Babylonian Journal of Machine Learning
Vol.2026, **pp**. 313–332
DOI: https://doi.org/10.58496/BJML/2026/018; ISSN: 3006-5429
https://mesopotamian.press/journals/index.php/BJML

Research Article

Symmetric PSO Tuning of Deep Tabular Networks versus Grid-Search-Optimized Gradient Boosting for FAO-56 Surrogate Modeling of Reference Evapotranspiration under Progressive Climatic Input Scarcity: A Case Study in Bechar, Algeria

Sarra Gourari [1,2,*], Wafa Difallah [2,3], Belkacem Draoui [3]

[1] *Laboratory of Information Processing and Telecommunication (LTIT), Department of Mathematics and Computer Science, Faculty of Exact Sciences, Tahri Mohammed University of Bechar, Bechar, Algeria*

[2] *Innovation in Informatics and Engineering Laboratory (INIE LAB), Department of Mathematics and Computer Science, Faculty of Exact Sciences, Tahri Mohammed University of Bechar, Bechar, Algeria*.

[3] *Laboratory of Energetic in Arid Zones (ENERGARID), Faculty of Technology, Tahri Mohammed University of Bechar, Bechar, Algeria*.

ARTICLE INFO

Article History

Received 28 Aug 2026
Revised 01 Nov 2026
Accepted 04 Nov 2026
Published 12 Nov 2026

Keywords

Reference evapotranspiration (ET0),

Machine learning,

Particle Swarm Optimization (PSO),

TabNet,

XGBoost.

ABSTRACT

Irrigated agriculture in arid regions have increased demand for accurate, low-cost estimation of reference evapotranspiration (ET0). Although the FAO-56 Penman–Monteith equation remains the reference method, its reliance on a complete meteorological record limits its applicability at automated stations with partial instrumentation, motivating machine-learning alternatives combined with metaheuristic tuning, such as Particle Swarm Optimization (PSO). However, comparative studies typically fine-tune only the proposed model, leaving baselines at their default settings. In order to close this gap, this study creates a transparent benchmarking protocol wherein Random Forest and XGBoost baselines are independently optimized via grid search, and the two deep tabular architectures, TabNet and Kolmogorov–Arnold Networks (KAN), receive an identical, symmetric Particle Swarm Optimization (PSO) budget. Daily meteorological data from an arid-zone station in Bechar, Algeria, are used to assess the models in three more enriched input scenarios: temperature alone, temperature and radiation, and the complete climatic set. Friedman-Nemenyi rank-based testing, ten random seeds, and a single chronological split were utilized in the primary comparison. Mean test R^2 rose from 0.843 to 0.894 to 0.993 across the three scenarios. Adding wind speed, wind direction, and relative humidity jointly reduced mean RMSE by 75%, compared with 18% for solar radiation alone; a feature ablation of three models attributed most of this gain to wind speed, with relative humidity contributing mainly in combination with it and wind direction giving no benefit. Friedman and Nemenyi tests ranked XGBoost first in S1 and S2. Under the full feature set, PSO-tuned TabNet matched XGBoost's average rank exactly (1.9 for both), and their pairwise comparison was non-significant ($p = 1.000$), indicating no detected difference in central tendency rather than demonstrated equivalence. These rank-based results are conditional on the seed set. An expanding-window diagnostic (one pass per fold) revealed that both top performing models gave strong performance with a mean R^2 value of 0.972 for XGBoost and 0.907 for the PSO tuned TabNet model. However, PSO tuning led to an obvious decrease in accuracy for the first fold compared to XGBoost (R^2 =0.663 vs. 0.952). These findings indicate that a grid-search-tuned gradient boosting model remains difficult to outperform under constrained inputs, and that metaheuristic tuning benefits deep tabular architectures conditionally rather than universally.

1. INTRODUCTION

Water scarcity has become one of the most pressing challenges to global food security, as agriculture accounts for roughly 70% of global freshwater withdrawals and irrigation effectiveness depends critically on accurate crop water demand

*Corresponding author. Email: gourari.sarra@univ-bechar.dz

forecasts [1]. In agricultural water management, irrigation scheduling, and hydrological evaluation, reference evapotranspiration (ET0) is essential. The reference evapotranspiration (ET0) was defined by Doorenbos and Pruitt [2] as "the rate of evapotranspiration from an extensive surface of 0.08–0.15 m tall, green grass cover of uniform height, actively growing, completely shading the ground, and not short of water." Allen et al. [3] later redefined it as the evapotranspiration rate from a hypothetical reference crop with an assumed height of 0.12 m, a fixed surface resistance of 70 s m^{-1}, and an albedo of 0.23, closely resembling a well-watered, actively growing, uniform grass surface. This is especially true in water-limited areas, where even little calculation mistakes can result in major inefficiencies in irrigation planning [3], [4]. Since agricultural water demand is very sensitive to climatic fluctuations and water resources are still limited and more susceptible to climate stress, it is crucial to accurately anticipate ET0 in arid and hyper-arid locations [5]. For this reason, agro-meteorology and environmental modeling continue to place a high priority on accurate and practical ET0 calculation [6], [7]. Although the FAO-56 Penman–Monteith equation is widely recognized as the standard physical method for estimating reference evapotranspiration (ET0), the requirement for several meteorological variables, such as air temperature, relative humidity, wind speed, and solar or net radiation, frequently limits its operational application [8]. One or more of the meteorological variables required by the FAO-56 Penman–Monteith method may be unavailable, incomplete, or inconsistent over extended periods of time at many operational stations, particularly in data-scarce regions [8]. This limitation has motivated extensive research into two complementary alternatives: simplified empirical formulations requiring fewer inputs and data-driven machine learning techniques capable of approximating FAO-56 behavior directly from the climatic variables that are locally available [9],[10]. Among these, classical empirical formulations remain attractive for their simplicity and minimal data requirements; however, their predictive accuracy is highly sensitive to local calibration and to the specific meteorological conditions of the target region, which limits their transferability across climates. In the Densu River Basin of Ghana, Adjei et al. [11] calibrated the Hargreaves–Samani and Priestley–Taylor equations against the FAO-56 Penman–Monteith reference. They discovered that, once locally calibrated, the Priestley–Taylor formulation offered a better match than the temperature-based Hargreaves model. Zhang et al. [12] assessed the suitability of several evapotranspiration models for various farmland types and verified that the underlying water-consumption regime has a significant impact on model transferability. Figueiró et al. [13] recalibrated the Hargreaves–Samani equation for Northern Minas Gerais, Brazil, using a 30-year dataset that included temperature, wind speed, relative humidity, and sunshine duration. Their results showed that the locally calibrated model achieved higher accuracy than both the original Hargreaves–Samani equation and the FAO-56 Penman–Monteith method under local conditions. To estimate sesame crop water requirements, Bhosale et al. [14] assessed the performance of a modified Penman–Monteith equation alongside the FAO Radiation, Priestley–Taylor, Hargreaves–Samani, and Pan Evaporation methods. Among the evaluated approaches, the modified Penman–Monteith formulation produced the most accurate results for the study area. Difallah et al. [15] evaluated AI-supported (MLP, DNN) and traditional (ARX, NLARX) system-identification models for ET0, and found that while all models had near-perfect fits (>98%), AI integration did not perform better than more straightforward linear identification.

Three evapotranspiration techniques—EVAPO, AgSAT, and Evapotranspiration Calculator—were used in [16] to determine ET0. The Thornthwaite model was used as a benchmark in this investigation to assess the three strategies. The study in [17] sought to estimate crop evapotranspiration (ETC). The Penman-Monteith model continued to be the best model for ET0 estimation after the estimation accuracy of thirty ET0 micrometeorological models was assessed. Three Penman-Monteith models were applied in arid and semi-arid regions in [18].

This study highlights the importance of data frequency. 5 different deep learning algorithms were used in this work including CatBoost, LightGBM, LSTM, GRU and Bi-LSTM [19]. The findings show that the CatBoost model performed better than the others when the models were applied to an irrigation dataset for Moringa oleifera in semi-arid regions. Artificial neural networks (ANN) were found to be the most effective method for ET0 calculation when compared to multiple linear regression (MLR) and random forest (RF) [20]. Support Vector Regression (SVR) outperforms Random Forest (RF) in predicting ET0 using rainfall, temperature, wind speed, and relative humidity data, as demonstrated by the authors of [21]. In [22], the neural network's ability to forecast ET0 based on the Penman-Monteith model at 14 weather stations was optimized using a genetic algorithm. The authors in [23] also employed a genetic algorithm in conjunction with differential evolution and particle swarm optimization techniques to determine ET0. Eleven ET0 prediction models were used in the study to examine the effects of climatic conditions. Local evapotranspiration was estimated in [24] using Random Forest Regressor (RFR), Decision Tree Regressor (DTR), and Support Vector Regressor (SVR). RFR was the best ET0 prediction model, according to R2, RMSE, MAE, and MAPE values. In [25], the ET0 estimate procedure was made simpler by using the wavelet neural network (WNN) and the Group Method of Data Handling.

The accuracy of three ET0 forecasting techniques spanning the FAO Penman–Monteith (FAO-PM) and Hargreaves–Samani (HS) models was covered in the study in [26]. Deep learning, machine learning, and statistical approaches were among the models that were examined. The findings showed that while the statistical models worked well for HS, the machine learning and deep learning models agreed well with FAO-PM. Researchers in [27] presented a Hybrid

Convolutional Neural Network with Long Short-Term Memory (H-ConvLSTM). With temperature and relative humidity as system inputs, the system outperformed existing deep learning algorithms, such as Recurrent Neural Network (RNN), Gated Recurrent Unit (GRU), and Long Short-Term Memory (LSTM). The gray wolf optimizer (GWO) was used in [28] to enhance Random Forests (RFs), Light Gradient Boosting (LGB), and Extreme Gradient Boosting (XGB) algorithms for an accurate ET0 estimate.

A trend in the literature has shown that proposed models for ET0 tends to be heavily optimised whilst baseline models are not. Additionally, meta-heuristics were not commonly used. TabNet and KAN were given equal PSO budgets. This means any difference in performance between the two deep learning models is due to their inherent model capabilities. The random forest and xgboost models were optimised using grid search. (See section 3.4 for discussion on these models). Figure 2 shows that there was a clear difference between the performance of TabNet and KAN. The main contributions of this study are outlined below:

1. Six regression models—Random Forest, XGBoost, standard TabNet, PSO-tuned TabNet, standard KAN, and PSO-tuned KAN—were benchmarked across three progressively reduced meteorological input scenarios representing realistic data-scarcity conditions in arid regions.
2. The statistical significance of PSO tuning for TabNet and KAN is assessed with the Friedman–Nemenyi post-hoc framework on seed-matched runs.
3. Temporal behavior is further examined through a retrospective expanding-window diagnostic, while the scenario-based accuracy decomposition (Section 3.1) and a factorial feature ablation (Section 3.5, Table 8) quantify the contribution of the individual climatic variables to estimation accuracy.
4. A transparent, symmetric-budget PSO tuning protocol for competing deep tabular architectures is proposed as a methodological reference for future ET0 studies, alongside explicit reporting of the markedly different search scale used for the grid-search-tuned baselines.

2. MATERIALS AND METHODS

2.1 Study Area and Climate

The meteorological data were collected at an experimental station operated by the Laboratory of Energetics in Arid Zones (ENERGARID), Tahri Mohammed University of Bechar, Algeria (Figure 1), at approximately 31°37′ N and 2°14′ W. The region is characterized by a hyper-arid climate, making it an ideal case study for evaluating reference evapotranspiration (ET0) estimation methods and irrigation management strategies.



Fig 1. Study area localization

2.2 Dataset Description

Bechar is characterized by a hyper-arid climate shaped by its surrounding Saharan desert landscape and limited water resources [29], [30]. The obtained data set includes 1,096 samples of 5 meteorological variables (air temperature T [°C], relative humidity Hr [%], wind speed Vv [m/s], wind direction Dv [°], and global radiation Rg-h [W/m^2]) measured on subsequent days from 1 January 2023 to 31 December 2025. The evolution of these variables during the study period is shown on Figure 2. Air temperature, global radiation and reference evapotranspiration demonstrate significant seasonal dependence with maximum values in the summer and minimum values in the winter. On the contrary, relative humidity demonstrates the opposite pattern. Wind speed and wind direction show high variability from one day to another, which is characteristic for arid climate. Wind direction (Dv) is a circular variable and 0° is the same as 360°. Therefore, to avoid any bias, it was converted to a sine and cosine transformation (sin θ and cos θ). This resulted in the feature vector S3 containing 6 input variables for the model.

