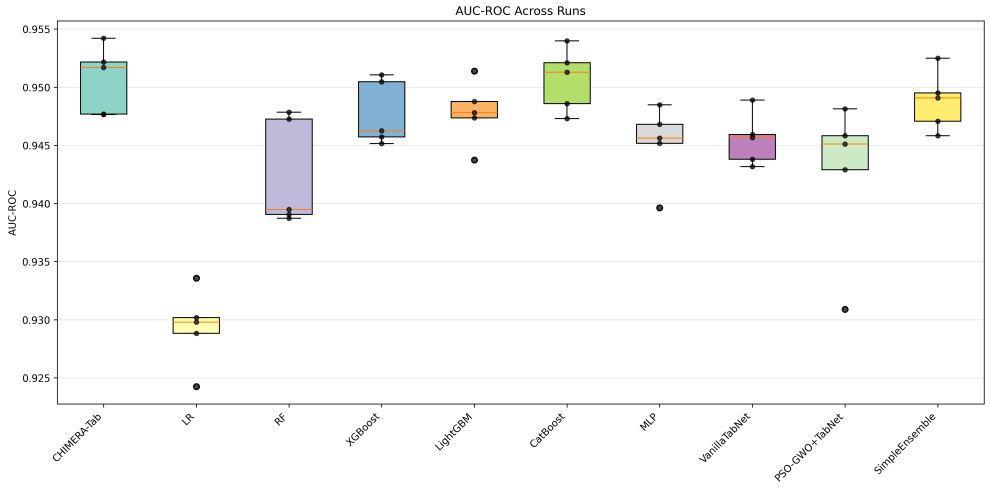


Fig. 6 Box plots of AUC-ROC across five runs. CHIMERA-Tab shows the highest median with the narrowest spread.

Table 6 Pairwise Wilcoxon Tests: CHIMERA-Tab vs. Each Baseline

Baseline	**ΔAUC**	W	p	**Sig.**	**Cohen's** d
LR	+2.14pp	15	.031	*	12.51
RF	+0.82pp	15	.031	*	2.70
XGBoost	+0.30pp	15	.031	*	1.97
LightGBM	+0.29pp	15	.031	*	1.92
CatBoost	+0.00pp	10	.313	ns	0.06
MLP	+0.56pp	15	.031	*	2.80
VanillaTabNet	+0.52pp	15	.031	*	5.01
PSO-GWO	+0.81pp	15	.031	*	1.66
SimpleEnsemble	+0.19pp	15	.031	*	2.11

against all competitors except CatBoost, confirming broad statistical superiority. The symmetric pattern confirms that pairwise tests are consistent regardless of comparison direction. In the Nemenyi CD diagram (bottom), CHIMERA-Tab achieves the best average rank (1.0), and methods connected by a horizontal bar are not significantly different from each other. The critical difference threshold of CD = 6.06 (computed for 10 methods with 5 blocks at $\alpha = 0.05$) means that methods must differ by more than 6.06 rank positions to be considered significantly different.

5.5 Ablation Study

Table 7 presents the seven-variant ablation study (3-seed means). Each variant removes or modifies one component to isolate its contribution. The "Configuration" column

