www.editada.org

International Journal of Combinatorial Optimization Problems and Informatics, 18(1), 2027, 1-13. ISSN: 2007-1558.
https://doi.org/10.61467/2007.1558.2027.v18i1.1473

Adaptive PSO-Optimized Random Forest for Power Output Prediction in Combined Cycle Power Plants

Norberto Hernández Romero[1], Juan Carlos Seck Tuoh Mora[1], Joselito Medina Marín[1], Mario Oscar Ordaz Oliver[2], Amadeo Manuel Hernández Hernández[2], Angélica Espejel Rivera[2]

[1] Área Académica de Ingeniería, Instituto de Ciencias Básicas e Ingeniería, Universidad Autónoma del Estado de Hidalgo

[2] Departamento de Ingeniería Eléctrica y Electrónica, Tecnológico Nacional de México, Campus Pachuca

nhromero@uaeh.edu.mx, jseck@uaeh.edu.mx, jmedina@uaeh.edu.mx, mario.oo@pachuca.tecnm.mx, amadeo.hh@pachuca.tecnm.mx, maria.er@tecnm.mx

✉Corresponding author: Norberto Hernández Romero

Abstract. A hybrid model (RF and PSO) is proposed to predict power generation in combined-cycle power plants. The study was conducted using 9,568 operational records collected over six years, with ambient temperature, atmospheric pressure, relative humidity, and exhaust vacuum as input conditions. Upon examining the dataset for physical consistency and identifying ambient temperature as the primary predictive variable in terms of power output, a correlation of -0.9485 was also obtained. During the training process, the baseline RF with grid search optimization achieved an RMSE of 3.334528 MW. Adaptive-PSO was evaluated by applying linear, exponential, and oscillatory update schemes. The results show that the oscillatory variant had the best cross-validation RMSE of 3.329416 MW, while the linear and exponential variants exhibited the lowest test RMSE of 3.088650 MW. A statistically significant difference was found between the oscillatory PSO variant and the reference model according to the Wilcoxon signed-rank test.

Keywords: Combined cycle power plant, power output prediction, Random Forest, adaptive Particle Swarm Optimization, hyperparameter optimization.

Article information
Received: Jun 10, 2026
Accepted: Aug 01, 2026

1 Introduction

Predicting electrical power output for combined cycle power plants (CCPPs) is an issue of great practical and economic concern. When these plants operate at full load under baseload conditions, short-term power output forecasts support efficient participation in electricity markets, improve dispatch scheduling, and facilitate maintenance planning and plant operation. However, power output does not depend on a single variable; rather, it results from nonlinear interactions among operating and environmental factors. Some of the most influential are ambient temperature (AT), atmospheric pressure (AP), relative humidity (RH), and steam turbine exhaust vacuum (V), the variations of which can cause substantial variation in the thermodynamic performance of the system.

Recent studies have reported major advances in machine-learning-based predictive modeling of CCPP power output. Tufekci (2014) demonstrated that ensemble-based methods, particularly Bagging with REPTree, can surpass standard thermodynamic methods. Subsequently, Lorencin et al. (2019) used genetic algorithms to design multilayer perceptron architectures automatically, and Alketbi et al. (2020) compared several regression algorithms and found RF to be a suitable compromise between accuracy and robustness for this task.

These results reinforce that data-driven models provide a reliable substitute for more complex or highly parameterized physical formulations. Likewise, Santarisi and Faouri (2021) studied different regression methods and demonstrated that boosting methods perform competitively in the industrial context. In parallel, Sun et al. (2021) applied deep learning architectures (recurrent and convolutional neural networks) for real-time power prediction in thermoelectric systems and achieved significant accuracy and computational efficiency improvements over classical models. Overall, there is ample evidence that power prediction in energy systems is improved by models dealing with nonlinear dependencies, variable interactions, and complex patterns in large datasets.

More recently, metaheuristic optimization has emerged as a promising approach for improving predictive performance in CCPP applications. Zhao and Foong (2022) designed a neural network optimized by an electrostatic discharge algorithm, whereas Sharma et al. (2023) proposed advanced sequential architectures for cyclic operating conditions in thermoelectric plants. Although promising results have been reported for these methods, the configuration of these methods and their computational effort may preclude practical applications unless they are carefully managed in terms of accuracy, interpretability, and training expenditure. Hybrid method approaches where the ensemble models are associated with the optimization algorithms offer a likely alternative in this respect.

Recent studies by Asghar et al. (2023), Song et al. (2024), Ameen and Hamarash (2025), Brahim et al. (2025), Dikra et al. (2026), and Kurker (2026) highlight both the persistence of this problem and the wide range of approaches proposed to address it (neural networks, super learner models, explainable approaches, and optimized hybrid schemes). Taken together, this evidence suggests that more generalizable and computationally efficient tuning procedures are still needed. Likewise, recent state-of-the-art reviews continue to identify RF as a strong alternative because of its robustness to overfitting, its ability to model nonlinear relationships, and its solid performance in supervised regression settings. However, few studies have explored RF combined with adaptive metaheuristic tuning schemes that preserve robustness and interpretability while enabling flexible hyperparameter search.

These insights imply that hybrid RF techniques augmented with adaptive optimization approaches warrant further investigation for hyperparameter tuning. This is particularly relevant in energy applications where even small gains in predictive accuracy translate to significant operational and economic improvements. Thus, the proposed model presents an efficient power output prediction algorithm that retains the robustness and interpretability provided by RF while using an adaptive strategy to probe the hyperparameter space.

This study contributes to predicting the power output of combined cycle power plants in three ways. First, it introduces a hybrid Random Forest–Particle Swarm Optimization model for understanding the impact of adaptive hyperparameter optimization on predictive performance. Second, it proposes an adaptive PSO scheme, which dynamically updates inertia, cognitive, and social weights throughout the optimization process. Third, it evaluates three adaptive update strategies—linear, exponential, and oscillatory—to see how this would change search behaviour and predictive results and compares them using a grid-search-optimized RF model as a yardstick. The hybrid model is validated by cross-validation to give a reliable measure of generalisation performance; furthermore, the study paints a detailed picture of the RF hyperparameter configurations in CCPP power output prediction tasks.

The rest of this paper is organized as follows. Section 2 presents the statistical analysis of the dataset used. Section 3 describes the fundamentals of RF, Particle Swarm Optimization, and the proposed hybridisation scheme. Section 4 presents the experiments and results obtained. Finally, Section 5 summarizes the main conclusions of the study.