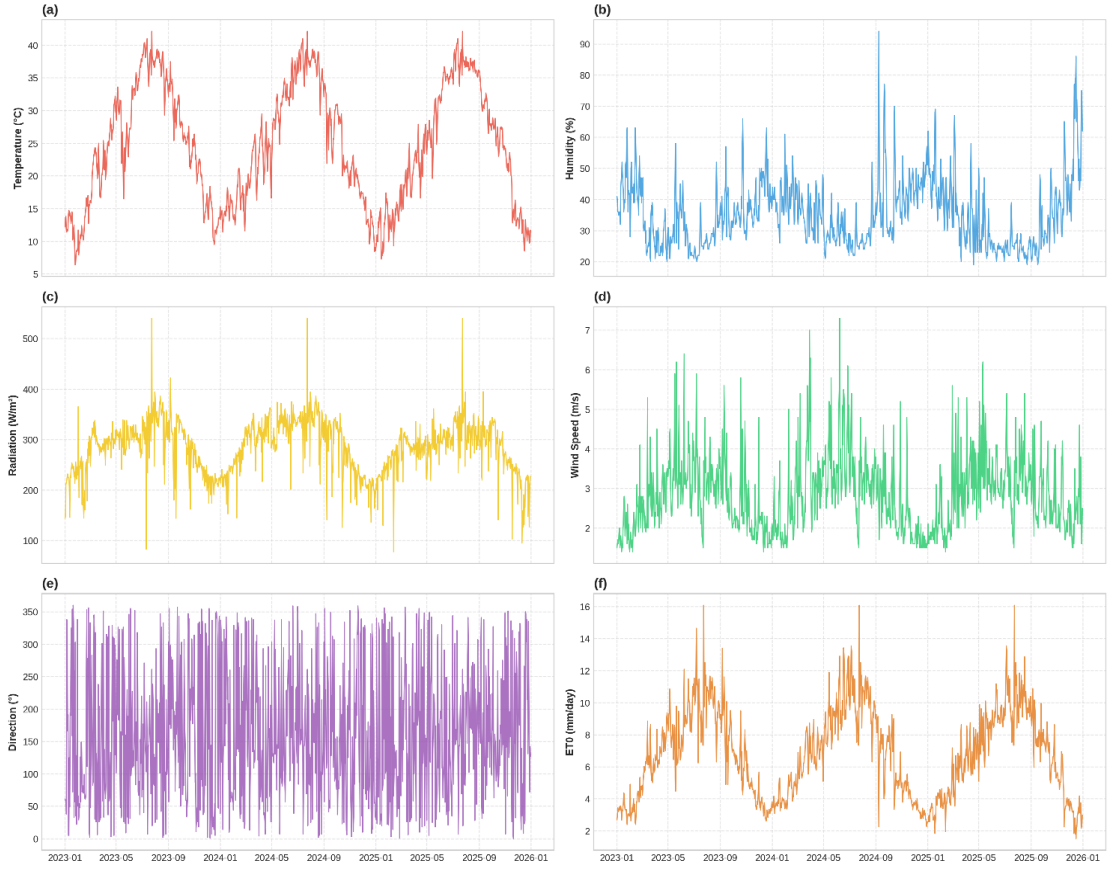


Fig 2. Time-series profiles of the meteorological input features and target parameter: (a) Temperature (T, °C), (b) Humidity (Hr, %), (c) Radiation (Rg-h, W/m²), (d) Wind Speed (Vv, m/s), (e) Direction (Dv, °), (f) ET0 (mm/day) in the study area.

2.3 Reference Evapotranspiration Computation (FAO-56 Penman–Monteith)

Direct measurement of reference evapotranspiration (ET0) can be obtained using weighing lysimeters, which is the most accurate method. However, this method requires complex equipment, extensive maintenance and high costs and is therefore not suitable for continuous long-term monitoring, particularly in arid regions. Instead, ET0 was calculated on a daily basis using the internationally accepted FAO-56 Penman-Monteith equation [3] recommended by the Food and Agriculture Organization (FAO). This has been implemented in the python package pyet.

$$ET_0 = \frac{\left(0.408\,\Delta(R_n - G) + \gamma\frac{900}{T+273}u_2(e_s - e_a)\right)}{(\Delta + \gamma(1 + 0.34\,u_2))} \quad (1)$$

where:

R_n: *net radiation* $(MJm^{-2}day^{-1})$

G: $soil\ heat\ flux\ density\ (assumed\ negligible\ at\ the, G \approx 0)$

T: $mean\ daily\ air\ temperature(°C)$

u^2: $wind\ speed\ at\ 2m\ height(ms^{-1})$

e_s and e_a: saturation and actual vapor pressures (kPa)

Δ : $slope\ of\ the\ saturation\ vapor\ pressure\ curve$

γ : $psychrometric\ constant.$

The FAO-56 Penman–Monteith computation [3] used the documented coordinates of the Bechar station (31.62° N, 747 m a.s.l.) to derive extraterrestrial and clear-sky radiation. Global horizontal solar radiation was converted from W m^{-2} to MJ m^{-2} day^{-1} using the standard factor 0.0864. Because only daily mean temperature and relative humidity were available, saturation vapor pressure was estimated using the FAO-56 mean-value formulation for stations lacking Tmax/Tmin or RHmax/RHmin records [3]. Daily wind speed (Vv) directly represented the standard 2 m reference wind speed (u_2). Wind direction (Dv) was not used in the physical computation and is retained solely as an additional candidate feature for the machine learning models. As ET0 is determined by a number of the climatic variables included in the model inputs as a deterministic function, the high accuracy achieved when all features are used (Section 3.1) can be attributed to the fact that there is a mathematical relationship between them. As wind speed is an input to the FAO-56 method, it is not possible to have computable states for S1 and S2. Therefore, this study focuses on surrogate modelling of the FAO-56 method with incomplete inputs. Three scenarios were created to ensure the models were robust to real-world data availability issues. The dataset was divided into training (70%), validation (15%) and test sets (15%) using chronological ordering rather than random shuffling. Any continuous features were standardised using training data only to give zero mean and unit variance. Scaling factors were calculated based solely on the training data and then applied to the validation and test data.

2.4 Proposed System Architecture

The experiment workflow used in this study is depicted in Figure 3. The initial step is preprocessing of the meteorological raw variables (Dv, Vv, T, Hr, Rg-h). After that three different input scenarios S1, S2 and S3 are obtained. The data set is divided into three parts: training, validation and testing. From each scenario, six different models are developed: Random Forest and XGBoost models optimized by means of Grid Search CV and Time Series Split; default TabNet and KAN (with default hyperparameter values); and TabNet and KAN optimized by means of PSO algorithm (20 particles, 20 iterations). Models' performance metrics were calculated, such as R^2, RMSE, MAE. A gap between train and test data by R^2 metric was calculated as well to diagnosed overfitting. Because the training of stochastic algorithms — including neural weight initialization, mini-batch ordering, and bootstrap sampling in ensemble methods — is inherently non-deterministic, a single training run does not necessarily represent the expected performance of a model and can, in some cases, reverse the apparent ranking between competing methods [31]. Each model configuration was therefore trained and evaluated over ten random seeds, with all reported metrics expressed as mean ± standard deviation to reflect both expected performance and its variability. The seed values used were {42, 101, 202, 303, 404, 505, 606, 707, 808, 909}, allowing exact reproducibility of the reported results. In the Friedman test (Section 3.2), each seed-matched run constitutes one block: for every scenario, the six models are ranked on test RMSE within each of the ten runs ($k = 6$, $N = 10$). The seeds are replicate runs on one fixed test window, not independent problem instances; behavior across successive time windows is examined separately in a retrospective expanding-window diagnostic reported in Section 3.4. Paired comparisons across models use identical seeds to control for shared initialization and sampling variance, increasing the statistical power of the paired t-test relative to unpaired comparisons. For completeness, pairwise paired t-tests with Holm correction were also computed for each model pair and scenario. Because they compare mean RMSE magnitudes and XGBoost has zero variance, they can flag differences that the rank-based Nemenyi procedure does not (for example, XGBoost versus Random Forest in S1); they are not used for the conclusions and are not tabulated, and the rank-based results of Section 3.2 are reported instead.

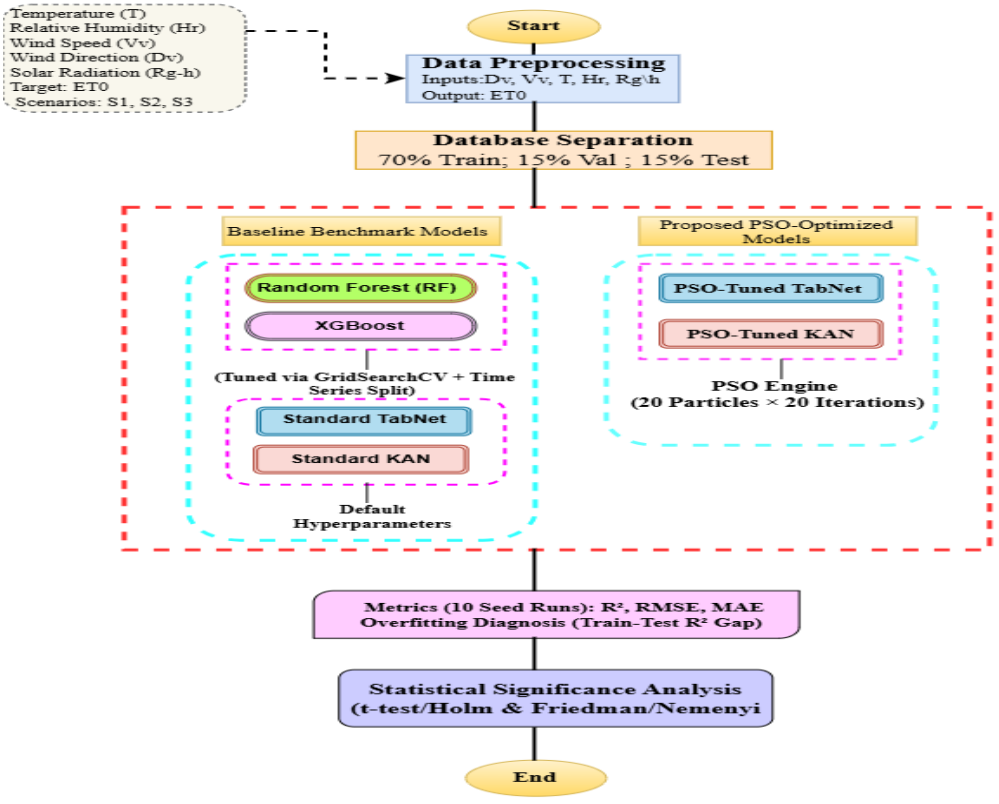


Fig 3. Design of the proposed methodology.

2.5 Machine Learning Models

2.5.1 Random Forest regression

The Random Forest (RF) algorithm is a classifier combination that builds an ensemble of decision trees based on Classification and Regression Trees (CART) without pruning [32]. To create diversity among trees, each tree is constructed using a random subset of the input characteristics and a random subset of the data. The number of trees (N_{tree}) and the number of randomly chosen features (M_{try}) utilized for splitting at each node are two crucial RF modeling parameters. Eq. 2 represents the RF prediction for regression issues, which is derived by averaging the outputs of each decision tree.

$$H(x) = \frac{1}{N}\sum_{n=1}^{N} h(x, \theta_n) \quad (2)$$

where x is the input vector, n = 1,..., N is the number of regression trees, and θ is an independent, uniformly distributed random vector that the prediction trees utilize as a numerical value.