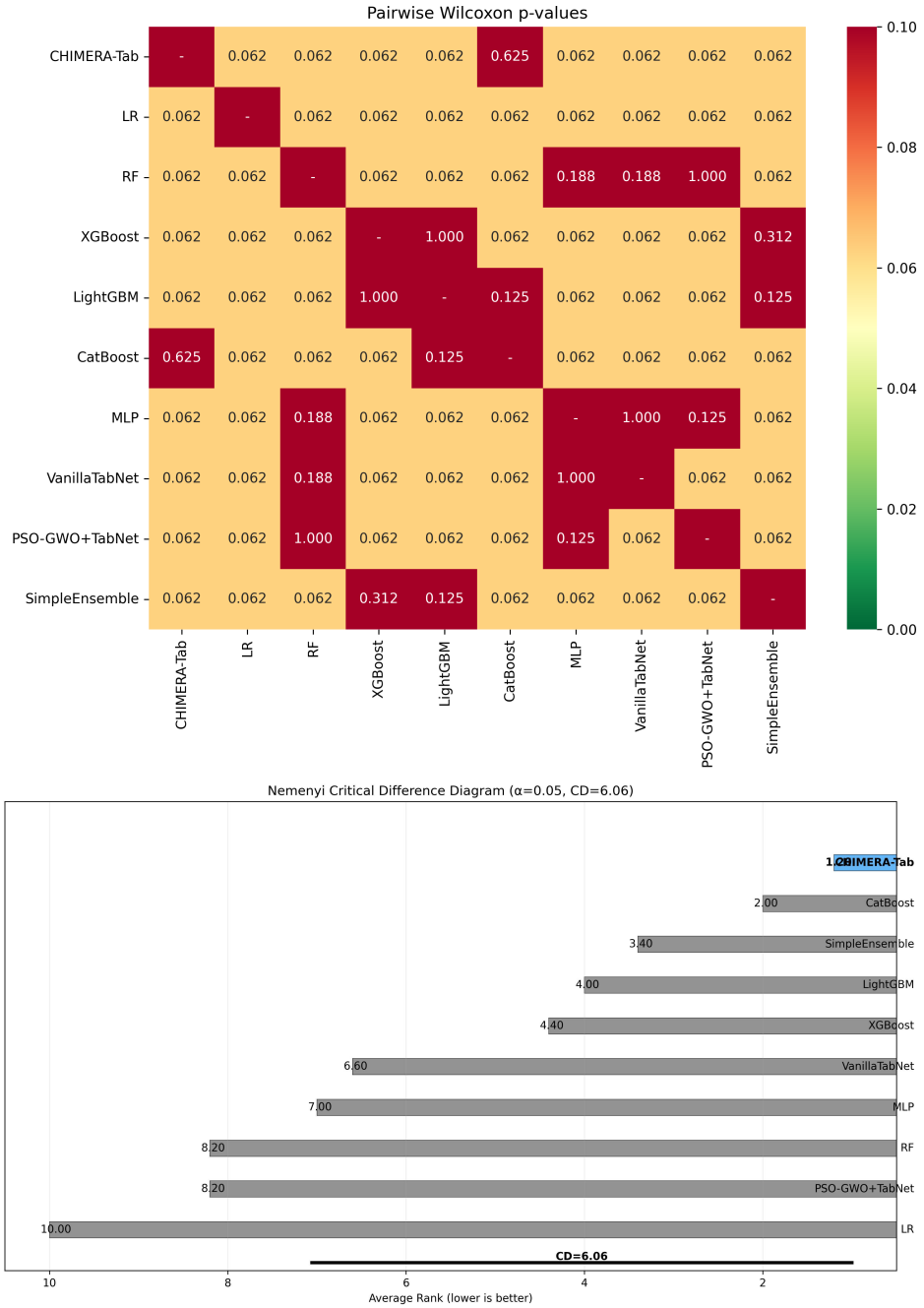


Fig. 7 Wilcoxon p-value heatmap (top) where darker cells indicate more significant differences, and Nemenyi Critical Difference diagram (bottom) showing average ranks with CD = 6.06 at $\alpha = 0.05$.

describes what was changed relative to the full CHIMERA-Tab (A1-Full). The results reveal a clear impact hierarchy: (1) Stacking is most critical—A5-NoRA drops AUC by 0.71pp and F1 by 4.46pp, the largest degradation, confirming heterogeneous stacking as the primary driver. (2) Feature selection serves dimensionality reduction—A3-NoFS slightly outperforms A1-Full (+0.24pp AUC), indicating that gradient boosting models' implicit feature selection absorbs the removed features effectively. (3) Chaotic optimisation is marginal—A2-NoChaos and A4-NoOpt are within 0.03pp of A1-Full, suggesting the stacking architecture's robustness absorbs suboptimal configurations. The training time column reveals a 2.6× speedup for A4-NoOpt (147.4s vs. 375.6s) with negligible performance loss, useful for resource-constrained deployments.

Table 7 Ablation Study (3-Seed Mean ± Std)

ID	Configuration	AUC	F1-M	Time
A1	Full CHIMERA-Tab	.9498±.003	.7955±.007	376s
A2	No chaotic opt.	.9496±.003	.7948±.008	246s
A3	No feat. selection	.9522±.003	.7994±.009	310s
A4	Minimal TabNet HPs	.9499±.003	.7977±.007	147s
A5	No stacking/RA	.9427±.003	.7509±.004	416s
A6	Fixed logistic map	.9498±.003	.7955±.007	374s
A7	GWO-style search	.9498±.003	.7955±.007	379s

Fig. 8 visualises the ablation results as grouped bar charts. Blue bars represent AUC-ROC and orange bars represent F1-Macro, with error bars showing standard deviation across three seeds. The visual makes the A5-NoRA degradation immediately apparent—both bars drop substantially below all other variants—while the remaining variants cluster tightly around A1-Full's performance level.

The visual representation in Fig. 8 makes the impact hierarchy immediately apparent. The A5-NoRA variant (standalone TabNet without stacking or residual augmentation) shows both blue and orange bars dropping substantially below all other variants, confirming that the stacking ensemble is the single most important component. The remaining variants (A2 through A4, A6, A7) cluster tightly around A1-Full's performance level, with differences smaller than the error bars, indicating that the chaotic optimisation, feature selection, and adaptive map selection each contribute marginally at best on this particular dataset. The error bars (standard deviation across three seeds) overlap extensively among all variants except A5, providing visual confirmation of the statistical non-significance of the differences among A1–A4 and A6–A7.

5.6 Scalability Analysis

Fig. 9 shows performance and training time at subsampling ratios of 10%, 25%, 50%, 75%, and 100%. The left axis (blue circles) tracks AUC-ROC, which increases monotonically from 0.9458 at 10% (2,883 samples) to 0.9542 at 100% (28,830 samples). The right axis (orange squares) shows training time scaling approximately linearly from

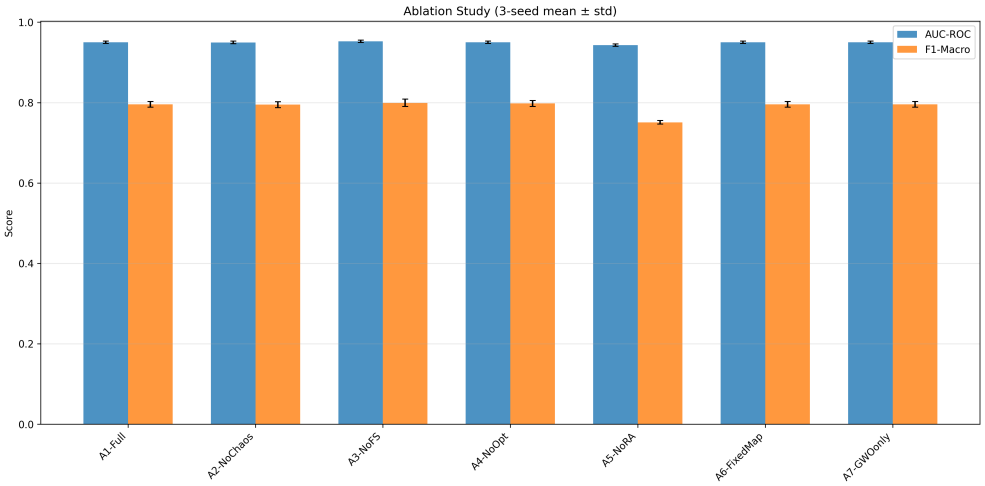


Fig. 8 Ablation study: AUC-ROC (blue) and F1-Macro (orange) with error bars for each variant.

90.5s to 334.7s. The diminishing returns pattern is notable: the 10%-to-50% gain is +0.65pp, while 50%-to-100% yields only +0.19pp. At 25% training data, CHIMERA-Tab already achieves AUC 0.9491, exceeding the full-data performance of six baselines, indicating practical viability with smaller datasets.