2 Statistical Analysis of the CCPP Dataset

Figure 1 illustrates the main components and operating cycle of the CCPP, comprising a gas turbine, a steam turbine, electric generators, a heat recovery steam generator (HRSG), and a condenser. The cycle begins with the combustion of natural gas, which drives the gas turbine and transmits mechanical torque to the associated electric generator. The residual heat from the gas turbine exhaust gases is recovered in the HRSG to generate steam, which then drives the steam turbine and delivers mechanical torque to a second electric generator. Finally, the steam at the outlet of the steam turbine is condensed and reintroduced into the HRSG to close the cycle. The dataset was collected over a period of six years (2006–2011) in a 480 MW CCPP. The resulting dataset contains 9,568 instances sampled over 674 days of full-load operation. The dataset was obtained from the UCI Machine Learning Repository (Tüfekci and Kaya, 2014). The dataset exhibits clear physical patterns consistent with thermodynamic behaviour. The input variables are ambient temperature (AT), exhaust vacuum (V), atmospheric pressure (AP) and relative humidity (RH), while electrical power output (PE) is the target variable.

Table 1 reports the descriptive statistics: ambient temperature ranges from 1.8 to 37.1 °C, with a mean of 19.7 °C, a coefficient of variation of 37.92%, slight negative skewness, and a platykurtic distribution. Exhaust vacuum ranges from 25.44 to 81.6 cm Hg, with a mean of 54.3 and a coefficient of variation of 23.4%. Atmospheric pressure is the most stable variable, with variability of 0.586%, whereas relative humidity has a mean of 73% and moderate variability of 19.92%. Power output ranges from 420 to 496 MW, with a mean of 454 MW and low variability of 3.8%. The interquartile range (IQR) reveals that, indeed, there are considerable differences among those values which cover some part of the data. Atmospheric pressure (AP) has the lowest IQR of all (8.16 mbar)—indicating stable operating conditions and suggesting less contribution to prediction variability. In contrast, exhaust vacuum (V) has the highest IQR (24.80 cm Hg), making it an important but more challenging factor to model. The ambient temperature (AT) has a moderate IQR (12.21 °C), yet in combination with its very strong correlation with power output (PE), it emerges as the most critical variable in the system.

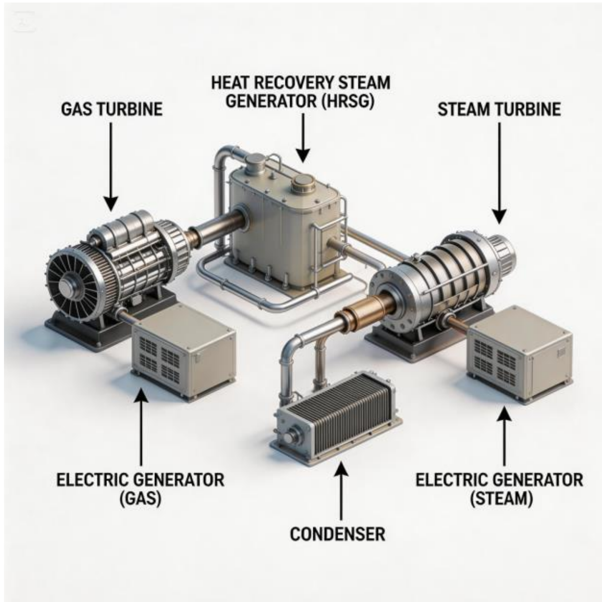


Fig. 1. *Schematic diagram of the combined cycle power plant.*

Table 1. Descriptive statistics for the CCPP dataset.

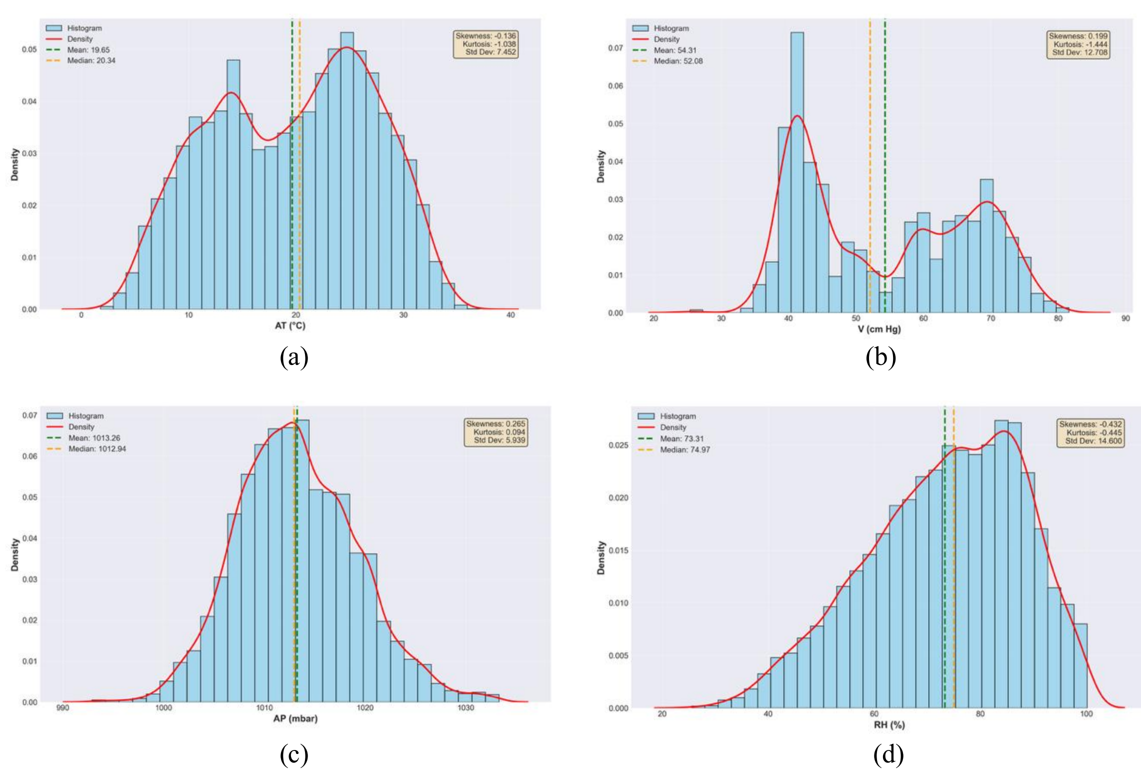
Variable	Min	Max	Mean	Variance	Std	Range	Skewness	Kurtosis	IQR
AT	1.81	37.1100	19.651200	55.53940	7.45250	35.30	-0.1364	-1.0375	12.21000
V	25.36	81.560	54.305800	161.4905	12.7079	56.20	0.1985	-1.4443	24.8000
AP	992.89	1033.30	1013.2591	35.26920	5.93880	40.41	0.2654	0.0942	8.1600
RH	25.56	100.160	73.309000	213.1678	14.6003	74.60	-0.4318	-0.4445	21.5025
PE	420.26	495.760	454.36500	291.2823	17.0670	75.50	0.3065	-1.0485	28.6800