2.5.2 XGBoost

Extreme Gradient Boosting is a model that effectively uses Gradient Boosting Machines. This model is based on the boosting principle, which builds an ensemble of decision trees one after the other while each model attempts to correct the errors of its predecessor. This model is often characterized by its performance, scalability, and speed, as well as its ability to extract insights from previous errors and optimize hyperparameters [33]. The XGBoost model iteratively builds an ensemble of K decision trees, where the final prediction y^i for a given sample i is the sum of the predictions from all trees, expressed as Eq. 3.

$$\hat{y}_i = \sum_{k=1}^{K} f_k(x_i), f_k \in F \quad (3)$$

2.5.3 TabNet: Attention-Based Deep Tabular Learning

Google Cloud proposed the TabNet model [34], which combines the benefits of DNNs and tree models. Sequential multistep processing is the foundation of TabNet, a single deep learning model [35]. This one deep architecture enhances

the ability to learn high-dimensional features and helps in feature selection [36]. A D-dimensional feature vector is processed in each n^{th} step, with each step's output going to a Feature Transformer block. The transformer block has several levels, some specific to a decision step and others common to all decision processes. Each block has a Gated Linear Unit (GLU) activation, a batch normalization layer, and fully connected layers. In the original TabNet, the GLU is additionally connected to a normalized residual link, which helps stabilize variance across the network. This multi-layered block improves the network's parameter efficiency and helps with feature selection. The TabNet design is described in depth in Figure 4, where the encoder and decoder's structural components are the feature transformer and the attentive transformer. The main job of the Encoder part of the TabNet model (Figure 4(a)) is to selectively focus on the most pertinent qualities for the prediction task by processing the input features through a series of decision stages. The feature transformer and the attentive transformer are the two primary parts of each decision step. The feature transformer uses a variety of FC layers, BN, and Gated Linear Units (GLU) to improve the feature representation inside the TabNet framework. Together, these layers map input features into a more practical encoded space. In the original TabNet, residual connections are scaled by $\sqrt{0.5}$ to preserve a stable variance across the network. Additionally, the attentive transformer promotes sparse feature selection through a learnable mask based on the sparsemax function. The mathematical representation of this mask is given by:

$$M[i] = Sparsemax(Z[i] \cdot a[i-1]) \quad (4)$$

where $M[i]$ is the mask for step i, $Z[i]$ is the transformed feature representation, and $a[i-1]$ is the attention from the previous step.

The next component is the TabNet decoder. In the self-supervised learning architecture of the TabNet model, the decoder is essential. It is specifically designed to use the encoded representations to recreate tabular features. Each decision step in the sequential procedure invokes fully connected layers to improve the encoder output (Figure 4(b)). A masking array informs the reconstruction of features, enabling the model to infer missing features and concentrate on known ones. In the implementation used in this study (a simplified TabNet-style regressor), softmax replaces sparsemax in the attentive transformer (so the masks are soft rather than sparse), a shared input projection feeds a single GLU block per decision step without residual scaling, no sparsity regularization is used, ordinary batch normalization replaces ghost batch normalization, and the decoder (used for self-supervised pre-training) is not employed.

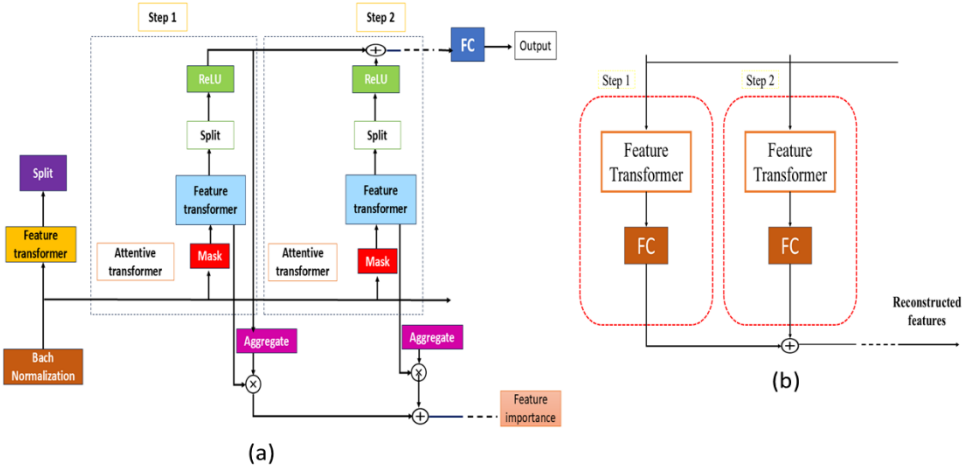


Fig 4. TabNet architecture: (a) encoder and (b) decoder

2.5.4 Kolmogorov–Arnold Networks (KAN)

The Kolmogorov-Arnold representation theorem inspired the innovative neural network architecture known as Kolmogorov-Arnold Networks (KANs) [37], [38], [39]. KANs introduce learnable activation functions on edges (weights), in contrast to conventional Multi-Layer Perceptrons (MLPs), which employ fixed activation functions on neurons. KANs [40], [41] replace each weight with a univariate spline-parameterized function, in place of linear weight matrices. This novel method improves function-approximation accuracy and interpretability. According to the Kolmogorov–Arnold representation theorem, any multivariate continuous function defined over a bounded domain can be represented as a finite combination of continuous univariate functions connected through addition. In the case of a smooth function, the representation can be expressed as follows: $f : [0,1]n \to \mathbb{R}$

$$f(\boldsymbol{x}) = f(x_1, \ldots, x_n) = \sum_{q=1}^{2n+1} \Phi_q \left(\sum_{p=1}^{n} \phi_{q,p}(x_p) \right) \quad (5)$$

where $\phi_{q,p}$: [0,1] →ℝ and Φ_q: ℝ→ℝ. In a sense, they showed that the only true multivariate function is addition, since every other function can be written using univariate functions and sums. The KAN implementation used in this study is a self-contained PyTorch implementation employing radial basis function (RBF, Gaussian) basis functions rather than the B-spline basis functions of the original KAN formulation of Liu et al. [37]. This RBF-KAN variant was adopted for computational efficiency and training stability; it is referred to throughout as 'KAN' for brevity but should be understood as an RBF-based approximation rather than the original spline-based architecture. Each RBF-KAN layer computes $y_j = \sum_{i=1}^{n} \varphi_{ij}(x_i)$, with edge functions $\varphi_{ij}(x) = \alpha_{ij} \cdot \mathrm{SiLU}(x) + \sum_{g=1}^{G} w_{ijg} \cdot \exp\left(-\left(\frac{x-c_g}{h}\right)^2\right)$, where α_{ij} is the base-branch weight and w_{ijg} are the RBF weights (n is the number of inputs of the layer). The G Gaussian centers c_g are equally spaced over [−2, 2] (about ±2 standard deviations of the standardized inputs) and $h = 4/(G - 1)$. The grid is fixed (not adapted during training), the same G is used in both layers, and all weights are initialized from $N(0, 0.1^2)$. There are no bias terms. The network has two such layers (inputs → hidden → 1) with a SiLU activation between them; G = 8 for Standard KAN and G ∈ [4, 16] for PSO-tuned KAN (Section 2.6).

2.6 Hyperparameter Tuning Protocol

To eliminate asymmetric-tuning bias between the two proposed deep architectures, TabNet and KAN were optimized under an identical Particle Swarm Optimization (PSO) search budget, while Standard TabNet and Standard KAN, kept at their default configurations, served as fixed-configuration reference models. Random Forest (RF) and XGBoost were tuned separately via grid search with 3-fold time-series cross-validation on the training set to minimize validation RMSE; this protocol has a markedly different search scale, detailed below and discussed further in Section 3.4. Search spaces encompassed $n_{estimators} \in \{100,150,200\}$ and $max_depth \in \{8,12,16\}$ for RF, $n_{estimators} \in \{100,200\}$, $max_depth \in \{3,5,7\}$, and $learning_rate \in \{0.03,0.05,0.1\}$ for XGBoost.

For TabNet and KAN, Particle Swarm Optimization (PSO; 20 particles, 20 iterations, w=0.6, c_1=c_2=1.5) was executed independently across each input scenario to dynamically align network capacity with feature dimensionality. Fitness was defined as the validation RMSE following a truncated 30-epoch training run (patience = 6), averaged over three random seeds (42, 101, 202). TabNet optimization targeted decision-step width $n_d \in [8,32]$, step count $n_{steps} \in [3,6]$, relaxation factor $\gamma \in [1.0,2.0]$, and learning rate $\in [10^{-3},10^{-1.5}]$. KAN optimization covered hidden width ∈ [8,32], RBF grid points ∈ [4,16], and learning rate $\in [10^{-3},10^{-1}]$. The number of RBF grid points was therefore part of the PSO search space for PSO-tuned KAN (selected values in Table 7), whereas Standard KAN kept the fixed default of 8 grid points. Both deep architectures received identical PSO settings (20 particles, 20 iterations, three seeds per fitness evaluation); their search spaces differ in dimensionality (four parameters for TabNet, three for KAN). Standard TabNet (n_d=n_a=16, n_{steps}=5, γ=1.5) and Standard KAN (hidden = 16, grid points = 8) retained fixed default parameters to quantify the explicit performance gains derived from PSO tuning. Finally, all six configurations were retrained from scratch across 10 random seeds (42–909) on a chronological 70/15/15 train/validation/test split. This characterizes run-to-run variability due to weight initialization.

Selected hyperparameters for each model and scenario are listed in Table 7 (Section 3.4). All networks used the Adam optimizer without dropout or weight decay; TabNet used a batch size of 64 (128 during PSO fitness evaluation), KAN used full-batch training, and the RBF-KAN weights were initialized from $N(0, 0.1^2)$. Standard TabNet was trained for at most 100 epochs (patience 15), PSO-tuned TabNet for at most 150 (patience 20), and both KAN models for at most 150 (patience 20); in every case the checkpoint with the lowest validation RMSE was retained. The longer epoch budget of PSO-tuned TabNet is part of its protocol and is not separated from the effect of its tuned hyperparameters. Inputs were standardized with training-segment statistics only. The ten seeds (42, 101, 202, 303, 404, 505, 606, 707, 808, 909) are replicate runs on one station, one target series and one chronological test window; they characterize training stochasticity and do not quantify uncertainty across stations, climates or periods. XGBoost is deterministic in this setting (fixed split, no subsampling), so its variance is zero by construction and it acts as a constant in the paired comparisons; the seed sweep therefore estimates run-to-run variance only for the five stochastic models. The representative run shown in Figures 6, 10 and 11 is the seed whose PSO-tuned TabNet RMSE is closest to the median across seeds (seed 505). The temporal analysis reported in Section 3.4 used random seed 42. The software environment comprised Python 3.12.13, PyTorch 2.10.0, scikit-learn 1.6.1, XGBoost 3.2.0, SciPy 1.16.3, pandas 2.3.3, and NumPy 2.0.2. Feature contributions were examined through an eight-combination ablation of the three added variables (wind speed, relative humidity, and wind direction, with its sin/cos components treated as one group), alongside separate removal tests for radiation and temperature from S3. XGBoost, Random Forest, and PSO-tuned TabNet were evaluated using S3-selected hyperparameters without subset-specific retuning. Ten seeds were used for Random Forest and PSO-tuned TabNet, while XGBoost underwent one deterministic run. An exact Shapley decomposition of RMSE quantified the contribution of each feature group.

2.7 Evaluation Metrics and Statistical Validation

Three widely used statistical criteria, namely R^2, RMSE, and MAE (Eqs. (6)-(8)) [42], were applied to extensively evaluate and compare the performance of the regression equations for daily ET0 estimation. The comparison between the predicted outputs of the models and the test dataset provides information regarding the accuracy and reliability of the models.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \quad (6)$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \quad (7)$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \quad (8)$$

Where y_i is the actual ET0 value on the i^{th} day, $\hat{y}_i$ is the predicted value using the model developed, $\bar{y}$ is the average of actual ET0 values and n is the number of test samples. The smaller the RMSE and MAE values the better the prediction. The closer the R^2 value is to 1.0 the better the prediction. Model generalizability is further supported when these performance levels are maintained on held-out test data; the expanding-window analysis of Section 3.4 serves only as a retrospective diagnostic.

3. RESULTS AND DISCUSSION

This section assesses Random Forest, XGBoost, TabNet, KAN, and their PSO-hybridized models for daily ET0 estimation across five main axes: predictive performance under climatic scenarios S1–S3 (Section 3.1), statistical significance of model rankings (Section 3.2), prediction diagnostics and overfitting behavior (Section 3.3), temporal behavior in a retrospective expanding-window diagnostic together with computational cost (Section 3.4), and the contribution of individual climatic variables through a factorial feature ablation (Section 3.5); the limitations of the study are summarized in Section 3.6.