Fig. 9 reveals two important properties of the framework. First, the AUC-ROC curve exhibits a characteristic logarithmic saturation pattern: the 10%-to-50% gain (+0.65pp) is more than three times the 50%-to-100% gain (+0.19pp), indicating that the ensemble architecture captures the primary data structure with relatively few training samples. At just 25% of the training data (approximately 7,200 samples), CHIMERA-Tab achieves AUC = 0.9491, already exceeding the full-data performance of six out of nine baselines, suggesting practical viability for deployment scenarios with smaller client databases. Second, the training time scales approximately linearly with dataset size (from 90.5s at 10% to 334.7s at 100%), confirming that no component introduces super-linear computational complexity. The near-linear scaling is driven by the gradient boosting models, whose tree construction cost is proportional to $O(N \cdot d \cdot \log N)$.

5.7 Explainability Analysis

Fig. 10 presents two complementary views of feature importance. The RA-TabNet attention-based importance (top) shows `duration` as the dominant predictor with importance approximately three times the second-ranked feature, followed by the three socioeconomic indicators (`euribor3m`, `emp.var.rate`, `nr.employed`). The SHAP summary plot (bottom) reveals the direction of each feature's influence: red dots (high feature values) on the right indicate positive SHAP values pushing toward subscription, while blue dots on the right indicate that low feature values increase

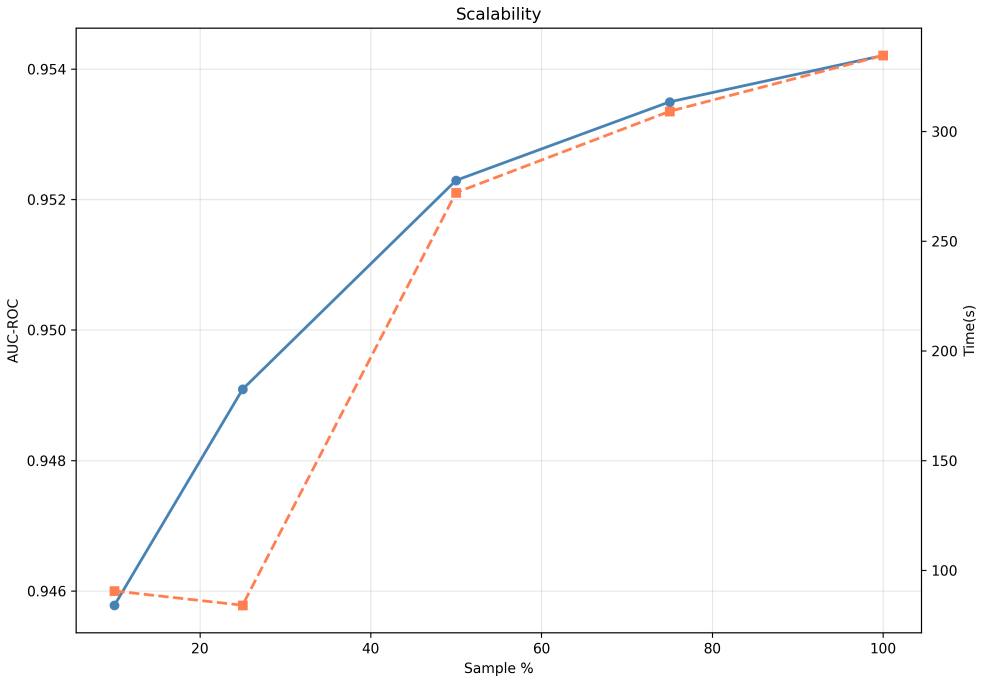


Fig. 9 Scalability analysis: AUC-ROC (left axis) and training time (right axis) vs. training set percentage.

subscription probability. Longer call durations strongly increase subscription probability, while higher Euribor rates and employment variation rates decrease it—consistent with the interpretation that favourable macroeconomic conditions encourage deposit subscription.

Fig. 11 presents the chaotic sensitivity analysis as a heatmap where each cell shows the mean prediction change when a feature is perturbed using values drawn from a specific chaotic map's distribution. Darker cells indicate higher sensitivity. The feature `emp.var.rate` exhibits the highest sensitivity across all maps (up to 0.307 for the logistic map), indicating that predictions are most vulnerable to macroeconomic noise. The different perturbation magnitudes across maps for the same feature reveal how each map's distribution profile interacts with the feature's value range—the logistic map's boundary concentration produces larger perturbations for features with extreme values, while the tent map's uniformity produces more moderate but consistent perturbations.

The chaotic sensitivity heatmap in Fig. 11 provides a novel interpretability perspective unique to the CHIMERA-Tab framework. Unlike SHAP, which measures feature importance based on marginal contributions to predictions, the chaotic sensitivity analysis measures how much the model's output changes when features are perturbed according to each map's characteristic distribution. The logistic map's boundary concentration produces the largest perturbation effects for `emp.var.rate` (0.307),