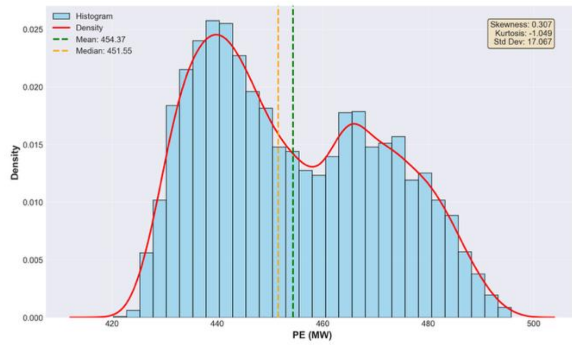
The correlation matrix between the variables is shown in Figure 2. The most notable result is the very strong negative correlation between ambient temperature and power output ($r = -0.9485$), indicating that higher temperatures are associated with lower power generation. Exhaust vacuum also shows a strong negative correlation of ($r=-0.87$) with output, whereas atmospheric pressure and relative humidity show moderate positive correlations of 0.52 and 0.39, respectively. The high correlation between ambient temperature and exhaust vacuum also shows some degree of collinearity between these predictors.

Fig. 2. *Correlation matrix of the input and output variables.*

The figure 3 shows that the variables do not follow strictly normal empirical distributions.

The variables exhibit different distributional patterns: AT is slightly negatively skewed and platykurtic, V is markedly platykurtic, AP is the closest to normality despite its positive skewness, RH shows moderate negative skewness, and PE has a flattened distribution with positive skewness. The Kolmogorov–Smirnov test confirms that none of the variables follows a strictly normal distribution.





(e)

Fig. 3. *(a) Histogram of Ambient Temperature, (b) Histogram of Exhaust Vacuum,*

(c) Histogram of Atmospheric Pressure, (d) Histogram of Relative Humidity

(e) Histogram of Electrical Power Output.

These statistical patterns suggest that simple linear formulations may not adequately capture the relationship between the predictors and power output. In particular, the dominant yet non-exclusive role of ambient temperature, together with intervariable interactions and departures from normality, supports the use of machine learning models capable of capturing nonlinear dependencies and complex feature interactions.

3 Hybrid Random Forest with PSO

3.1 Random Forest

Random Forest is an ensemble technique that constructs multiple decision trees from resampled subsets of the training set and averages their predictions (Breiman, 2001). Given a dataset of N observations $\{(x_i, y_i)\}$ with $y_i \in \mathbb{R}$, T subsets are created via bootstrap sampling (with replacement). In regression settings, each tree is trained on a bootstrap sample, and only a randomly selected subset of m features are considered at each node; typically, m is chosen according to the square root of the total number of features. Node splits are selected according to the reduction in variance. For a node with n values of y_i with mean $\bar{y}$, variance can be written as $\frac{1}{n}\sum(y_i - \bar{y})^2$. If a node is split into two children with n_i and n_d elements respectively, the split gain is computed as follows:

$$\Delta = \text{Variance(parents)} - \frac{n_i}{n}\text{Variance(left child)} - \frac{n_d}{n}\text{Variance(right child)} \tag{1}$$

The split that maximizes Δ is chosen. Each tree is grown without pruning until a stopping criterion is reached. It is passed through each tree to determine the new data point x^* and averaged as follows for new output:

$$\hat{y}^* = \frac{1}{T}\sum_{t=1}^{T} f_t\,(x^*) \tag{2}$$

where $f_t(x^*)$ is the prediction from an individual tree.

The model behaviour is regulated by five classical hyperparameters in Python's scikit-learn library (Python 3.12.9, scikit-learn 1.8.0). The main hyperparameters considered in this study are n_estimators, max_depth, min_samples_split, min_samples_leaf, and max_features. The number of trees, n_estimators, controls the ensemble size and affects the trade-off between variance reduction and computational cost. The maximum depth, max_depth, accounts for model complexity, whereas min_samples_split and min_samples_leaf limit tree growth and mitigate overfitting. Finally, max_features determines the proportion of predictors considered at each split, influencing both tree diversity and correlation within the forest. Together with bootstrap sampling and random feature selection, appropriate hyperparameter tuning helps RF achieve robust regression performance.

3.2 Particle Swarm Optimization

In classical PSO, the inertia, cognitive and social weights are constant in the optimization process. On the other hand, this fixed setup may restrain the algorithm as well from successfully balancing exploitation and exploration. Adaptive PSO extends the classical formulation by dynamically changing these weights according to iteration progress or the evolutionary state of the swarm, thus making it more convergent and less likely to get stuck in local optima (Zhan et al., 2009). Consider a swarm of P particles moving in a D-dimensional search space. At iteration t, each particle i is characterised by its position $\mathrm{x}_i(t) = (x_{i1}, x_{i2}, \dots, x_{iD})$, velocity $\mathbf{v}_i(t) = (v_{i1}, v_{i2}, \dots, v_{iD})$; personal best position $\mathrm{p}_i^{\mathrm{best}}$; and the swarm´s global best position $\mathrm{g}^{\mathrm{best}}$. Adaptive PSO uses the same update equations as classical PSO, but with time-varying parameters, defined by:

$$v_{id}(t+1) = w(t) \cdot v_{id}(t) + c_1(t) \cdot r_1 \cdot (p_{id}^{\mathrm{best}} - x_{id}(t)) + c_2(t) \cdot r_2 \cdot (g_d^{\mathrm{best}} - x_{id}(t)) \tag{3}$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1) \tag{4}$$

where r_1, r_2 are random numbers in the interval [0,1], $w(t)$ is the inertia weight, $c_1(t)$ is the cognitive weight, and $c_2(t)$ is the social weight. These parameters vary across iterations according to the selected adaptation strategy. In the linear strategy, the inertia weight decreases linearly at each iteration, from an initial value to a final value over the maximum number of iterations T_{max}. The cognitive weight increases linearly from its minimum c_1^{min} to its maximum value c_2^{max}, progressively strengthening each particle's individual search. Conversely, the social weight starts high and decreases linearly, encouraging stronger collective guidance early in the search and reducing it as optimization progresses. For the exponential and oscillatory strategies, the same parameters follow the corresponding nonlinear update patterns shown in Table 2.

Table 2. Adjustment strategies for the inertia, cognitive and social weights.