3.1 Overall Predictive Performance Across Climatic Scenarios

Table 1 summarizes the predictive performance of the six evaluated models (Random Forest, XGBoost, Standard TabNet, PSO-tuned TabNet, Standard KAN, and PSO-tuned KAN) under three input scenarios of increasing climatic complexity: S1 (temperature only), S2 (temperature and solar radiation, referred to as the thermal path), and S3 (the full set of five climatic variables). This progressive design was adopted to identify the minimum climatic information required to reach an operationally acceptable ET0 estimation accuracy, which is directly relevant for stations where only a partial meteorological record is available. All metrics reported in Table 1 correspond to the mean and standard deviation computed over ten replicate random seeds (N = 10), rather than a single training run, to provide a statistically defensible estimate of model performance. Averaged across the six architectures, mean test R^2 rose from 0.843 (S1) to 0.894 (S2) and 0.993 (S3), while mean RMSE fell from 1.138 to 0.932 and 0.235 mm day^{-1} (Table 1Adding wind speed, wind direction and relative humidity (S2→S3) reduced the mean RMSE (averaged over the six models) by 75%, compared with 18% when solar radiation was added to temperature (S1→S2); these contrasts depend on the order of addition, and the ablation of Section 3.5 shows that radiation and temperature remain essential once the other variables are present. In the three models examined in the ablation, wind speed was the dominant variable among those added to temperature and radiation, which differs from many ET0 studies conducted in humid or sub-humid climates, where solar radiation is frequently reported as a dominant driver. However, it is consistent with sensitivity analyses across multiple climate regimes, which show that wind speed can dominate ET0 sensitivity at arid stations, while net radiation tends to dominate mainly at humid stations [43]. It also agrees with FAO-56 guidance that the aerodynamic term becomes more influential under arid conditions because of the larger vapor pressure deficit. Given the hyper-arid setting of the present station (Bechar, Algeria), this climate-dependent shift in variable importance is physically plausible and is consistent with wind speed being a direct physical input of the FAO-56 formulation itself. XGBoost achieved the highest R^2 in S1 and S2 (0.882, 0.919), while under S3 it showed no detected difference in central tendency from the proposed PSO-tuned TabNet ($p = 1.000$; Section 3.2), both reaching $R^2 = 0.996$. Standard KAN was weakest under reduced inputs ($R^2 = 0.785$–0.836), a gap that closed to under 0.01 in S3. Under S1 the tuned version of TabNet using PSO showed poorer performance than the non-tuned variant (0.843 vs 0.864). This is because, during the short period of 30 epochs used for the PSO fitness function search, over-fitting occurred during validation on the most extreme data-scarce scenario. The performance of all models increased monotonically and the

performance gap between models decreased. This can be seen in Figure 5. The individual contributions of the added variables are quantified by the feature ablation of Section 3.5 (Table 8); the 75% figure is an average over the six models.

TABLE I. COMPREHENSIVE PERFORMANCE EVALUATION OF THE PREDICTIVE MODELS ACROSS THE THREE INPUT CLIMATIC SCENARIOS (S1–S3).

Input scenarios	Model	R^2 Train (Mean ± Std)	R^2 Test (Mean ± Std)	RMSE Test (Mean ± Std)	MAE Test (Mean ± Std)
S1_Temp_Only	Random Forest	0.9322 ± 0.0003	0.8626 ± 0.0020	1.0702 ± 0.0077	0.8509 ± 0.0055
	XGBoost	0.9083 ± 0.0000	0.8815 ± 0.0000	0.9935 ± 0.0000	0.7835 ± 0.0000
	Standard TabNet	0.8739 ± 0.0024	0.8640 ± 0.0075	1.0642 ± 0.0294	0.8469 ± 0.0314
	PSO-tuned TabNet	0.8595 ± 0.0117	0.8434 ± 0.0133	1.1412 ± 0.0478	0.9217 ± 0.0515
	Standard KAN	0.8193 ± 0.0098	0.7847 ± 0.0122	1.3389 ± 0.0375	1.1181 ± 0.0362
	PSO-tuned KAN	0.8421 ± 0.0238	0.8194 ± 0.0329	1.2215 ± 0.1150	0.9890 ± 0.1058
S2_Thermal_Path	Random Forest	0.9866 ± 0.0003	0.9155 ± 0.0007	0.8393 ± 0.0033	0.6179 ± 0.0026
	XGBoost	0.9413 ± 0.0000	0.9187 ± 0.0000	0.8230 ± 0.0000	0.6162 ± 0.0000
	Standard TabNet	0.9097 ± 0.0069	0.9038 ± 0.0095	0.8941 ± 0.0445	0.7000 ± 0.0473
	PSO-tuned TabNet	0.9079 ± 0.0069	0.9050 ± 0.0040	0.8896 ± 0.0186	0.6911 ± 0.0108
	Standard KAN	0.8505 ± 0.0139	0.8357 ± 0.0144	1.1691 ± 0.0518	0.9367 ± 0.0619
	PSO-tuned KAN	0.9119 ± 0.0317	0.8849 ± 0.0197	0.9758 ± 0.0830	0.7488 ± 0.0822
S3_Full_Features	Random Forest	0.9981 ± 0.0001	0.9857 ± 0.0006	0.3457 ± 0.0067	0.2448 ± 0.0049
	XGBoost	0.9988 ± 0.0000	0.9958 ± 0.0000	0.1868 ± 0.0000	0.1431 ± 0.0000
	Standard TabNet	0.9963 ± 0.0016	0.9938 ± 0.0028	0.2218 ± 0.0474	0.1648 ± 0.0261
	PSO-tuned TabNet	0.9977 ± 0.0004	0.9958 ± 0.0011	0.1855 ± 0.0223	0.1391 ± 0.0140
	Standard KAN	0.9956 ± 0.0007	0.9934 ± 0.0017	0.2332 ± 0.0302	0.1672 ± 0.0213
	PSO-tuned KAN	0.9974 ± 0.0005	0.9933 ± 0.0012	0.2352 ± 0.0207	0.1589 ± 0.0091

Note: Unless otherwise stated, all metrics reported in Tables 1, 2 and 3 derive from the same set of ten replicate seeded runs per scenario (Section 2.6). Table 4 synthesizes these ten-seed metrics alongside a separate single-run, retrospective five-fold expanding-window R^2 (final row), which is not part of the ten-seed basis and is detailed fold-by-fold in Table 5. Standard deviations are population standard deviations (ddof = 0).

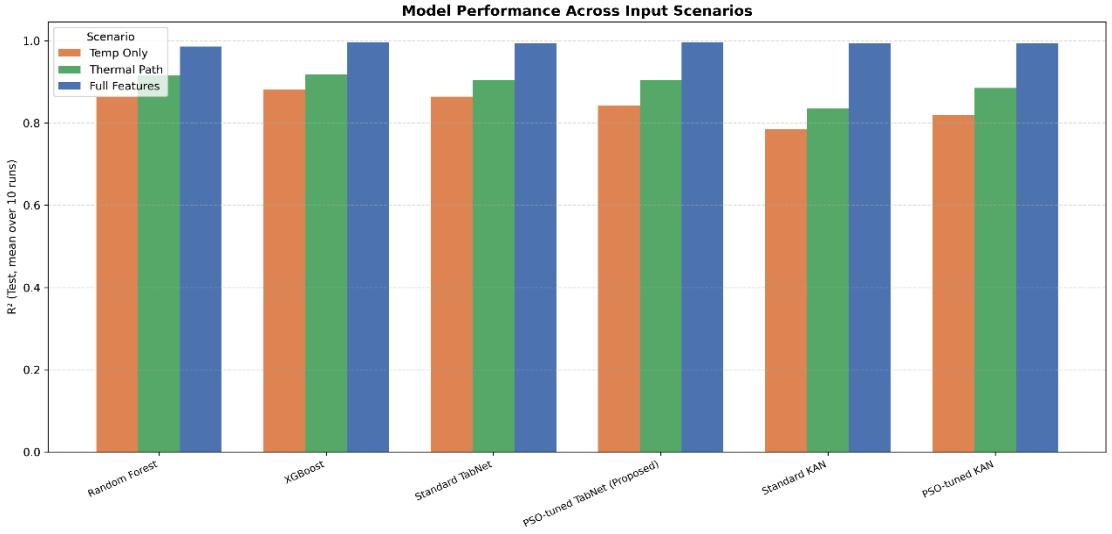


Fig 5. Impact of progressive climatic feature addition on model predictive performance (R^2) across scenarios S1, S2, and S3.

Beyond the aggregate R^2 values, the observed-versus-predicted scatter plots (Figure 6) reveal that all six models track the 1:1 reference line closely across the ET0 range of the test set (approximately 2–15 mm day^{-1}), with no visible funnel-shaped heteroscedasticity that would indicate error growth at high evaporative demand — a reassuring property for irrigation scheduling during peak summer periods. The main qualitative difference is that the Random Forest model has slightly greater spread about the 1:1 line in the upper range of ET0 (> 14 mm day^{-1}). This is reflected in the slightly lower R^2 value (0.986) and increased RMSE value compared to the other five models. The XGBoost and TabNet models have the tightest clustering overall.

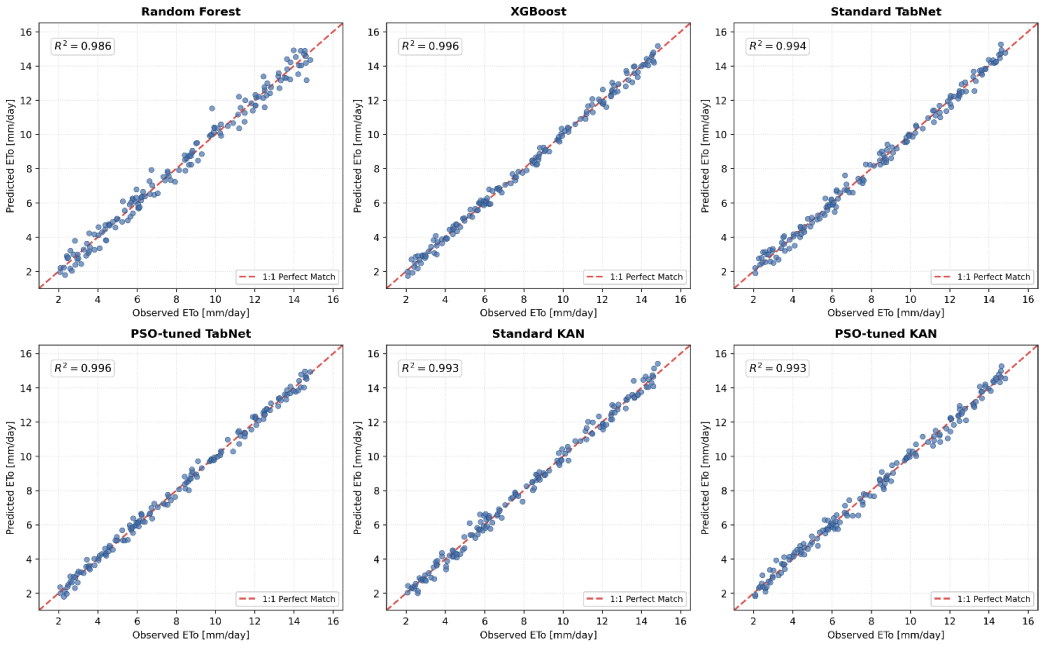


Fig 6. Scatter plots of observed versus predicted ET0 values along the 1:1 line for all models under scenario S3.

3.2 Statistical Significance and Robustness of Model Rankings

Although average performance rankings can be used to understand how well one model performed compared to another, to determine if there is actually any significant difference, a Friedman test was conducted on the ranked RMSEs across the ten runs of each scenario. A Nemenyi post-hoc method and critical difference (CD) plot [44] were then used to identify which methods were significantly different. The Friedman test was chosen over multiple t-tests due to its ability to evaluate all six models together under a single hypothesis, reducing the risk of false positives. However, since the ten seeds are replicate runs using a single station, target series and test window the rankings are only representative of that particular set of seeds. The Demšar methodology is explicitly for independent data sets [44] so the test is being used descriptively. This means that the effective sample size is less than the number of runs.