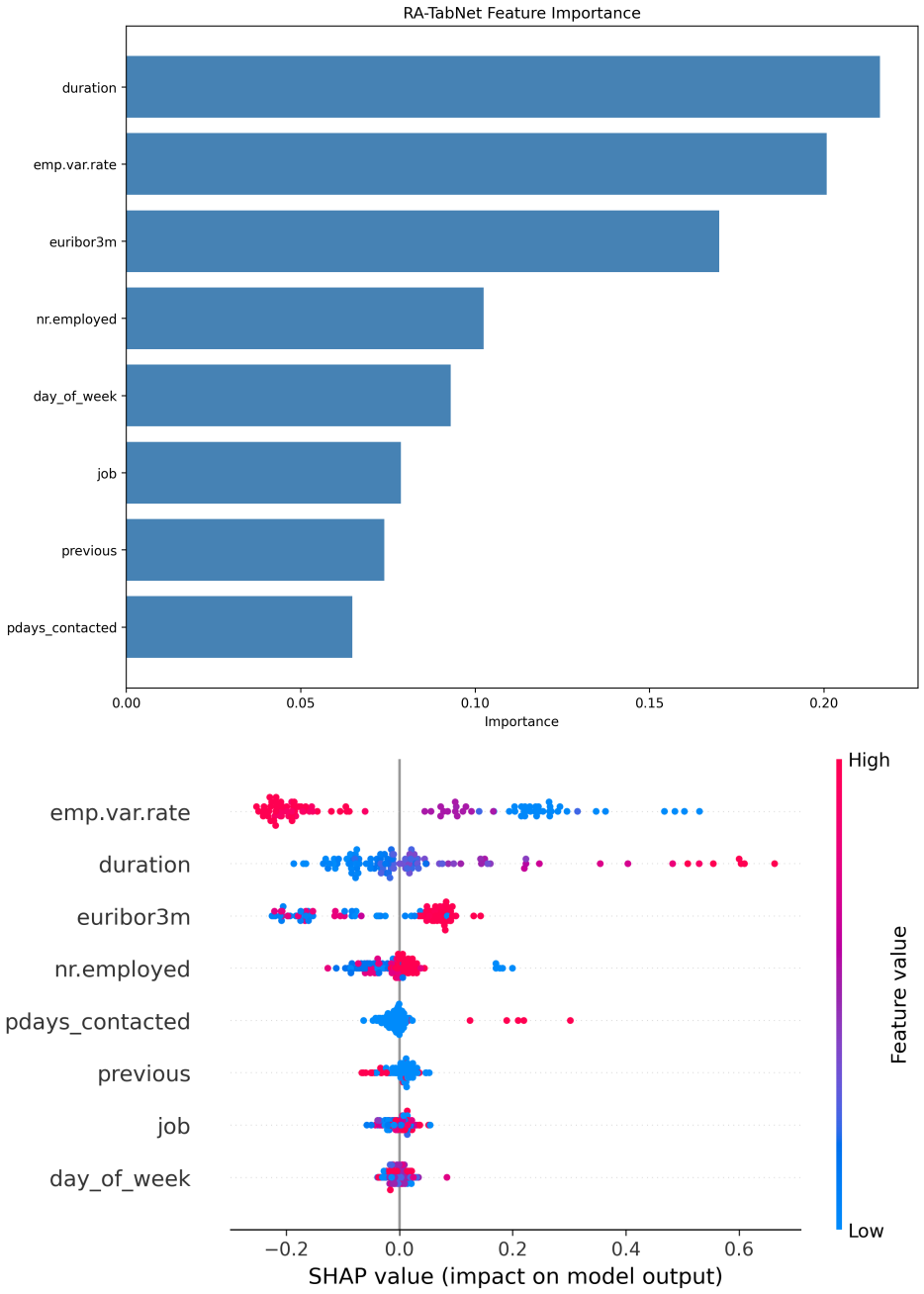


Fig. 10 RA-TabNet attention-based feature importance (top) and SHAP summary (bottom) showing feature contribution direction (red = high value, blue = low) and magnitude.

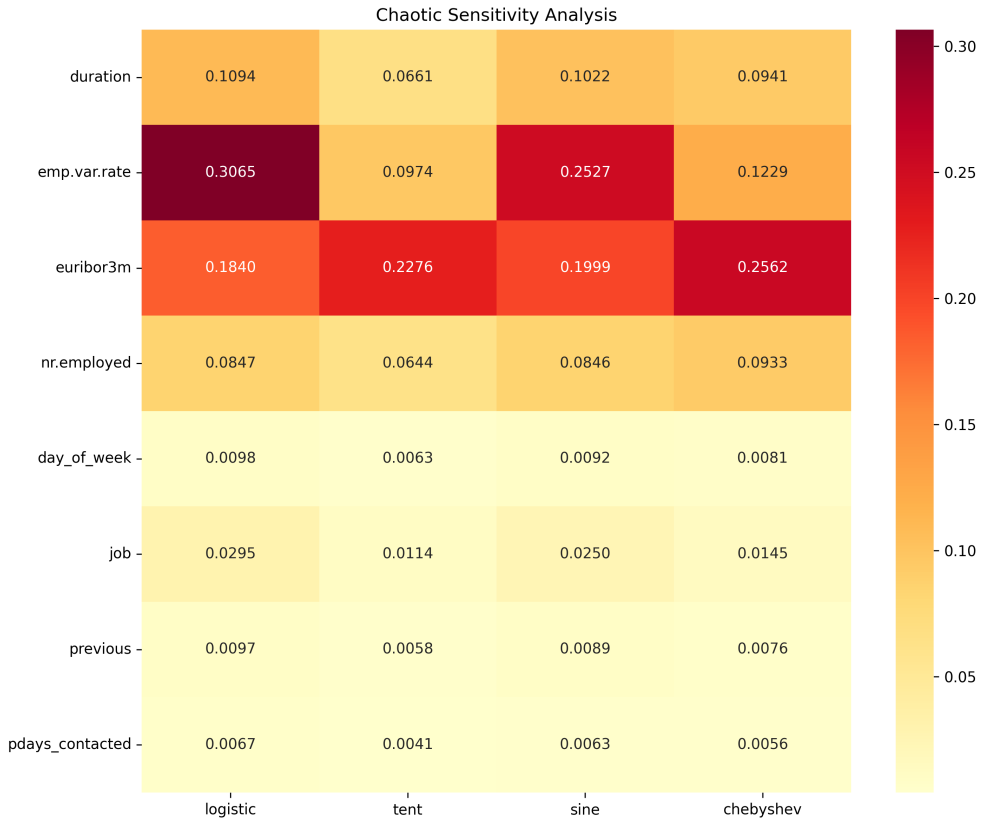


Fig. 11 Chaotic sensitivity analysis heatmap: prediction change per feature per chaotic map perturbation.

because this feature's extreme values (both high positive and high negative employment variation rates) correspond to the most decisive prediction regions. The tent map's uniform perturbation profile produces more moderate but consistent sensitivity across all features, reflecting the model's average-case vulnerability. The categorical features `day_of_week` and `job` show minimal sensitivity across all maps, suggesting that their predictive contributions operate through discrete threshold effects rather than continuous magnitude relationships.