Strategy	Inertia weight	Cognitive weight	Social weight
Linear	$w(t) = w_{max} + \frac{w_{min} - w_{max}}{T_{\mathrm{max}}} \cdot t$	$c_1(t) = c_1^{max} - \frac{c_1^{min} - c_1^{max}}{T_{\mathrm{max}}} \cdot t$	$c_2(t) = c_2^{\mathrm{min}} + \frac{c_2^{\mathrm{max}} - c_2^{\mathrm{min}}}{T_{\mathrm{max}}} \cdot t$
Exponential	$w(t) = w_{max} - (w_{min} - w_{max}) \cdot e^{-\frac{5t}{Tmax}}$	$c_1(t) = c_1^{\mathrm{max}} + (c_1^{min} - c_1^{\mathrm{max}}) \cdot \left(e^{-\frac{5t}{Tmax}}\right)$	$c_2(t) = c_2^{max} + (c_2^{min} - c_2^{max})\left(1 - e^{-\frac{5t}{Tmax}}\right)$
Oscillatory	$osc_1(t) = 0.5 + 0.5 \cdot \cos(2\pi t/T_{max})$ $w(t) = w_{max} - (w_{min} - w_{max}) \cdot osc_1(t)$	$c_1(t) = c_1^{\mathrm{max}} + (c_1^{min} - c_1^{\mathrm{max}}) \cdot osc_1(t)$	$c_2(t) = c_2^{\mathrm{max}} - (c_2^{\mathrm{min}} - c_2^{\mathrm{max}}) \cdot osc_1(t)$

3.3 Hybrid Random Forest

Algorithm 1 summarizes the proposed RF–adaptive PSO procedure for predicting CCPP electrical power output. The algorithm takes as input a dataset of 9,568 samples, each containing four environmental features—ambient temperature, exhaust vacuum, atmospheric pressure, and relative humidity—with power output as the target variable. The procedure returns the optimal hyperparameter configuration, performance metrics, and predicted power values. The algorithm proceeds in several phases. First, the preprocessing stage splits the dataset into 80% training and 20% testing subsets, followed by standard scaling to ensure that all features have zero mean and unit variance. Next, a baseline RF model is trained and evaluated to provide a performance benchmark. Then, the PSO algorithm is initialised with N particles, each representing a candidate solution in a five-dimensional hyperparameter space. Subsequently, the optimization loop runs for T_{max} iterations. For each particle, the continuous position is decoded into discrete hyperparameters, and an RF model is then evaluated using five-fold cross-validation with Root Mean Square Error as the fitness metric. Local and global best positions are updated whenever a lower error is found. After all particles have been evaluated, their velocities and positions are updated using the standard PSO equations with stochastic random components.

The global best position is then decoded into the optimal hyperparameters, and a final RF model is trained on the complete training set. Predictions are generated for the unseen test set, and performance is assessed using MAE, RMSE, and R^2. Finally, the algorithm returns the optimal hyperparameters, performance metrics, predictions, and feature-importance scores. This implementation includes cross-validation to reduce overfitting risk, a baseline model for comparison, and fixed settings to ensure reproducibility. The selected parameter settings balance computational cost and optimization quality for the CCPP power prediction task.

Input: Dataset $\mathcal{D} = \{(AT_i,\ V_i,\ AP_i,\ RH_i,\ PE_i)\}_{i=1}^{9568}$
Output: Optimal RF hyperparameters θ^*, test metrics $\mathcal{M}$, predictions $\hat{y}$
1. Preprocessing:
$X \leftarrow [AT, V, AP, RH], y \leftarrow PE$
$X_{train}, X_{test}, y_{train}, y_{test} \leftarrow$ train _test _split(X, y, 0.2)
$X_{train} \leftarrow$ StandardScaler(X_{train}), $X_{test} \leftarrow$ StandardScaler(X_{test})
2. Baseline:
Parameters
$\boldsymbol{f_{base}} = \mathbf{RF}(Parameters);\ \boldsymbol{f_{base}.fit(X_{train}, y_{train})}$
$\boldsymbol{\mathcal{M}_{base}} = \mathbf{evaluate}(\boldsymbol{f_{base}, X_{test}, y_{test}})$
3. PSO Initialization:
$\mathcal{L} \leftarrow [50, 3, 2, 1, 0.3], \mathcal{U} \leftarrow [500, 30, 20, 10, 1.0]$
$\mathcal{P} \leftarrow 30, T_{max} \leftarrow 20, w_{min} \leftarrow 0.2, w_{max} \leftarrow 1.2$
$c_1^{min} \leftarrow 0.5, c_1^{max} \leftarrow 2.5, c_2^{min} \leftarrow 0.5, c_2^{max} \leftarrow 2.5$
$\{x_i\}_{i=1}^{P} \sim \mathcal{U}(\mathcal{L}, \mathcal{U}), \{v_i\}_{i=1}^{P} \sim \mathcal{U}(-0.5, 0.5)$
4. Optimization:
$\mathbf{for}\ t \leftarrow 1\ \boldsymbol{to}\ T_{max}\ \boldsymbol{do}$
$w(t) \leftarrow w_{max} + \frac{w_{min} - w_{max}}{T_{max}} \cdot t,\ c_1(t) \leftarrow c_1^{max} - \frac{c_1^{min} - c_1^{max}}{T_{max}} \cdot t,$
$c_2(t) \leftarrow c_2^{min} + \frac{c_2^{max} - c_2^{min}}{T_{max}} \cdot t$
$\mathbf{for}\ i \leftarrow 1\ \boldsymbol{to}\ P\ \boldsymbol{do}$
$\theta \leftarrow round(x_i[1{:}3]) \cup \{x_i[4]\}$
$RMSE_i \leftarrow$ CV_score($RF(\theta), X_{train}, y_{train}, k\_fold = 5$)
$\boldsymbol{if}\ RMSE_i < f(pbest_i)\ \boldsymbol{then}$
$pbest_{pos_i} \leftarrow x_i$
$\boldsymbol{end}$
$\boldsymbol{if}\ RMSE_i < gbest_i\ \boldsymbol{then}$
$gbest_{pos_i} \leftarrow x_i$
$\boldsymbol{end}$
$\boldsymbol{end}$
$r_1, r_2 \sim \mathcal{U}(0,1)$
$v_i \leftarrow w(t)v_i + c_1(t)r_1(pbest_{pos_i} - x_i) + c_2(t)r_2(gbest_{pos} - x_i)$
$x_i \leftarrow clip(x_i + v_i, \mathcal{L}, \mathcal{U})$
$\boldsymbol{end}$
5. Final model:
$\theta^* \leftarrow decode(gbest_{pos})$
$f^* \leftarrow RF(\theta^*), f^*.$fit (X_{train}, Y_{train})
$\hat{y} \leftarrow f^*.$predict (X_{test})
6. Metrics:

$$\mathbf{MAE} = \frac{1}{n}\sum |y_i - \hat{y}_i|,\ \mathrm{RMSE} = \sqrt{\frac{1}{n}\sum (y_i - \hat{y}_i)^2},\ R^2 = 1 - \sum \frac{(y_i - \hat{y}_i)^2}{\sum (y_i - \bar{y})^2}$$