TABLE II. SUMMARY OF FRIEDMAN TEST STATISTICS (X^2 AND P-VALUE) AND AVERAGE MODEL RANKINGS ACROSS THE THREE INPUT CLIMATIC SCENARIOS (S1–S3).

Scenario	**Friedman χ^2**	**Value_*p***	**Average Model** Rank					
			RF	**XGBoost**	**TabNet**	**PSO-TabNet**	**KAN**	**PSO-KAN**
S1 (Temp-Only)	40.57	1.15×10^{-7}	2.60	1.10	2.70	4.20	5.80	4.60
S2 (Thermal-Path)	44.11	2.20×10^{-8}	2.20	1.00	3.70	3.50	6.00	4.60
S3 (Full-Features)	34.23	2.14×10^{-6}	6.00	1.90	3.20	1.90	3.90	4.10

Note: χ^2 = Friedman chi-square statistic; $p < 0.05$ indicates a statistically significant difference in performance among models within a scenario. Average ranks (AvgRank) are computed per model per scenario, where a lower value indicates better relative performance (rank 1 = best, rank 6 = worst).

One result stands out above all others: Random Forest, competitive in the reduced-input scenarios, falls to the last rank (last in all ten seeds) once the full feature set is introduced — despite reporting one of the highest absolute R^2 values in the entire study. The Friedman statistics indicate that this is not seed-to-seed noise. The null hypothesis of equal performance was rejected at the standard significance level ($\alpha = 0.05$) across all three scenarios (S1: $\chi^2 = 40.57$, $p = 1.15 \times 10^{-7}$; S2: $\chi^2 = 44.11$, $p = 2.20 \times 10^{-8}$; S3: $\chi^2 = 34.23$, $p = 2.14 \times 10^{-6}$), indicating rank differences among the models within this seed set. XGBoost obtained the best or tied-best average rank throughout (1.1, 1.0, and 1.9 for S1, S2, and S3, respectively). Random Forest's average rank moved from 2.2–2.6 under S1/S2 to 6.0 under S3—the largest rank reversal among the six models, and a striking one given its apparently strong R^2 of 0.986 in Table 1. This is because when the five climate variables are combined, the absolute difference between model performances falls, leading to a range of performances from 0.986 to 0.996. This means that the consistent underperformance (by a small margin) for one seed set will affect the ranking of the

models. This would not have been picked up by looking at the accuracy measures in isolation. Within each scenario the Friedman test was performed which is an overall test across the 6 models. The Nemenyi method was used to adjust the family-wise error rate for the 15 pairwise comparisons (α = 0.05, CD = 2.384). There was no pooling of inference between scenarios; however, the three Friedman p-values were all less than 0.05/3, meaning that the overall conclusion still holds when correcting using the Bonferroni method.

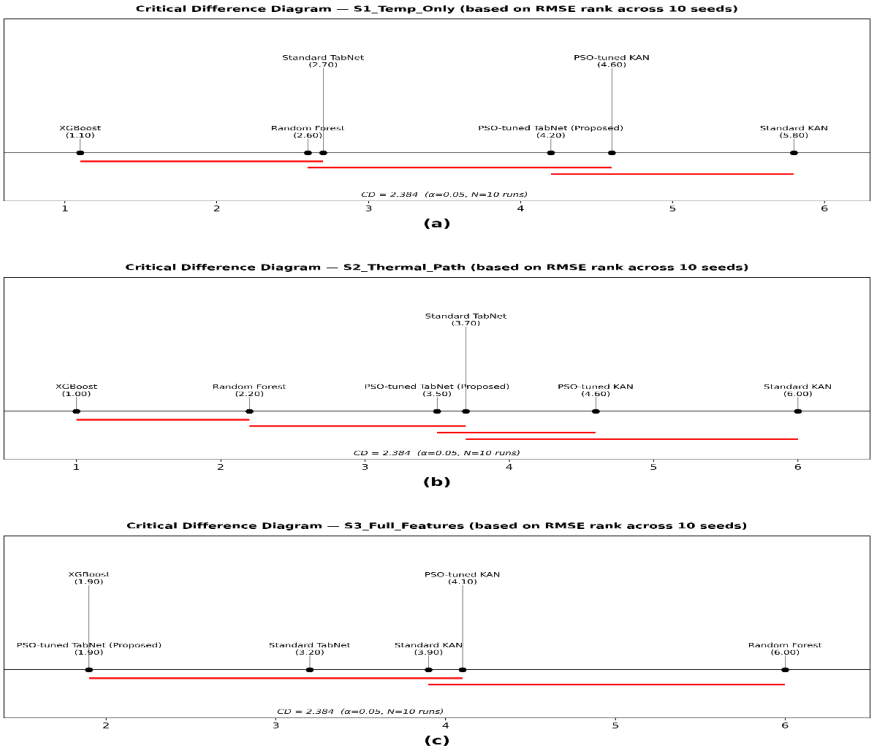


Fig 7. Critical-difference diagrams for (a) S1, (b) S2, and (c) S3. Models connected by a horizontal bar do not differ significantly (CD = 2.384, α = 0.05, N = 10).

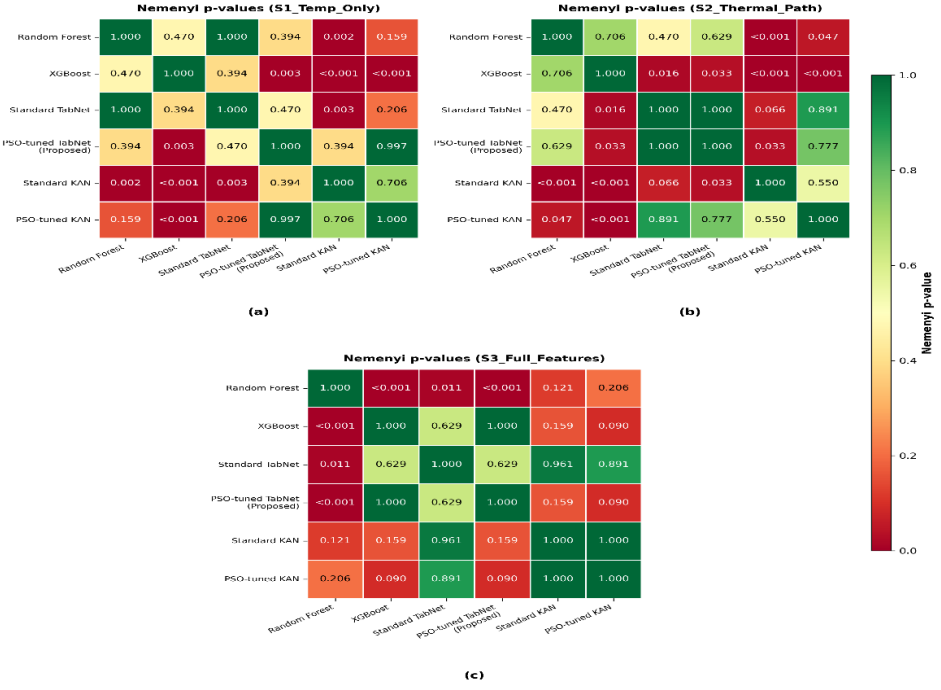


Fig 8. Nemenyi post-hoc pairwise significance heatmaps for the three experimental scenarios: (a) S1_Temp_Only, (b) S2_Thermal_Path, and (c) S3_Full_Features.

Critical difference charts (Figure 7 (CD = 2.384, α = 0.05), calculated over N = 10 replicated seeds) show the significance of the rank differences. In S3, the proposed PSO tuned TabNet is tied with XGBoost via the critical difference bars meaning there was no difference found between the models once the entire set of climatic features are available. However, in S1 the proposed model is tied with the Random Forest, Standard TabNet and both KAN models but separated from the top ranked XGBoost model. Taken together, the two panels suggest that, in this experimental setup, the position of PSO-tuned TabNet relative to XGBoost depends on the input scenario. The corresponding Nemenyi pairwise p-value matrices (Figure 8) are consistent with this pattern at the individual-pair level: the PSO-tuned TabNet–XGBoost comparison remains non-significant in S3 ($p = 1.000$) while its S1 comparison with XGBoost falls below the 0.05 threshold ($p = 0.003$). In S1, its comparison with Random Forest remains non-significant ($p = 0.394$); these are the exact significance values underlying the grouping shown in Figure 7. A best observed mean rank is not the same as significant superiority: in S1, XGBoost's best rank is significantly separated from PSO-tuned TabNet ($p = 0.003$) and from both KAN variants ($p < 0.001$), but not from Random Forest ($p = 0.470$) or Standard TabNet ($p = 0.394$). Comparisons are made assuming the ten seed set. Comparisons using the five expanding window folds (Table 5) show that the XGBoost model (mean = $R^2 = 0.972$) performs better than the PSO tuned TabNet (0.907), primarily due to its performance on fold 1. This is complementary evidence of the temporal nature of the models but should be considered a post-hoc diagnostic (Section 3.4).

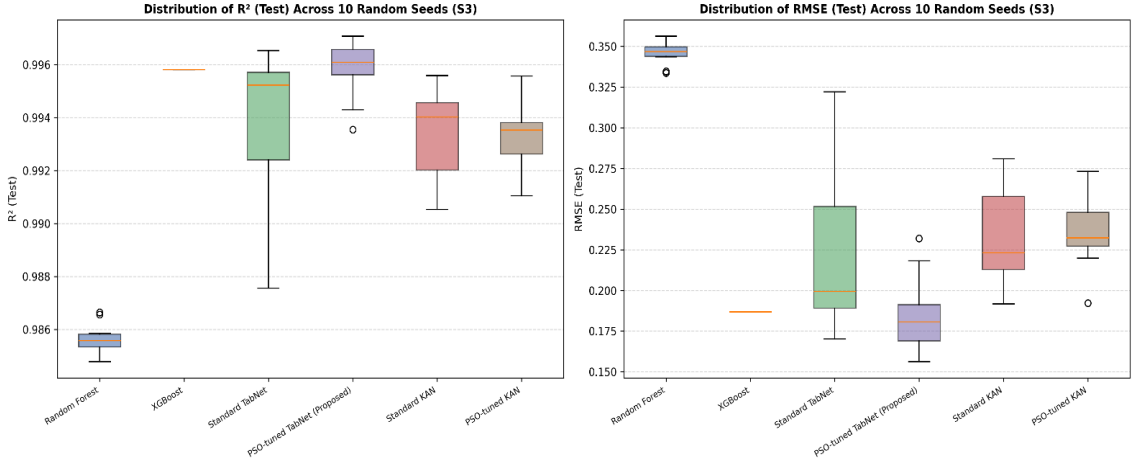


Fig 9. Distribution of test R^2 and RMSE across ten random seeds (S3_Full_Features).

A statistically competitive mean rank is not the same as a stable model, and Figure 9 makes this distinction relevant to practical deployment. PSO-tuned TabNet's run-to-run RMSE standard deviation across the ten seeds was 0.0223 mm day^{-1} (Table 1). Its average Friedman rank in S3 (1.9) was numerically identical to that of XGBoost, and the corresponding Nemenyi pairwise comparison was non-significant ($p = 1.000$; Figure 8), indicating no detected difference in central tendency between the two models; this should not be read as equivalent run-to-run stability, since the rank test does not evaluate variance. XGBoost's zero observed standard deviation across all metrics reflects the deterministic nature of its fixed-split tree construction rather than noise-free superiority. Nonetheless, given that ten seeds remain a finite sample of the initialization space, we recommend reporting ensemble-averaged predictions from multiple independently seeded runs before the model is used for operational, day-to-day irrigation scheduling, consistent with general best practice for stochastic deep-learning models.

3.3 Prediction Diagnostics and Overfitting Behavior

To determine whether the high test-set accuracy reported in Section 3.1 reflects genuine generalization rather than memorization of the training data, the train–test R^2 gap was computed for each model under the full-feature scenario (S3).

TABLE III. TRAIN–TEST R^2 GAP ANALYSIS FOR OVERFITTING ASSESSMENT (FULL-FEATURE SCENARIO).