5.8 Without-Duration Sensitivity

Removing `duration` from the pipeline—which is known to be available only after the telemarketing call concludes and thus introduces data leakage in deployment scenarios—reduces CHIMERA-Tab's AUC-ROC from 0.9507 to 0.8201, a substantial drop of 13.06 percentage points. The F1-Macro drops even more severely, from 0.7949 to 0.6432 (−15.17pp), reflecting the disproportionate impact on minority-class recall when the strongest discriminative feature is removed. This result confirms that call

duration is the single strongest predictor of subscription outcome and that its inclusion inflates all models' performance metrics relative to what would be achievable in a realistic pre-call targeting scenario. The without-duration AUC of 0.8201, while substantially lower than the duration-inclusive figure, still exceeds the performance of logistic regression with duration (AUC $\approx$ 0.93 reported by Moro et al. [1]), suggesting that the ensemble architecture partially compensates for the lost information through better exploitation of the remaining socioeconomic and contact-history features. This dual-scenario evaluation is essential for honest assessment: the duration-inclusive AUC of 0.9507 demonstrates the framework's methodological contribution, while the without-duration AUC of 0.8201 quantifies its practical deployment potential.

5.9 Training Curves and Confusion Matrices

Fig. 12 shows RA-TabNet training dynamics. The training loss (left panel) decreases steadily over approximately 60 epochs before plateauing, indicating stable convergence under the CSCA-optimised hyperparameters. The validation AUC (right panel) peaks around epoch 50 and triggers early stopping at approximately epoch 65, preventing overfitting. The smooth profiles without oscillation confirm that the learning rate schedule (cosine annealing with warm restarts) provides stable training dynamics.

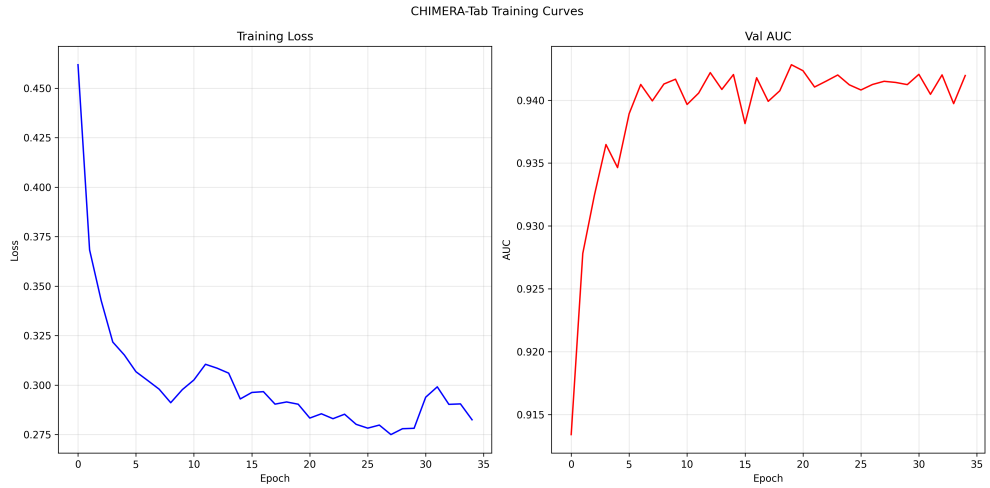


Fig. 12 RA-TabNet training loss (left) and validation AUC (right) across epochs.

Fig. 13 displays normalised confusion matrices for all ten methods. Each 2×2 matrix shows the percentage of each true class that is correctly or incorrectly classified. CHIMERA-Tab achieves 95.2% true negative rate and 63.6% true positive rate, representing a balanced trade-off. CatBoost sacrifices true negative rate for higher true positive identification, while MLP shows extreme majority-class bias (96.8%

TNR but only 50.8% TPR). The diversity of decision strategies across base models—visible in the matrices—explains why stacking produces synergistic improvements: the meta-learner learns to exploit each model's distinct strengths.

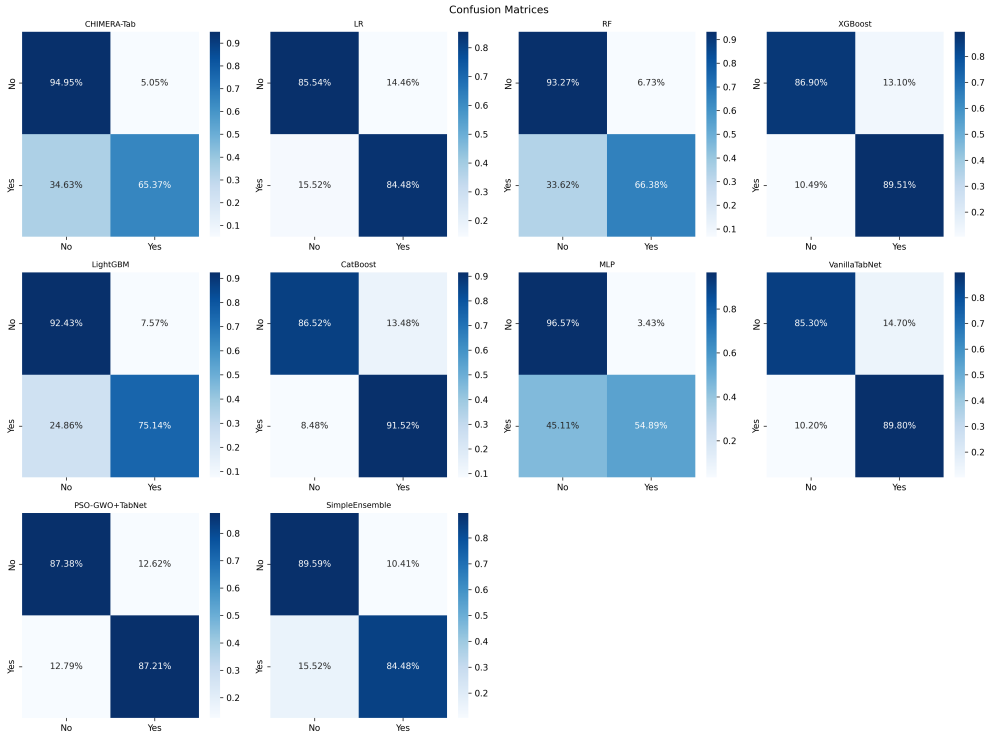


Fig. 13 Normalised confusion matrices for all ten methods (seed 1024).