7. Output:
$\boldsymbol{\mathcal{M}} \leftarrow \{MAE, RMSE, R^2\}$
return $\theta^*, \boldsymbol{\mathcal{M}}, \hat{y}$

Algorithm 1. General procedure for RF hyperparameter optimization using adaptive PSO

4 Experimental Results

The dataset comprises 9,568 records, of which 80% were used for training and 20% for testing. The test set was kept separate from the optimization process. A random seed of 42 was used in all runs to ensure reproducibility. To establish a performance benchmark, the baseline RF model was first tuned using grid search, which evaluated all predefined hyperparameter combinations. In scikit-learn, this procedure was implemented with *GridSearchCV()* using 5-fold cross-validation. Table 3 lists the hyperparameter values explored in the grid search.

Table 3. Grid search for exhaustive exploration of the hyperparameters of an RF baseline.

Hyperparameters	Grid Search
n_estimators	[50, 100, 150, 200, 250, 300]
max_depth	[10, 15, 20, 25, 30, None]
min_samples_split	[2, 5, 10]
min_samples_leaf	[1, 2, 4,10]
max_features	[0.3, 0.5, 0.7]

The best baseline configuration used *n_estimators* = 250, *max_depth* = 20, *min_samples_split* = 2, *min_samples_leaf* = 10, and *max_features* = 0.7, achieving a cross-validation RMSE of 3.334528. On the test set, the optimized baseline achieved MAE = 2.206687 MW, RMSE = 3.089224 MW, and R^2 = 0.967099.

The adaptive PSO scheme was implemented using three weight-update strategies: linear, exponential, and oscillatory. The search intervals for the five RF hyperparameters are given in Table 4. At each iteration, every PSO particle was evaluated by training an RF model using fixed 5-fold cross-validation. Table 4 also reports on the main PSO settings, including swarm size, number of iterations, and the inertia, cognitive, and social weight ranges for each adaptation strategy.

Table 4. Configuration of the PSO parameters and search ranges for the linear, exponential and oscillatory strategies.

Strategy	PSO parameters
Search ranges	n_estimators = [50, 500], max_depth = [3, 30], min_samples_split = [2, 20], min_samples_leaf = [1, 10], max_features = [0.3, 1.0]
Linear, exponential, oscillatory	$P = 30, M = 20, w_{max} = 1.2, w_{min} = 0.2, C_1^{max} = 2.5, C_1^{min} = 0.5, C_2^{min} = 0.5, C_2^{max} = 2.5$

All experiments were performed on a PC equipped with an Intel Core Ultra i7 processor and 16 GB of RAM, running Windows 11 and Python 3.12. Figure 4 presents the convergence of the cost function for the three PSO strategies over 20 iterations, showing that the oscillatory strategy achieved the lowest RMSE value (3.3295). To evaluate computational performance, execution time and memory consumption were measured using the time, psutil, and tracemalloc libraries. The results presented in Table 5 show that all PSO strategies outperformed the baseline approach in terms of computational efficiency. The exponential strategy achieved the shortest execution time, requiring 190.6292 s, representing a 33.8% reduction compared with the 287.9147 s required by GridSearchCV. Regarding memory consumption, the PSO strategies exhibited virtually negligible memory usage compared with the 164.28 MB recorded by GridSearchCV, corresponding to a reduction of more than 99.9%. In addition, the PSO-based models achieved the lowest peak memory values, a metric that reflects the maximum amount of memory allocated during execution.

Table 5. Time/Memory computational efficiency analysis.

Model	Time (s)	Used memory (MB)	Peak memory (MB)
Baseline	287.9147	164.28	7.62
Linear	262.9626	0.70	1.79
Exponential	190.6292	0.74	1.82
Oscillatory	216.3914	0.04	1.87

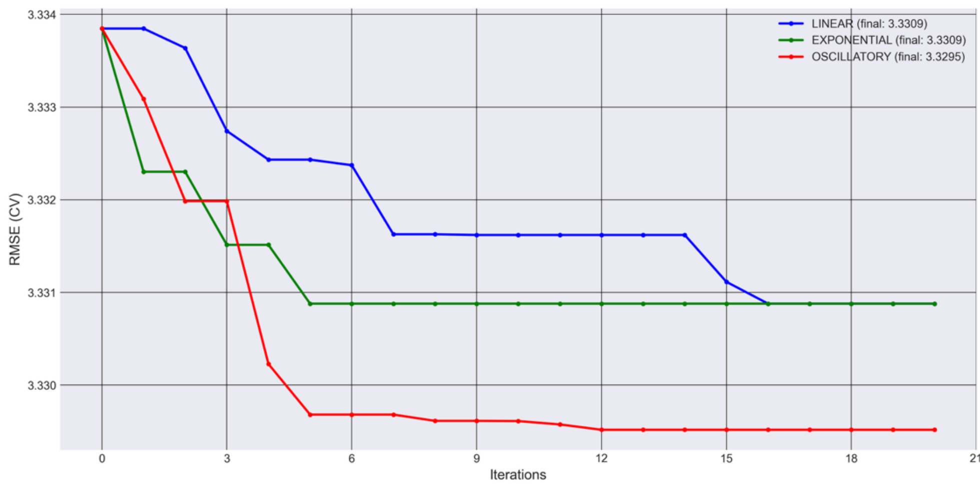


Fig. 4. *Convergence behavior of the adaptive PSO strategies.*

Table 6 compares the grid-search baseline with the PSO-optimized RF models. It reports training RMSE from 5-fold cross-validation, along with test MAE, RMSE, and R^2. Overall, model performance differed only marginally. The oscillatory strategy achieved the best training RMSE and the lowest test MAE, whereas the linear and exponential strategies produced the lowest test RMSE and slightly higher test R^2 values.

Table 6. Performance of the baseline and PSO models under training and testing conditions.