Model	R2_Train	R2_Test	Gap
Random Forest	0.9981	0.9857	0.0124
XGBoost	0.9988	0.9958	0.0030
Standard TabNet	0.9963	0.9938	0.0025
PSO-tuned TabNet	0.9977	0.9958	0.0019
Standard KAN	0.9956	0.9934	0.0022
PSO-tuned KAN	0.9974	0.9933	0.0041

Random Forest exhibited the largest generalization gap (ΔR^2 = 0.0124), roughly 3 to 6.5 times larger than the other five models, despite its competitive absolute test R^2 (0.986, Table 3). This finding is consistent with its poor average rank in the Friedman test (Section 3.2), suggesting, though not demonstrating, that its accuracy is comparatively more sample-dependent. In contrast, Standard TabNet (0.0025), Standard KAN (0.0022), and XGBoost (0.0030) showed small gaps, whereas PSO-tuned KAN had the second-largest (0.0041). At the same time, the proposed PSO-tuned TabNet exhibited the smallest generalization gap among all six models (0.0019), slightly below its non-tuned counterpart (0.0025), a difference smaller than the seed-to-seed spread of the untuned model's test R^2 (± 0.0028), so it should not be read as a benefit of tuning. However, a small train–test gap alone does not rule out memorization. The circular sin/cos encoding of wind direction was present in all S3 models, including this one. Because wind direction is not an FAO-56 input and the ablation of Section 3.5 (Table 8) showed no accuracy benefit from it, no contribution of this variable to accuracy or to the generalization gap is demonstrated.

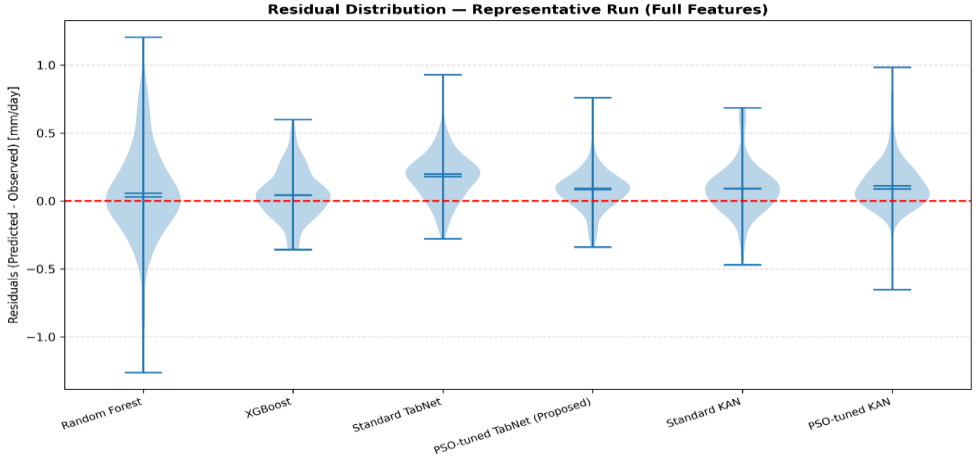


Fig 10. Residual distribution across models (full-feature scenario; representative run, seed = 505).

The residual distributions (Figure 10; representative run, seed = 505) are consistent with this pattern for Random Forest, which displays the widest error tails (residuals extending to approximately −1.3 mm day^{-1}) in line with its larger train–test gap, followed by PSO-tuned KAN (approximately −0.7 mm day^{-1}); XGBoost, PSO-tuned TabNet and Standard KAN show more concentrated distributions (lower tails near −0.3 to −0.5 mm day^{-1}); Standard KAN, whose train–test gap is the second smallest in Table 3, also shows narrow tails, and no general relation between tail width and gap is claimed. All six models remained centered close to the zero-bias line, but every model showed a small positive median residual (about +0.03 to +0.2 mm day^{-1}, largest for Standard TabNet), indicating a slight tendency toward over-prediction that warrants monitoring in operational deployment.

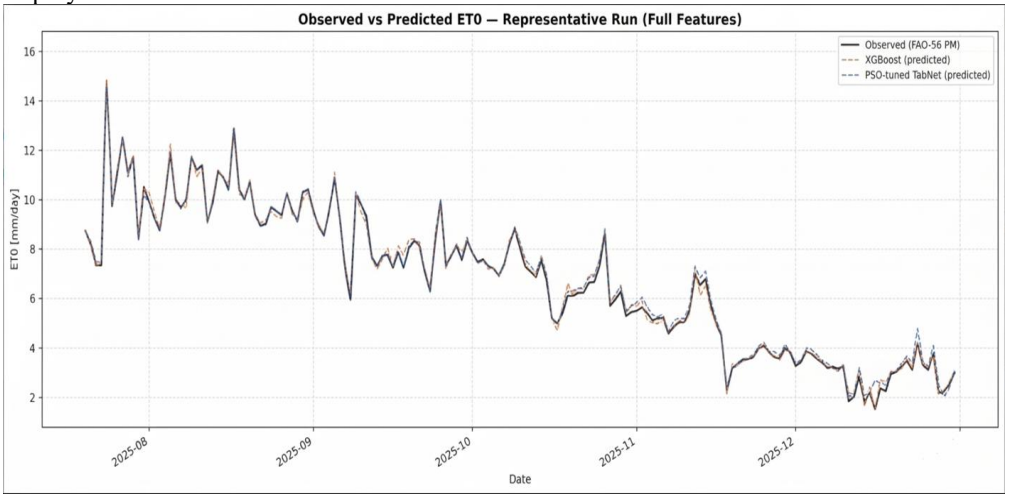


Fig 11. Observed versus predicted ET0 time series for the two top-performing models over the held-out test period.

The time series plots in Figure 11 show that the two best performing models have a consistent performance across time. The XGBoost and PSO tuned TabNet models successfully captured the high peak (around 15mm day^{-1}) and low demand

(below 4mm day^{-1}) fluctuations in the observed ET0 data in the final time series chunk. This chunk was held out and not used during training. There is no sign of seasonality or lag affecting the models. This behavior is particularly relevant for irrigation scheduling, where underestimation during peak demand periods carries a higher practical cost than comparable errors during low-demand periods. While Figure 11 illustrates this behavior for a single representative test window, robustness across multiple, seasonally distinct test periods is explored in a retrospective expanding-window diagnostic reported in Section 3.4.

3.4 Temporal Behavior (Retrospective Diagnostic) and Computational Cost

While the single chronological 70/15/15 split used in Sections 3.1–3.3 provides a consistent benchmark, it evaluates the models on only one test period and cannot establish robustness across different seasonal windows. Following established recommendations against standard random cross-validation for temporally dependent data [45], [46], an expanding-window Timeseries Split procedure was applied to the training and validation data (S3), yielding five successive folds of increasing training-window length (481–841 days) and fixed 90-day test windows, spanning the training-plus-validation portion of the record (January 2023–July 2025). As the hyperparameters were chosen in advance using overlapping segments of the data (see below), this experiment represents a configuration diagnostic rather than an independent temporal validation.
The overall results for the expanding window test procedure, along with the test accuracy, generalization gap and rankings are summarized in Table 4.

TABLE IV. COMPREHENSIVE SYNTHESIS OF PREDICTIVE ACCURACY, GENERALIZATION BEHAVIOR, STATISTICAL RANKING, AND TEMPORAL BEHAVIOR (RETROSPECTIVE DIAGNOSTIC) ACROSS THE SIX EVALUATED MODELS UNDER THE FULL-FEATURE CLIMATIC SCENARIO (S3).

Metric	RF	XGBoost	Std. TabNet*	PSO-TabNet*^	Std. KAN*	PSO-KAN
Test R^2	0.986	0.996	0.994	0.996	0.993	0.993
Test RMSE (mm/day)	0.346	0.187	0.222	0.186	0.233	0.235
Generalization gap[#]	0.0124	0.0030	0.0025	0.0019	0.0022	0.0041
Avg. Friedman rank[*#]	6.0	1.9	3.2	1.9	3.9	4.1
5-fold expanding-window R^2 (retrospective)[*##]	0.949	0.972	0.921	0.907	−1.820	0.842

Std.= Standard (non-optimized) architecture. ^ Proposed model, combining TabNet with Particle Swarm Optimization for hyperparameter tuning. [#] Generalization gap = R^2Train − R^2Test (Table 3). [#] Mean rank from the Friedman test applied to RMSE across ten random seeds; lower rank = better performance (Section 3.2). [*##] Mean test R^2 across five expanding-window folds (Section 3.4); see Table 5 for fold-by-fold results. Single run per fold (retrospective diagnostic). Three-decimal values are rounded from Tables 1, 3 and 5.

TABLE V. EXPANDING-WINDOW FOLDS (S3; RETROSPECTIVE DIAGNOSTIC, SINGLE RUN, SEED 42).

Fold	Training Window (days)	R^2 RF	R^2 (XGBoost)	R^2 Std TabNet	R^2 (PSO-TabNet)	R^2 Std KAN	R^2 PSO-KAN
1	481	0.924	0.952	0.948	0.663	-2.734	0.425
2	571	0.953	0.987	0.971	0.984	0.917	0.984
3	661	0.939	0.955	0.919	0.932	0.753	0.867
4	751	0.960	0.981	0.991	0.989	-0.472	0.960
5	841	0.968	0.985	0.778	0.968	-7.562	0.975

The hyperparameters of the TabNet model optimized by PSO (n_d, n_steps, γ, learning rate) were chosen once using PSO on the whole training/validation split and kept constant across the 5 folds. Since the test window for each fold is part of this training/validation split, we are not avoiding data leakage, and this analysis can be seen as a post-hoc analysis on the performance of different configurations. This was also applied to the other models optimized by hyperparameter tuning. The KAN model optimized by PSO uses hyperparameters optimized by PSO on the same training/validation split. The Random Forest and XGBoost models use the hyperparameters from the grid search on training segment. For each fold, each model was only trained once. Within each fold, the feature scaler was fitted on that fold's training data only. TabNet models were trained for a fixed 60 epochs without early stopping or checkpoint selection (final-epoch weights); KAN models were fitted on the first 85% of the training window, with early stopping and checkpoint selection on its last 15% (at least 30 days), for at most 60 epochs (patience 10); Random Forest and XGBoost were fitted on the full training window. The evaluation window was never used for fitting, early stopping or checkpoint selection, and all folds used seed 42. These fold-level training budgets are shorter than in the ten-seed experiments (Section 2.6), so the fold-level results are not directly comparable with Table 1. XGBoost maintained a higher mean R^2 across the five folds (0.972 ± 0.015) than the proposed PSO-tuned TabNet (0.907 ± 0.124, Table 4), a gap driven primarily by Fold 1 (Table 5), where, with the shortest training window (481 days), PSO-tuned TabNet's R^2 dropped to 0.663 against XGBoost's 0.952 (Standard TabNet, trained under the same protocol, reached 0.948 in this fold; with one run per fold, configuration effects cannot be separated from initialization variability). Tree-based ensembles are known to remain highly competitive under limited training data [47]. Once the training window is extended beyond 570 days (Folds 2-5 in Table 5), the difference is much smaller. PSO-Tuned

TabNet performs on par or similar to XGBoost for 3 out of the 4 folds. However, there is not a linear improvement. Fold 3 (661 days) still shows some moderate decline ($R^2 = 0.932$) in comparison to the surrounding folds. This is simply what we have seen happen at 570 days. A markedly different pattern emerged for the KAN architecture: Standard KAN was highly unstable across folds ($R^2 = -1.820 \pm 3.154$), with three of five folds — including Fold 1 ($R^2 = -2.734$) and Fold 5 ($R^2 = -7.562$) — yielding negative R^2 values, whereas PSO-tuned KAN remained positive across all folds ($R^2 = 0.842 \pm 0.213$), though with a lower minimum (0.425 in Fold 1) than its mean would suggest, indicating meaningful but incomplete stabilization relative to Standard KAN's outright collapse. Unlike TabNet, there was no obvious trend in KAN's instability. The best performing option for these folds was the PSO option, however this does not guarantee the need for PSO. This is due to only one run being conducted per fold. Furthermore, the initialisation, configuration and optimisation were not separated. Finally, the PSO options were also chosen based on data overlapping the fold windows whereas the defaults weren't. A possible reason (that has not been tested) is that the KAN with RBFs is very sensitive to the grid resolution (which is always 8 points). Therefore when the training window size is small or not representative, the difference between the distribution of features in the training fold and the effective range of the RBF basis functions causes the extrapolation error to accumulate quickly causing large negative R^2 values such as those seen in Folds 1 and 5. However in S3 the PSO selected KAN had the same grid size (8) as the default options but the hidden width and learning rate were different (19 vs 16 and 0.085 vs 0.01). Thus, the grid resolution cannot be the cause of the difference.