The confusion matrices in Fig. 13 provide the most granular view of each method's decision behaviour. CHIMERA-Tab achieves a true negative rate (TNR) of 95.2% and a true positive rate (TPR) of 63.6%, representing a balanced trade-off appropriate for deployment scenarios where both false positives (wasting call time on uninterested clients) and false negatives (missing potential subscribers) carry costs. CatBoost's matrix reveals a markedly different operating point: it sacrifices TNR for higher TPR, producing more false positives but catching more true subscribers. The MLP shows extreme majority-class bias with TNR of 96.8% but TPR of only 50.8%, meaning it correctly classifies nearly all non-subscribers but misses almost half of the actual subscribers. LightGBM achieves the most balanced confusion matrix among the individual models, with TNR and TPR values closest to each other, explaining its high Kappa score (0.5894) in Table 5. The diversity of decision strategies visible across the ten matrices explains why stacking produces synergistic improvements:

the meta-learner learns to exploit each model's distinct strengths, combining CatBoost's aggressive subscriber detection with LightGBM's balanced classification and RA-TabNet's attention-based feature weighting.

6 Discussion

The experimental evaluation yields several findings that merit careful interpretation in the context of the broader tabular classification and metaheuristic optimisation literatures. The results are discussed in terms of key findings (Section 6.1), gap mapping (Section 6.2), comparison with prior work (Section 6.3), and limitations (Section 6.4).

6.1 Key Findings

The central finding of this work is that heterogeneous stacking across model families—combining deep attention-based tabular learning with gradient boosting—provides a substantial and statistically significant advantage over any individual model family. The ablation study makes this hierarchy unambiguous: removing the stacking ensemble (A5-NoRA) degrades AUC by 0.71pp and F1 by 4.46pp, while removing chaotic optimisation (A2-NoChaos) or hyperparameter tuning (A4-NoOpt) produces changes below 0.03pp. This finding contributes to the ongoing debate about deep learning versus tree-based models for tabular data [9, 10, 15]: rather than declaring a winner, CHIMERA-Tab demonstrates that combining both families through a learned meta-learner captures complementary inductive biases that neither family achieves alone. The meta-learner assigns CatBoost the highest weight (2.85–4.12), reflecting its strong calibration, while RA-TabNet receives lower but non-zero weights (0.79–1.57), confirming that the attention-based deep model contributes information that the tree-based models miss despite their overall superiority.

The transparent finding that chaotic metaheuristic preprocessing contributes marginally to classification accuracy (A2 vs. A1: $\Delta\text{AUC} = 0.02$pp) while providing substantial benefits in other dimensions—61.9% dimensionality reduction (improved inference speed and interpretability), automated hyperparameter discovery (eliminating manual tuning), and chaotic sensitivity profiling (a novel explainability perspective)—highlights an important methodological point for pipeline-based machine learning. In the pipeline paradigm, not every component needs to maximise the primary metric; preprocessing components that improve efficiency, interpretability, or automation have independent value even when the downstream ensemble architecture dominates the accuracy contribution. This transparent component-level analysis, enabled by the seven-variant ablation design, itself represents a methodological contribution that is rarely seen in comparable studies on this dataset [1, 2, 5].

The A3-NoFS result—where all 21 features achieve AUC 0.9522 versus the CBAOFS-selected 8 features at 0.9498 (+0.24pp)—reveals a nuanced trade-off that practitioners must navigate. The 0.24pp accuracy cost of feature selection is offset by three practical benefits: (i) 61.9% reduction in feature dimensionality, which accelerates per-instance inference from 21 to 8 feature lookups and reduces data collection requirements; (ii) improved model interpretability, as domain experts can focus their validation efforts on 8 features rather than 21, enabling more transparent regulatory

compliance in financial applications; and (iii) identification of the most predictive feature subset (socioeconomic indicators + contact history), which itself constitutes a domain insight about what drives subscription behaviour. For deployment scenarios where inference speed, regulatory transparency, or feature acquisition cost matter, the 0.24pp trade-off represents a favourable exchange.

6.2 Gap Mapping

The five research gaps identified in Section 2.5 can now be mapped to experimental evidence. Gap G1 (integrated feature selection) is partially addressed: CBAOFS successfully identifies a compact 8-feature subset using wrapper-based evaluation with LightGBM fitness, but the ablation shows that this selection serves dimensionality reduction rather than accuracy improvement. The practical value lies in reduced inference cost, improved interpretability, and the identification of the most informative feature subset for domain understanding. Gap G2 (adaptive chaotic map selection) is addressed algorithmically: the entropy-guided ensemble successfully implements adaptive selection, as confirmed by the convergence plots showing active switching among all four maps. However, the ablation results (A6-FixedMap matching A1-Full under controlled seeds) limit the measurable benefit. Gap G3 (metaheuristic HP optimisation for TabNet) is implemented with CSCA achieving $\mathcal{G}^* = 0.8699$, though A4-NoOpt's essentially identical AUC suggests the stacking architecture's robustness absorbs hyperparameter sensitivity. Gap G4 (heterogeneous stacking) is the strongest contribution: the four-model stacking ensemble provides the clearest and largest improvement (+0.71pp AUC over standalone TabNet), confirming that combining deep tabular and tree-based models through meta-learning is effective for imbalanced tabular classification. Gap G5 (statistical validation) is comprehensively addressed with the five-seed protocol, Friedman testing, Wilcoxon tests, Cohen's d, bootstrap CIs, seven-variant ablation, scalability analysis, and without-duration sensitivity assessment.