Model	Best training RMSE CV	Test MAE	Test RMSE	Test R^2
Baseline	3.334528	2.206687	3.089224	0.967099
Linear	3.330879	2.212921	3.088650	0.967111
Exponential	3.330879	2.212921	3.088650	0.967111
Oscillatory	3.329416	2.206245	3.089263	0.967098

Table 7 summarizes the RF hyperparameter configurations associated with the best cross-validation RMSE values. Across all four models, $min\_samples\_split = 2$ and $min\_samples\_leaf = 10$ were consistently selected as optimal values. The optimal max_depth values were between 20 and 21, while n_estimators ranged from 242 to 450, suggesting that large ensembles offered limited additional benefit within the explored search space. The oscillatory strategy found the best solution with 450 estimators. The linear and exponential strategies converged to the same configuration with 242 estimators, whereas grid search selected 250 estimators.

Table 7. Characterisation of RF model hyperparameters by grid search and adaptive PSO.

Model	*n_estimators*	*max_depth*	*min_samples_split*	*min_samples_leaf*	*max_features*
Baseline	250	20	2	10	0.700000
Linear	242	21	2	10	0.628733
Exponential	242	21	2	10	0.625567
Oscillatory	450	20	2	10	0.649754

Statistical significance was assessed using 10 repetitions of 5-fold cross-validation for the four models. Figure 5 presents the resulting RMSE distributions as box plots. The oscillatory model achieved the lowest minimum RMSE, while the linear and exponential models showed the smallest interquartile range. In contrast, the baseline exhibited the widest interquartile range.

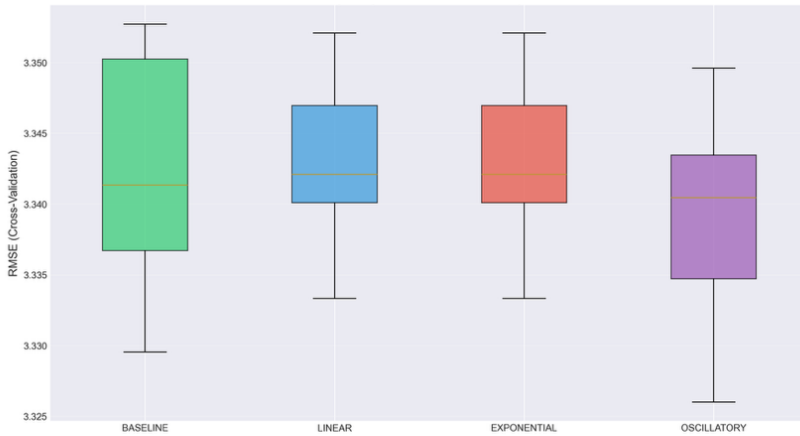


Fig. 5. *Comparison of the RMSE distributions across baseline and adaptive PSO strategies, based on 10 repetitions of 5-fold cross-validation.*

Statistical significance was assessed using Friedman and Wilcoxon nonparametric tests based on 30 repetitions of 5-fold cross-validation. The Friedman test revealed significant differences among the four evaluated models ($\chi^2 = 19.0277$, $p = 0.000270$). Subsequent Wilcoxon post hoc comparisons showed that the Linear and Exponential PSO strategies did not significantly outperform the optimized Baseline model ($p = 0.5564$), whereas the Oscillatory strategy achieved a statistically significant improvement ($p < 0.001$). Furthermore, the Oscillatory strategy significantly outperformed both the Linear and Exponential variants ($p = 0.00038$). The Friedman ranking also identified the Oscillatory strategy as the best-performing approach (rank = 1.63), achieving the lowest mean RMSE (3.3363). These results indicate that the oscillatory PSO strategy provides a consistent and statistically significant improvement in RF model performance compared with the baseline and the other PSO variants.

For the RF model optimized using the oscillatory PSO strategy, a sensitivity analysis was conducted to evaluate the effect of four configuration factors: the initial random seed, PSO population size, number of PSO iterations, and number of folds used in k-fold cross-validation. This analysis assessed the robustness of the optimized model under variations in stochastic initialization, swarm configuration, optimization effort, and validation structure. The following parameter ranges were considered: random_seed = [0, 42, 50, 100, 500, 1000, 3000], population_size = [10, 20, 30, 50], number_iterations = [10, 20, 30, 50, 100], and number_kfolds_CV = [3, 5, 7]. Table 8 summarizes the global sensitivity statistics for these factors in terms of mean RMSE, standard deviation, and coefficient of variation (CV). The results indicate that the oscillatory PSO-optimized RF model is highly stable under variations in the initial random seed, population size, and number of iterations, as shown by the very low coefficients of variation. The number of folds used in cross-validation produced the highest variability, although this variation remained relatively low. Therefore, the model can be considered robust, with the validation scheme having a more noticeable influence on RMSE estimation than the PSO configuration parameters.

Table 8. Global sensitivity analysis statistics.

Statistic	Random seed	Population-size	Number of iterations	Number of folds
Mean RMSE	3.093009	3.099498	3.092054	3.354943
Standard deviation	0.003866	0.001721	0.002537	0.040023
CV (%)	0.120000	0.060000	0.080000	1.190000

To evaluate the potential presence of overfitting in the RF model optimized using the oscillatory PSO strategy, a generalization analysis was conducted using test proportions of 20%, 30%, 40%, and 50%, with 30 repetitions for each configuration. Performance degradation was calculated as the percentage increase in test RMSE relative to training RMSE, considering mild, moderate, and severe overfitting for degradation values exceeding 20%, 50%, and 100%, respectively.

Figure 6 shows that the model does not exhibit evidence of overfitting for any of the evaluated data partitions. RMSE degradation remained between 17.9% and 18.2%, consistently below the 20% threshold. For the 80/20 train-test split, the average RMSE was 3.0567 MW for training and 3.6051 MW for testing, while the maximum observed degradation was 18.2%. These results confirm the robustness of the model and its ability to generalize effectively to unseen data, with the 80/20 split providing the best overall performance.