TABLE VI. MEAN TRAINING TIME PER MODEL (S3, AVERAGED OVER TEN SEEDS).

Model	Training Time (s)
XGBoost	0.1
Random Forest	0.3
Standard KAN	0.5
PSO-tuned KAN	0.7
Standard TabNet	4.6
PSO-tuned TabNet (Proposed)	5.9

All experiments were implemented in Python using PyTorch, scikit-learn, and XGBoost. Under the full-feature scenario, mean training time per model (10-seed average) ranged from 0.1–0.3 s for Random Forest and XGBoost to 4.6–5.9 s for Standard and PSO-tuned TabNet, and 0.5–0.7 s for Standard and PSO-tuned KAN (Table 6). Reported training times exclude the PSO search cost of 400 fitness evaluations per scenario for TabNet and KAN. Per scenario and architecture, PSO required 1,200 truncated trainings, whereas the grid searches required 27 fits for Random Forest and 54 for XGBoost, i.e., roughly 22 to 44 times more model fits (counts of fits, not wall-clock times). The complete pipeline — PSO search (20 particles × 20 iterations) across three input scenarios for both TabNet and KAN, training of all six configurations over ten random seeds, the expanding-window diagnostic, and figure generation— required approximately 70 minutes total wall-clock time. The hyperparameters selected for each model and input scenario are listed in Table 7.

TABLE VII. SELECTED HYPERPARAMETERS FOR EACH MODEL AND INPUT SCENARIO.

Model	Hyperparameter	S1	S2	S3
PSO-tuned TabNet	n_d = n_a	27	8#	9
	n_steps	6#	3#	5
	γ	1.552	1.535	1.089
	learning rate	0.0316#	0.0249	0.0316#
PSO-tuned KAN	hidden width	20	19	19
	RBF grid points	13	16#	8
	learning rate	0.100#	0.100#	0.0854
Random Forest	max_depth	8#	16#	12
	n_estimators	100#	100#	150
XGBoost	learning_rate	0.03#	0.03#	0.1#
	max_depth	3#	3#	3#
	n_estimators	200#	200#	200#

#Values at or within 0.3% of the search-range or grid boundaries are flagged. Fixed configurations: Standard TabNet n_d = n_a = 16, n_steps = 5, γ = 1.5, lr = 0.02; Standard KAN hidden width 16, 8 RBF grid points, lr = 0.01.

3.5 Feature Contribution (Ablation)

To isolate the contribution of the added variables, a factorial feature ablation was performed with XGBoost, Random Forest and PSO-tuned TabNet (Table 8). Relative humidity, wind speed and wind direction were added to S2 (temperature + radiation) in all combinations, and radiation and temperature were removed in turn from S3, with hyperparameters fixed at the S3 values. Wind speed was the dominant added variable: alone it reduced RMSE by 55–63%, and a Shapley decomposition attributed 90.4–98.0% of the total reduction to it. Relative humidity helped mainly in the presence of wind speed (a further 18.5–50.4% reduction), consistent with the product of wind speed and vapor-pressure deficit in the FAO-

56 aerodynamic term, although this was not tested separately. Wind direction, which is not an FAO-56 input, gave no benefit (RMSE +4% to +14% when added to the other two). Removing radiation or temperature raised RMSE by 54–162% and 137–324%, so both remain essential.

TABLE VIII. FEATURE ABLATION (TEST RMSE, mm day⁻¹) AND SHAPLEY ATTRIBUTION OF THE RMSE REDUCTION RELATIVE TO S2.

(a) Test RMSE by input set (Δ relative to S2).

Input set	XGBoost RMSE (Δ vs S2)	Random Forest RMSE (Δ vs S2)	PSO-tuned TabNet RMSE (Δ vs S2)
S2 (T + Rg)	0.833	0.835	0.920
S2 + Hr	0.961 (+15.3%)	0.844 (+1.0%)	0.867 (-5.8%)
S2 + Vv	0.331 (-60.3%)	0.378 (-54.8%)	0.342 (-62.8%)
S2 + Dv	0.850 (+2.0%)	0.845 (+1.1%)	0.946 (+2.8%)
S2 + Hr + Vv	0.179 (-78.5%)	0.308 (-63.1%)	0.170 (-81.6%)
S2 + Hr + Dv	0.936 (+12.4%)	0.825 (-1.2%)	0.952 (+3.4%)
S2 + Vv + Dv	0.335 (-59.8%)	0.404 (-51.7%)	0.373 (-59.5%)
S3 (all inputs)	0.187 (-77.6%)	0.346 (-58.6%)	0.193 (-79.0%)
S3 without Rg	0.474 (-43.1%)	0.531 (-36.4%)	0.506 (-45.1%)
S3 without T	0.775 (-7.0%)	0.818 (-2.1%)	0.821 (-10.8%)

(b) Shapley attribution of the RMSE reduction S2 → S3 over the three added groups (RMSE reduction in mm day⁻¹; share of the total).

Variable group	XGBoost (mm/day; share)	Random Forest (mm/day; share)	PSO-tuned TabNet (mm/day; share)
Wind speed (Vv)	+0.633; +98.0%	+0.475; +97.1%	+0.657; +90.4%
Relative humidity (Hr)	+0.018; +2.7%	+0.031; +6.4%	+0.105; +14.5%
Wind direction (Dv, sin/cos)	-0.005; -0.7%	-0.017; -3.5%	-0.036; -4.9%

Note: Hr = relative humidity, Vv = wind speed, Dv = wind direction (sin and cos components treated as one group), Rg = global radiation, T = air temperature. S2 = T + Rg; S3 = all inputs. All sets used the hyperparameters selected for S3 (Table 7), not re-tuned per subset. Random Forest and PSO-tuned TabNet: mean over ten seeds; XGBoost: single deterministic run. PSO-tuned TabNet was re-trained with the rounded Table 7 values (S3 RMSE 0.193 ± 0.043 versus 0.186 ± 0.022 in Table 1); the S2 reference values differ slightly from Table 1 because the S3 hyperparameters were used for all input sets (e.g., XGBoost 0.833 versus 0.823). Negative Δ = lower RMSE than S2. The origin of this difference was not isolated; TabNet differences of about 0.01–0.02 mm day⁻¹ in Table 8 are therefore not interpreted.

3.6 Limitations

This study has the following limitations. The conclusions apply to a single hyper-arid station, and should not be extrapolated to other climates or stations. The ten seeds are replicate runs, so the Friedman–Nemenyi results are descriptive and conditional on the seed set, and the expanding-window analysis is a retrospective configuration diagnostic because its hyperparameters were selected on overlapping segments; nested forward-chaining validation with re-tuning inside each fold was not performed. The rank differences between the PSO-tuned and the default models are below the critical difference, PSO-tuned TabNet also had a longer epoch budget (150 versus 100 epochs), and PSO was not compared with other optimizers under an equal budget, so any benefit is attributed to the tuned configuration and not specifically to PSO, which required roughly 22 to 44 times more model fits than the grid searches (Section 3.4).

4. CONCLUSION

This study evaluated six machine learning architectures for daily reference evapotranspiration estimation under progressively enriched climatic input scenarios, combining accuracy assessment, non-parametric statistical ranking, overfitting diagnostics, and a retrospective temporal diagnostic into a single evaluation framework. Three conclusions can be drawn. Firstly, when there is severe predictor sparsity, it was observed that the different model architectures had the greatest differences in performance in the case of only temperature variables. The differences between model architectures were reduced when using the complete set of features. This suggests that selecting the algorithm is more important when there are fewer surrogate predictors to choose between. Secondly, in the case of the full climatic scenario, the proposed PSO tuned TabNet architecture achieved the exact same mean rank as the XGBoost model. There was no difference in the central tendencies ($p = 1.000$). However, this was not consistent throughout all input scenarios. Additionally, the benefits gained from tuning with metaheuristics varied depending on the architecture being used. This is highlighted by looking at the results from the Kolmogorov-Arnold Network. In the expanding window diagnostic scenario, the Standard KAN achieved very poor negative results ($R^2 = -7.562$ in some folds). Whereas the PSO tuned KAN maintained positive values throughout ($R^2 = 0.842 \pm 0.213$). In these cases, it can be seen that the PSO tuned set up performed better than the specific default set up used. However, this does not necessarily mean that PSO is required. Tuning was only conditionally beneficial for TabNet, in contrast with XGBoost, whose deterministic, near-zero-variance behavior required only a lightweight grid search to reach top-ranked performance. These rank-based conclusions are conditional on the ten-seed set, one station and one test window. Third, the largest accuracy gains were observed when wind speed, wind direction, and relative humidity were added to temperature and radiation. A feature ablation of three models attributed most of this gain to wind speed, with relative humidity contributing mainly in combination with it and wind direction giving no benefit; these are marginal

sensitivities of an FAO-56 surrogate under fixed hyperparameters at one station, and should not be read as a validated instrumentation recommendation. Practically, XGBoost remains the safer default for short-record stations, while PSO-tuned TabNet merits consideration once multi-year records are available, as it approached XGBoost in most later folds of the retrospective diagnostic. Future work should extend this evaluation to additional stations, climatic zones and longer records, and should employ nested forward-chaining validation with hyperparameter re-tuning inside each fold.

Acknowledgment
None.

Funding
No funding was received.

CRediT authorship contribution statement

Sarra Gourari: conceptualization, methodology, software, formal analysis, investigation, visualization, Writing – original draft, Writing – review & editing.
Wafa Difallah: Conceptualization, Supervision, methodology, Validation, Writing – review & editing.
Belkacem Draoui: Supervision, methodology, Validation.

Conflicts of Interest
The authors declare no conflicts of interest.

Ethics Approval and Informed Consent
Not applicable. This study used only meteorological data and did not involve human participants, animals, or personal data.

Data availability
The data that support the findings of this study are available from the corresponding author upon reasonable request.

References

[1] La Situation mondiale de l'alimentation et de l'agriculture *2024*. FAO, 2024. doi: 10.4060/cd2616fr.
[2] J. Doorenbos and W. O. Pruitt, "Guidelines for Predicting Crop Water Requirements," Food and Agriculture Organization of the United Nations, Rome, 1977.
[3] R. G. Allen, L. S. Pereira, D. Raes, and M. Smith, "Crop evapotranspiration: Guidelines for computing crop water requirements," Food and Agriculture Organization of the United Nations (FAO), Rome, Italy, 56, 1998. [Online]. Available: https://www.fao.org/4/X0490E/X0490E00.htm
[4] W. Difallah, K. Benahmed, B. Draoui, and F. Bounaama, "Linear Optimization Model for Efficient Use of Irrigation Water," *International Journal of Agronomy*, vol. 2017, pp. 1–8, 2017, doi: 10.1155/2017/5353648.
[5] N. Malamos, A. Tegos, G. Bourantas, C. Chalvantzis, and D. Koutsoyiannis, "Global reference evapotranspiration clustering and its relation to the Köppen-Geiger climate classification," *Journal of Hydrology*, vol. 660, p. 133342, Oct. 2025, doi: 10.1016/j.jhydrol.2025.133342.
[6] Y. Zhang, T. He, S. Liang, Y. Ma, and Y. Yao, "A novel approach for estimating evapotranspiration by considering topographic effects in radiation over mountainous terrain," *Agricultural and Forest Meteorology*, vol. 366, p. 110468, May 2025, doi: 10.1016/j.agrformet.2025.110468.
[7] Y. Chang, C. Zhang, J. Huang, H. Chang, C. Wang, and Z. Huo, "Machine Learning for Reference Crop Evapotranspiration Modeling: A State-of-the-Art Review and Future Directions," *Agronomy*, vol. 15, no. 9, p. 2038, Aug. 2025, doi: 10.3390/agronomy15092038.
[8] G. Garbanzo *et al.*, "Addressing Weather Data Gaps in Reference Crop Evapotranspiration Estimation: A Case Study in Guinea-Bissau, West Africa," *Hydrology*, vol. 12, no. 7, p. 161, Jun. 2025, doi: 10.3390/hydrology12070161.
[9] A. Ayaz, M. Rajesh, S. K. Singh, and S. Rehana, "Estimation of reference evapotranspiration using machine learning models with limited data," *AIMS Geosciences*, vol. 7, no. 3, pp. 268–290, 2021, doi: 10.3934/geosci.2021016.
[10] S. Acharki *et al.*, "Comparative assessment of empirical and hybrid machine learning models for estimating daily reference evapotranspiration in sub-humid and semi-arid climates," *Sci Rep*, vol. 15, no. 1, p. 2542, Jan. 2025, doi: 10.1038/s41598-024-83859-6.
[11] F. O. Adjei, K. A. Adjei, and E. Obuobie, "Calibration of the Hargreaves and Priestley-Taylor equations for estimating reference evapotranspiration in the Densu River Basin in Ghana," *African Journal of Environmental Science and Technology*, vol. 19, no. 3, pp. 131–142, 2025.