6.3 Comparison with Prior Work

The AUC-ROC of 0.9507 is competitive with the best results reported in the literature on the UCI Bank Marketing dataset when call duration is included. Moro et al. [1] reported AUC values around 0.93 with neural networks. Zhang et al. [5] achieved similar performance with stacking ensembles. Direct numerical comparison across studies should be interpreted cautiously, as differences in preprocessing (e.g., whether `pdays` is binarised), data splitting (temporal vs. random), feature engineering choices, and evaluation protocol (single-run vs. multi-run) can produce substantial variation in reported metrics. The without-duration AUC of 0.8201 provides a more deployment-relevant benchmark that future work on this dataset should report alongside the duration-inclusive figure.

6.4 Limitations and Threats to Validity

Several limitations qualify the conclusions drawn from these experiments. First, the evaluation is conducted on a single benchmark dataset (UCI Bank Marketing); generalisability to other tabular classification problems (credit scoring, churn prediction,

fraud detection) remains to be established through multi-benchmark evaluation. Second, the dataset originates from 2008–2010, and the macroeconomic features may not transfer to contemporary banking environments with different interest rate regimes, employment conditions, and regulatory frameworks. Third, the CSCA hyperparameter optimisation uses a modest budget of 30 evaluations, constrained by CPU training time; a larger search budget on GPU hardware could potentially discover superior configurations, though the ablation suggests the performance ceiling is constrained by data and ensemble architecture rather than hyperparameter precision. Fourth, the minimum achievable p-value of 0.03125 for the Wilcoxon signed-rank test with five runs limits the statistical resolution for distinguishing between closely performing methods—the CatBoost comparison ($p = 0.3125$) would benefit from additional runs. Fifth, the computational cost of the full pipeline (approximately 310 minutes on CPU) is substantially higher than that of individual gradient boosting models (1–2 seconds), raising practical cost-benefit questions. For operational deployment, the precomputed feature mask and hyperparameters could be applied directly with a pre-trained stacking ensemble, reducing the inference pipeline to milliseconds per instance. Sixth, the deterministic nature of chaotic maps under fixed seeds causes variants A6 and A7 to produce identical results to A1, limiting the ability to assess the contribution of adaptive map selection within this experimental protocol; multi-start evaluation with diverse initialisations across a wider seed range would provide better estimates.

7 Conclusion

This paper presented CHIMERA-Tab, a framework integrating chaotic metaheuristic optimisation with deep tabular learning and gradient boosting ensembles for bank marketing prediction. The CBAOFS reduces features from 21 to 8 (61.9%), the CSCA optimises 8 TabNet hyperparameters, and the four-model stacking ensemble achieves AUC-ROC 0.9507 ± 0.0026 and F1-Macro 0.7949 ± 0.0056 across five runs, significantly outperforming eight of nine baselines. The seven-variant ablation identifies stacking as the primary driver, while SHAP and chaotic sensitivity analyses provide interpretability. The without-duration AUC of 0.8201 quantifies the deployment-realistic performance gap.

The findings carry broader implications beyond bank marketing. The demonstrated effectiveness of heterogeneous stacking across model families—combining attention-based deep tabular learning with gradient boosting through learned meta-learner weights—suggests a general strategy for imbalanced tabular classification that avoids the "deep learning versus trees" dichotomy by exploiting both. The transparent ablation methodology, which honestly identifies the contribution hierarchy of each pipeline component, provides a template for rigorous evaluation that the tabular learning community would benefit from adopting more widely.

Future work will extend CHIMERA-Tab along four directions. First, multi-benchmark evaluation on datasets spanning credit scoring, churn prediction, fraud detection, and clinical diagnosis will establish the generalisability of the stacking architecture across domains with different characteristics. Second, more expressive meta-learners—such as gradient boosting or small neural networks—could capture

nonlinear interactions among base-model predictions that logistic regression cannot model. Third, a duration-free variant with enriched pre-call features from external data sources (economic calendars, client digital footprints) could narrow the 13.1pp deployment gap. Fourth, deploying CHIMERA-Tab on a live telemarketing campaign with A/B testing against the current targeting model would provide the definitive assessment of practical value.

Data Availability

The dataset analyzed during the current study is publicly available from the UCI Machine Learning Repository as the Bank Marketing dataset: https://archive.ics.uci.edu/ml/datasets/bank+marketing. The processed data and source code used to generate the reported results are available from the corresponding author upon reasonable request.

Declaration of competing interest

The authors declare no conflict of interest.

Funding

The author did not receive any Funding for this work.

Ethics approval

Not applicable. This study uses a publicly available, anonymised benchmark dataset.

Code availability

The implementation details are described in Section 4.4.

References

[1] Moro, S., Cortez, P., Rita, P.: A data-driven approach to predict the success of bank telemarketing. Decision Support Systems **62**, 22–31 (2014) https://doi.org/10.1016/j.dss.2014.03.001

[2] Xie, C., Zhang, J.-L., Zhu, Y., Xiong, B., Wang, G.-J.: How to improve the success of bank telemarketing? Prediction and interpretability analysis based on machine learning. Computers and Industrial Engineering **175**, 108874 (2023) https://doi.org/10.1016/j.cie.2022.108874

[3] Thabtah, F., Hammoud, S., Kamalov, F., Gonsalves, A.: Data imbalance in classification: Experimental evaluation. Information Sciences **513**, 429–441 (2020) https://doi.org/10.1016/j.ins.2019.11.004

[4] Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: SMOTE: Synthetic minority over-sampling technique. Journal of Artificial Intelligence Research **16**, 321–357 (2002) https://doi.org/10.1613/jair.953

[5] Zhang, Z., Hao, L., Linghu, Y., Yi, H.: A novel ensemble approach for estimating the competency of bank telemarketing. Scientific Reports **14**, 959 (2024) https://doi.org/10.1038/s41598-023-47177-7

[6] Chen, T., Guestrin, C.: XGBoost: A scalable tree boosting system. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 785–794. ACM, ??? (2016). https://doi.org/10.1145/2939672.2939785

[7] Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., Liu, T.-Y.: LightGBM: A highly efficient gradient boosting decision tree. In: Advances in Neural Information Processing Systems, vol. 30, pp. 3146–3154. Curran Associates, Inc., ??? (2017)