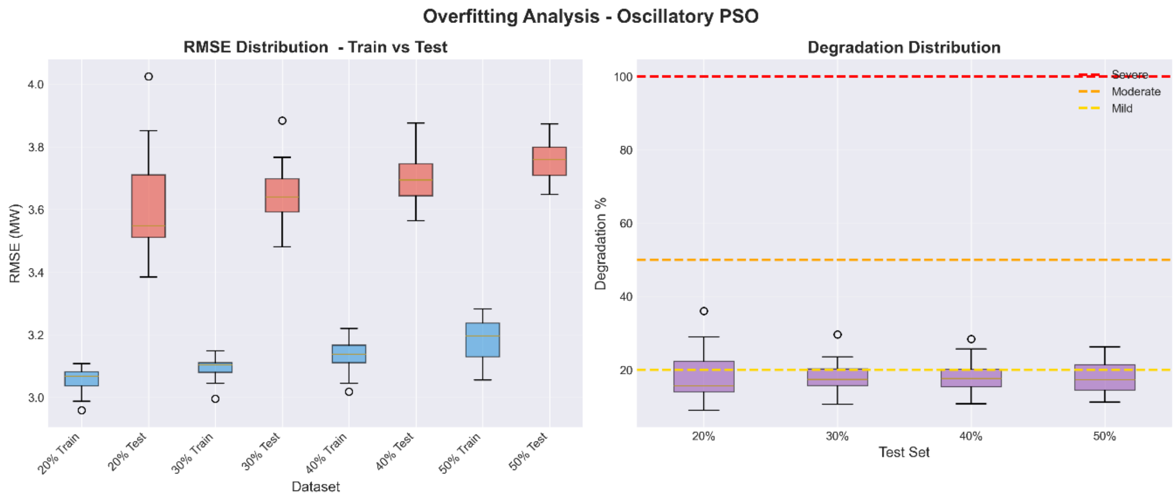


Fig. 6. *Overfitting Analysis of the Oscillatory PSO-Optimized RF Model Under Different Train-Test Split Ratios.*

To assess the influence of the thermodynamic variables on the RF model optimized with oscillatory PSO, 1,000 test samples were analyzed using SHAP. Figure 7 shows that AT has the strongest effect on predictions, followed by V, while AP contributes moderately and RH has little influence. Low AT and V values tend to increase predicted power output, whereas high values reduce it, indicating an inverse relationship with the target variable. Overall, SHAP confirms that AT and V are the main drivers of the model's predictions. These findings are consistent with the correlation analysis presented in Figure 2 and the thermodynamic behavior of combined cycle power plants.

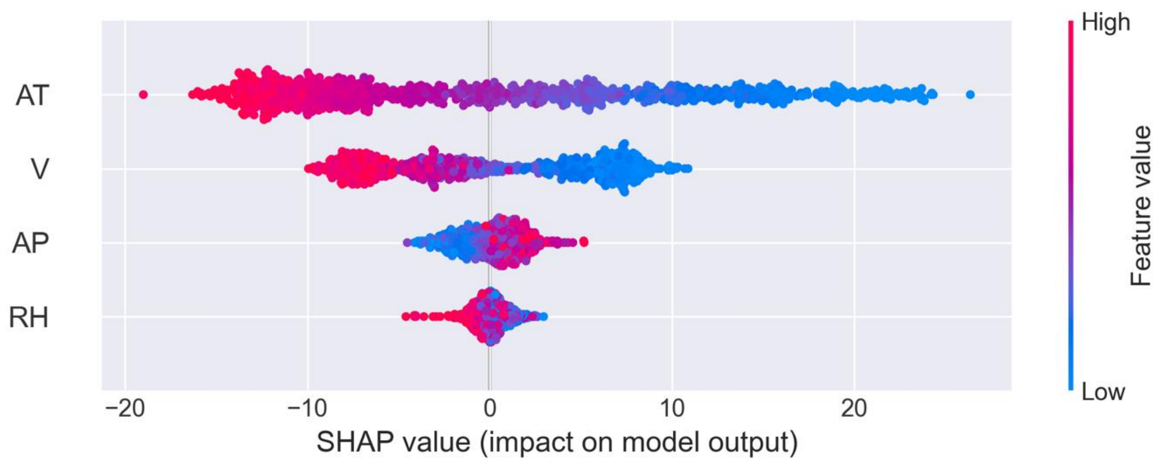


Fig. 7. *SHAP-based importance of the thermodynamic features (AT, V, AP and RH) in CCPP power output prediction.*

Table 9 compares the proposed RF-based models with previous studies using the same dataset. The proposed baseline RF achieved RMSE = 3.089224 MW and MAE = 2.206687 MW, outperforming several previously reported methods in terms of RMSE, including Tufekci (2014), Lorencin (2019), Saleel (2021), Ntantis (2024), Song (2024), and Kurker (2026). The oscillatory, linear, and exponential PSO variants produced results comparable to the baseline, indicating that a well-tuned PSO–RF model can achieve similar predictive performance. The obtained R^2 value of 0.967 is higher than the R^2 values reported by Saleel (2021) and Kurker (2026), although such comparisons should be interpreted cautiously because experimental setup may differ.

Table 9. Comparative analysis of the baseline RF, adaptive PSO-optimized RF and other studies.

	MAE	RMSE	R^2
Baseline RF	2.206687	3.089224	0.967099
PSO Oscillatory RF	2.206245	3.089263	0.967098
Saleel (2021)	Not reported	Not reported	0.94000
Tufekci (2014)	2.8180	3.7870	Not reported
Ntantis (2024)	Not reported	3.8395	Not reported
Lorencin (2019)	Not reported	4.305	Not reported
Song (2024)	2.2537	3.1772	Not reported
Kurker (2026)	2.4566	3.4885	0.9579

5 Conclusions

This study demonstrates that integrating RF with an adaptive PSO scheme provides a methodologically sound and competitive approach for predicting power output in combined cycle power plants. Based on 9,568 operating records, the statistical analysis confirmed the physical consistency of the problem and identified ambient temperature as the dominant variable, with a correlation coefficient of -0.9485 with power output, followed by exhaust vacuum with -0.87. These findings validate the need for nonlinear models capable of capturing complex interactions among environmental and operational variables. In predictive terms, the baseline model optimized through grid search achieved RMSE = 3.089224 MW, MAE = 2.206687 MW, and R^2 = 0.967099, explaining approximately 96.71% of the variability in electrical production. Relative to this reference, adaptive PSO enabled a broader exploration of the hyperparameter space and produced the best training solution with the oscillatory strategy, reducing the cross-validation RMSE from 3.334528 to 3.329416 MW. Statistical validation further supported these results. The Friedman test detected significant differences among the evaluated models (χ^2 = 19.0277, p = 0.000270), whereas Wilcoxon post hoc comparisons revealed that the oscillatory strategy significantly outperformed both the optimized baseline model and the other PSO variants. The Friedman rankings also identified the oscillatory approach as the best-performing strategy, confirming that the observed improvements were not attributable to random variation.

Beyond predictive accuracy, the adaptive PSO strategies demonstrated important computational advantages over GridSearchCV. The exponential strategy achieved the shortest execution time, requiring 190.6292 s, which represents a 33.8% reduction compared with the 287.9147 s required by GridSearchCV. Moreover, all PSO-based approaches exhibited negligible memory consumption relative to the 164.28 MB required by the baseline method, resulting in memory savings exceeding 99.9%.