[12] J. Zhang, L. Wang, G. Cheng, and L. Jia, "Applicability of Evapotranspiration Models and Water Consumption Characteristics Across Different Croplands," *Agronomy*, vol. 15, no. 6, p. 1441, Jun. 2025, doi: 10.3390/agronomy15061441.
[13] L. S. D. P. Figueiró, C. S. Bonfá, and L. D. C. Santos, "Calibration and evaluation of the Hargreaves-Samani equation for estimating reference evapotranspiration: a case study in Northern Minas Gerais, Brazil," *R. G. Secr.*, vol. 16, no. 2, p. e4638, Feb. 2025, doi: 10.7769/gesec.v16i2.4638.
[14] M. S. Bhosale, S. A. Kadam, N. J. Danwale, M. S. Mane, V. S. Malunjkar, and J. K. Joshi, "Estimation of Actual Evapotranspiration and Crop Coefficient to Improve Water Use Efficiency of Sesamum (Sesamum indicum L.) in Kharif Season Using Digital Weighing Lysimeter," *Asian J. Env. Ecol.*, vol. 24, no. 1, pp. 67–78, Jan. 2025, doi: 10.9734/ajee/2025/v24i1651.
[15] W. Difallah, B. Ouradj, F. Bounaama, B. Draoui, K. Benahmed, and K. Lammari, "High-Accuracy Estimation of Reference Evapotranspiration Using Classical and AI-Supported System Identification Approaches Under Different Climatic Conditions in Arid Zones," *Electronics*, vol. 15, no. 4, p. 799, Feb. 2026, doi: 10.3390/electronics15040799.
[16] P. V. Lykhovyd, R. A. Vozhehova, and O. O. Piliarska, "Comparison of smart applications for evapotranspiration assessment automation with manual estimation using the Thornthwaite method," *AL*, vol. 8, no. 2, pp. 89–95, Jul. 2025, doi: 10.32819/202512.
[17] P. E. Ratshiedana, M. A. M. Abd Elbasit, E. Adam, and J. G. Chirima, "Evaluation of Micrometeorological Models for Estimating Crop Evapotranspiration Using a Smart Field Weighing Lysimeter," *Water*, vol. 17, no. 2, p. 187, Jan. 2025, doi: 10.3390/w17020187.
[18] A. A. Alazba, M. A. Mattar, A. El-Shafei, F. Radwan, M. Ezzeldin, and N. Alrdyan, "Daily Reference Evapotranspiration Derived from Hourly Timestep Using Different Forms of Penman–Monteith Model in Arid Climates," *Water*, vol. 17, no. 15, p. 2272, Jul. 2025, doi: 10.3390/w17152272.
[19] Y. Boukhali, M. N. Kabbaj, and M. Benbrahim, "Evapotranspiration estimation using deep learning models for robust sensor performance," *Franklin Open*, vol. 12, p. 100335, Sep. 2025, doi: 10.1016/j.fraope.2025.100335.
[20] J. C. Paul, P. K. Paramaguru, and S. Swain, "Application of innovative data-driven techniques for modeling reference evapotranspiration," *Journal of Soil and Water Conservation, India*, vol. 24, no. 1, pp. 47–53, 2025, doi: 10.5958/2455-7145.2025.00005.3.
[21] M. A. M. Nasir, S. Harun, Z. M. Zainuddin, M. R. Kamal, and F. Z. C. Rose, "Machine learning techniques for reference evapotranspiration and rice irrigation requirements prediction: a case study of Kerian irrigation scheme, Malaysia," *Paddy Water Environ*, vol. 23, no. 4, pp. 671–683, Oct. 2025, doi: 10.1007/s10333-025-01040-9.
[22] E. R. Coutinho, J. G. F. Madeira, R. M. D. Silva, A. R. S. Delgado, and A. L. G. A. Coutinho, "Genetic Algorithms Applied to Optimize Neural Network Training in Reference Evapotranspiration Estimation," *Rev. bras. meteorol.*, vol. 40, p. e40240009, 2025, doi: 10.1590/0102-778640240009.
[23] Z. Li, Z. Zhao, L. Xing, L. Zhao, N. Cui, and H. Cai, "Estimating Daily Reference Crop Evapotranspiration in Northeast China Using Optimized Empirical Models Based on Heuristic Intelligence Algorithms," *Agronomy*, vol. 15, no. 3, p. 599, Feb. 2025, doi: 10.3390/agronomy15030599.
[24] M. Tausif, M. W. Iqbal, R. N. Bashir, B. AlGhofaily, A. Elyassih, and A. R. Khan, "Federated learning based reference evapotranspiration estimation for distributed crop fields," *PLoS ONE*, vol. 20, no. 2, p. e0314921, Feb. 2025, doi: 10.1371/journal.pone.0314921.
[25] A. Nasrolahi, Y. Sabzevari, and S. Eslamian, "Estimation of reference evapotranspiration using experimental models and approaches based on neural network (case study: Khorramabad City, Iran)," *Water Practice & Technology*, vol. 20, no. 6, pp. 1437–1449, Jun. 2025, doi: 10.2166/wpt.2025.079.
[26] M.-A. Lahkim-Bennani, B. Bonaccorso, A. A. Tribak, and A. Rossi, "Reference evapotranspiration forecasting in the North Etna aquifer: a comparative analysis of statistical, deep learning, and machine learning models," *Theor Appl Climatol*, vol. 156, no. 9, p. 492, Sep. 2025, doi: 10.1007/s00704-025-05726-2.
[27] A. Haris, M. Marimin, S. Wahjuni, and B. I. Setiawan, "H-ConvLSTM to Estimate Reference Evapotranspiration From Air Temperature and Relative Humidity," *IEEE Access*, vol. 13, pp. 128789–128801, 2025, doi: 10.1109/ACCESS.2025.3585771.
[28] P. Heramb, K. V. Ramana Rao, A. Subeesh, and A. Srivastava, "Predictive Modelling of Reference Evapotranspiration Using Machine Learning Models Coupled with Grey Wolf Optimizer," *Water*, vol. 15, no. 5, p. 856, Feb. 2023, doi: 10.3390/w15050856.
[29] A. Belkendil *et al.*, "Integrated GIS-based approach for groundwater potential mapping in arid regions: A case study of Kenadsa-Bechar using AHP and frequency ratio methods," *Groundwater for Sustainable Development*, vol. 30, p. 101461, Aug. 2025, doi: 10.1016/j.gsd.2025.101461.

[30] S. Mebarki, M. A. Kendouci, and A. Bendida, “Monitoring the spatial evolution of groundwater quality during its diversion in the drinking water supply network in arid areas, case of Bechar city (Algeria Sahara),” *Appl Water Sci*, vol. 14, no. 6, p. 118, Jun. 2024, doi: 10.1007/s13201-024-02157-8.
[31] P. Henderson, R. Islam, P. Bachman, J. Pineau, D. Precup, and D. Meger, “Deep Reinforcement Learning That Matters,” *AAAI*, vol. 32, no. 1, Apr. 2018, doi: 10.1609/aaai.v32i1.11694.
[32] J. Žižka, F. Dařena, and A. Svoboda, “Random Forest,” in *Text Mining with Machine Learning*, 1st ed., Boca Raton: CRC Press, 2019, pp. 193–200. doi: 10.1201/9780429469275-8.
[33] Z. Ma, J. Guo, S. Mao, and T. Gu, “An interpretability research of the Xgboost algorithm in remaining useful life prediction,” in *2020 International Conference on Big Data & Artificial Intelligence & Software Engineering (ICBASE)*, Bangkok, Thailand: IEEE, Oct. 2020, pp. 433–438. doi: 10.1109/ICBASE51474.2020.00098.
[34] S. Ö. Arik and T. Pfister, “TabNet: Attentive Interpretable Tabular Learning,” *AAAI*, vol. 35, no. 8, pp. 6679–6687, May 2021, doi: 10.1609/aaai.v35i8.16826.
[35] C. Shah, Q. Du, and Y. Xu, “Enhanced TabNet: Attentive Interpretable Tabular Learning for Hyperspectral Image Classification,” *Remote Sensing*, vol. 14, no. 3, p. 716, Feb. 2022, doi: 10.3390/rs14030716.
[36] L. P. Joseph, E. A. Joseph, and R. Prasad, “Explainable diabetes classification using hybrid Bayesian-optimized TabNet architecture,” *Computers in Biology and Medicine*, vol. 151, p. 106178, Dec. 2022, doi: 10.1016/j.compbiomed.2022.106178.
[37] Z. Liu *et al.*, “KAN: Kolmogorov-Arnold Networks,” 2024, *arXiv*. doi: 10.48550/ARXIV.2404.19756.
[38] Z. Liu, P. Ma, Y. Wang, W. Matusik, and M. Tegmark, “KAN 2.0: Kolmogorov-Arnold Networks Meet Science,” 2024, *arXiv*. doi: 10.48550/ARXIV.2408.10205.
[39] S. Somvanshi, S. A. Javed, M. M. Islam, D. Pandit, and S. Das, “A Survey on Kolmogorov-Arnold Network,” 2024, *arXiv*. doi: 10.48550/ARXIV.2411.06078.
[40] S. S. Costa, M. Pato, and N. Datia, “An empirical study on the application of KANs for classification,” in *Proceedings of the 2024 8th International Conference on Advances in Artificial Intelligence*, London United Kingdom: ACM, Oct. 2024, pp. 308–314. doi: 10.1145/3704137.3704166.
[41] A. Shaushenova, O. Kuznetsov, A. Nurpeisova, and M. Ongarbayeva, “Implementation of Kolmogorov–Arnold Networks for Efficient Image Processing in Resource-Constrained Internet of Things Devices,” *Technologies*, vol. 13, no. 4, p. 155, Apr. 2025, doi: 10.3390/technologies13040155.
[42] D. N. Moriasi, M. W. Gitau, N. Pai, and P. Daggupati, “Hydrologic and Water Quality Models: Performance Measures and Evaluation Criteria,” *Trans. ASABE*, vol. 58, no. 6, pp. 1763–1785, Dec. 2015, doi: 10.13031/trans.58.10715.
[43] G. M. Kovoor and L. Nandagiri, “Sensitivity Analysis of FAO-56 Penman–Monteith Reference Evapotranspiration Estimates Using Monte Carlo Simulations,” in *Hydrologic Modeling*, vol. 81, V. P. Singh, S. Yadav, and R. N. Yadava, Eds., in Water Science and Technology Library, vol. 81. , Singapore: Springer Singapore, 2018, pp. 73–84. doi: 10.1007/978-981-10-5801-1_6.
[44] J. Demšar, “Statistical Comparisons of Classifiers over Multiple Data Sets,” *Journal of Machine Learning Research*, vol. 7, no. 1, pp. 1–30, 2006.
[45] C. Bergmeir and J. M. Benítez, “On the use of cross-validation for time series predictor evaluation,” *Information Sciences*, vol. 191, pp. 192–213, May 2012, doi: 10.1016/j.ins.2011.12.028.
[46] V. Cerqueira, L. Torgo, and I. Mozetič, “Evaluating time series forecasting models: an empirical study on performance estimation methods,” *Mach Learn*, vol. 109, no. 11, pp. 1997–2028, Nov. 2020, doi: 10.1007/s10994-020-05910-7.
[47] L. Grinsztajn, E. Oyallon, and G. Varoquaux, "Why do tree-based models still outperform deep learning on typical tabular data?," Advances in Neural Information Processing Systems (NeurIPS), vol. 35, pp. 507–520, 2022.