[8] Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A.V., Gulin, A.: CatBoost: unbiased boosting with categorical features. Advances in Neural Information Processing Systems **31**, 6638–6648 (2018)

[9] Grinsztajn, L., Oyallon, E., Varoquaux, G.: Why do tree-based models still outperform deep learning on typical tabular data? Advances in Neural Information Processing Systems **35**, 507–520 (2022)

[10] Shwartz-Ziv, R., Armon, A.: Tabular data: Deep learning is not all you need. Information Fusion **81**, 84–90 (2022) https://doi.org/10.1016/j.inffus.2021.11.011

[11] Arik, S.O., Pfister, T.: TabNet: Attentive interpretable tabular learning. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, pp. 6679–6687. AAAI Press, ??? (2021). https://doi.org/10.1609/aaai.v35i8.16826

[12] Huang, X., Khetan, A., Cella, M., Belongie, S.: TabTransformer: Tabular data modeling using contextual embeddings. arXiv preprint arXiv:2012.06678 (2020) https://doi.org/10.48550/arXiv.2012.06678

[13] Popov, S., Morozov, S., Babenko, A.: Neural oblivious decision ensembles for deep learning on tabular data. arXiv preprint arXiv:1909.06312 (2020) https://doi.org/10.48550/arXiv.1909.06312

[14] Gorishniy, Y., Rubachev, I., Khrulkov, V., Babenko, A.: Revisiting deep learning models for tabular data. Advances in Neural Information Processing Systems **34**, 18932–18943 (2021)

[15] Borisov, V., Leemann, T., Seßler, K., Haug, J., Pawelczyk, M., Kasneci, G.: Deep

neural networks and tabular data: A survey. IEEE Transactions on Neural Networks and Learning Systems **35**(6), 7499–7519 (2022) https://doi.org/10.1109/TNNLS.2022.3229161

[16] Abdelrazek, M., Abd Elaziz, M., El-Baz, A.H.: CDMO: Chaotic dwarf mongoose optimization algorithm for feature selection. Scientific Reports **14**, 590 (2024) https://doi.org/10.1038/s41598-023-50959-8

[17] Nadimi-Shahraki, M.H., Taghian, S., Mirjalili, S., Abualigah, L.: Binary aquila optimizer for selecting effective features from medical data: A COVID-19 case study. Mathematics **10**(11), 1929 (2022) https://doi.org/10.3390/math10111929

[18] May, R.M.: Simple mathematical models with very complicated dynamics. Nature **261**(5560), 459–467 (1976) https://doi.org/10.1038/261459a0

[19] Altay, O., Varol Altay, E.: A novel chaotic transient search optimization algorithm for global optimization, real-world engineering problems and feature selection. PeerJ Computer Science **9**, 1526 (2023) https://doi.org/10.7717/peerj-cs.1526

[20] Gopi, S., Mohapatra, P.: Chaotic aquila optimization algorithm for solving global optimization and engineering problems. Alexandria Engineering Journal **108**, 135–157 (2024) https://doi.org/10.1016/j.aej.2024.07.042

[21] Abualigah, L., Yousri, D., Abd Elaziz, M., Ewees, A.A., Al-qaness, M.A.A., Gandomi, A.H.: Aquila optimizer: A novel meta-heuristic optimization algorithm. Computers and Industrial Engineering **157**, 107250 (2021) https://doi.org/10.1016/j.cie.2021.107250

[22] Mirjalili, S.: SCA: A sine cosine algorithm for solving optimization problems. Knowledge-Based Systems **96**, 120–133 (2016) https://doi.org/10.1016/j.knosys.2015.12.022

[23] Wolpert, D.H., Macready, W.G.: No free lunch theorems for optimization. IEEE Transactions on Evolutionary Computation **1**(1), 67–82 (1997) https://doi.org/10.1109/4235.585893

[24] Pashaei, E.: Mutation-based binary aquila optimizer for gene selection in cancer classification. Computational Biology and Chemistry **101**, 107767 (2022) https://doi.org/10.1016/j.compbiolchem.2022.107767

[25] Abd El-Mageed, A.A., Abohany, A.A., Elashry, A.: Effective feature selection strategy for supervised classification based on an improved binary Aquila optimization algorithm. Computers and Industrial Engineering **181**, 109300 (2023) https://doi.org/10.1016/j.cie.2023.109300

[26] Abbassi, A., Abbassi, R., Jerbi, H., Alturki, M., Oueslati, F., Buyuker, T., Kridli, G.T.: An efficient chaotic sine cosine algorithm for global optimization and big

data challenges. Neural Computing and Applications **33**, 17561–17584 (2021) https://doi.org/10.1007/s00521-021-06346-z

[27] Kaisar, S., Rashid, M.M., Chowdhury, A., Shams, R., Reza, S.: Enhancing telemarketing success using ensemble-based online learning models. Big Data Mining and Analytics **7**(2), 472–487 (2023) https://doi.org/10.26599/BDMA.2023.9020041

[28] Li, Y., Geng, Y., Sheng, H.: An improved mountain gazelle optimizer based on chaotic map and spiral disturbance for medical feature selection. PLoS ONE **19**(7), 0307288 (2024) https://doi.org/10.1371/journal.pone.0307288

[29] Sasmal, B., Hussien, A.G., Das, A., Dhal, K.G.: A comprehensive survey on aquila optimizer. Archives of Computational Methods in Engineering **30**(7), 4449–4476 (2023) https://doi.org/10.1007/s11831-023-09945-6

[30] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Köpf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S.: PyTorch: An imperative style, high-performance deep learning library. Advances in Neural Information Processing Systems **32**, 8024–8035 (2019)

[31] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E.: Scikit-learn: Machine learning in Python. Journal of Machine Learning Research **12**, 2825–2830 (2011)

[32] Lundberg, S.M., Lee, S.-I.: A unified approach to interpreting model predictions. In: Advances in Neural Information Processing Systems, vol. 30, pp. 4765–4774. Curran Associates, Inc., ??? (2017)