The robustness of the proposed framework was assessed through an overfitting analysis using train-test splits ranging from 80/20 to 50/50. Across all evaluated configurations, RMSE degradation remained between 17.9% and 18.2%, consistently below the predefined overfitting threshold of 20%. These results indicate that the oscillatory PSO-optimized RF model maintains stable generalization performance and does not exhibit evidence of overfitting when applied to unseen data.

Model interpretability was enhanced through SHAP analysis, which identified ambient temperature (AT) as the most influential predictor, followed by exhaust vacuum (V), while atmospheric pressure (AP) and relative humidity (RH) showed comparatively lower contributions. These findings are consistent with both the correlation analysis and the known thermodynamic behavior of combined cycle power plants, providing additional confidence in the model's ability to capture meaningful physical relationships.

From a comparative perspective, the proposed approach achieved lower RMSE values than those reported by Tufekci (3.7870), Song (3.1772), and Kurker (3.4885). Overall, the results demonstrate that the oscillatory PSO-RF framework combines predictive accuracy, statistical significance, computational efficiency, robustness against overfitting, and model interpretability. These characteristics support its use as a reproducible and practically relevant tool for operation support, scheduling, and decision-making in CCPP environments. Future work should evaluate the proposed framework using additional industrial datasets and investigate the hybridization of machine learning models with other metaheuristic optimization techniques. Although the proposed framework demonstrated robust performance on the CCPP benchmark dataset, further validation using additional thermodynamic datasets is required to fully assess its external generalizability.

References

Alketbi, S., Nassif, A. B., Eddin, M. A., Shahin, I., & Elnagar, A. (2020). Predicting the power of a combined cycle power plant using machine learning methods. In *2020 International Conference on Communications, Computing, Cybersecurity, and Informatics (CCCI)* (pp. 1–5). IEEE. https://doi.org/10.1109/CCCI49893.2020.9256742

Ameen, A. S. H., & Hamarash, I. I. (2025). A machine learning models for predicting power output in gas turbines: A case study from Perdawood Power Station, Erbil, Iraqi Kurdistan. *Zanco Journal of Pure and Applied Sciences, 37*(5), 115–124. https://doi.org/10.21271/ZJPAS.37.5.9

Arakelyan, E. K., Boldyrev, I. A., Evseev, K. V., & Gorban, Y. A. (2021). Application of machine learning methods for optimizing the technical and economic performance of generating systems. *IOP Conference Series: Materials Science and Engineering, 1035*(1), 012011. https://doi.org/10.1088/1757-899X/1035/1/012011

Asghar, A., Ratlamwala, T. A. H., Kamal, K., Alkahtani, M., Mohammad, E., & Mathavan, S. (2023). Sustainable operations of a combined cycle power plant using artificial intelligence based power prediction. *Heliyon, 9*(9), e19562. https://doi.org/10.1016/j.heliyon.2023.e19562

Bishop, C. M. (1995). *Neural networks for pattern recognition*. Oxford University Press.

Breiman, L. (2001). Random forests. *Machine Learning, 45*(1), 5–32. https://doi.org/10.1023/A:1010933404324

Dikra, S. H., Gawanmeh, A., & Rabia, K. (2026). Data-driven analysis and optimization of combined cycle power plants using machine learning models. *Journal of Engineering and Technological Sciences, 58*(2), 202–216. https://doi.org/10.5614/j.eng.technol.sci.2026.58.2.4

Huang, G.-B., Zhu, Q.-Y., & Siew, C.-K. (2006). Extreme learning machine: Theory and applications. *Neurocomputing, 70*(1–3), 489–501. https://doi.org/10.1016/j.neucom.2005.12.126

Kurker, F. (2026). PSO–fmincon optimization-based explainable ensemble learning approach for reliable estimation of electrical power output in combined cycle power plants. *Electric Power Systems Research, 257*, 112969. https://doi.org/10.1016/j.epsr.2026.112969

Lorencin, I., Anđelić, N., Mrzljak, V., & Car, Z. (2019). Genetic algorithm approach to design of multi-layer perceptron for combined cycle power plant electrical power output estimation. *Energies, 12*(22), 4352. https://doi.org/10.3390/en12224352

Rouibah, B., Labdaoui, A., Aggoun, H., & Güler, İ. (2025). Power forecasting using ANN and ELM: A comparative evaluation of machine learning approaches. *Mathematical Modelling of Engineering Problems, 12*(1), 1–8. https://doi.org/10.18280/mmep.120101

Saleel, C. A. (2021). Forecasting the energy output from a combined cycle thermal power plant using deep learning models. *Case Studies in Thermal Engineering, 28*, 101693. https://doi.org/10.1016/j.csite.2021.101693

Santarisi, N. S., & Faouri, S. S. (2021). Prediction of combined cycle power plant electrical output power using machine learning regression algorithms. *Eastern-European Journal of Enterprise Technologies, 6*(8 [114]), 16–26. https://doi.org/10.15587/1729-4061.2021.245663

Sharma, H., Marinovici, L., Adetola, V., & Schaef, H. T. (2023). Data-driven modeling of power generation for a coal power plant under cycling. *Energy and AI, 11*, 100214. https://doi.org/10.1016/j.egyai.2022.100214

Song, Y., Park, J., Suh, M.-S., & Kim, C. (2024). Prediction of full-load electrical power output of combined cycle power plant using a super learner ensemble. *Applied Sciences, 14*(24), 11638. https://doi.org/10.3390/app142411638

Sun, L., Liu, T., Xie, Y., Zhang, D., & Xia, X. (2021). Real-time power prediction approach for turbine using deep learning techniques. *Energy, 233*, 121130. https://doi.org/10.1016/j.energy.2021.121130

Tüfekci, P., & Kaya, H. (2014). *Combined cycle power plant* [Data set]. UCI Machine Learning Repository. https://doi.org/10.24432/C5002N

Wang, J., Lu, S., Wang, S.-H., & Zhang, Y.-D. (2022). A review on extreme learning machine. *Multimedia Tools and Applications, 81*(29), 41611–41660. https://doi.org/10.1007/s11042-021-11007-7

Zhan, Z.-H., Zhang, J., Li, Y., & Chung, H. S.-H. (2009). Adaptive particle swarm optimization. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 39*(6), 1362–1381. https://doi.org/10.1109/TSMCB.2009.2015956

Zhao, Y., & Foong, L. K. (2022). Predicting electrical power output of combined cycle power plants using a novel artificial neural network optimized by electrostatic discharge algorithm. *Measurement, 198*, 111405. https://doi.org/10.1016/j.measurement.2022.111405