


Article

Short-Term Photovoltaic Power Prediction Based on Extreme Learning Machine with Improved Dung Beetle Optimization Algorithm

Yuhao Zhang ¹, Ting Li ^{1,*} , Tianyi Ma ¹, Dongsheng Yang ² and Xiaolong Sun ¹

¹ School of Mechanic and Electronic Engineering, Beijing Institute of Graphic Communication, Beijing 102600, China; egimtarupli@mail.com (Y.Z.); matianyi@bigc.edu.cn (T.M.); shannonsun7@gmail.com (X.S.)

² School of Information Science and Engineering, Northeastern University, Shenyang 110006, China; yangdongsheng@mail.neu.edu.cn

* Correspondence: liting@bigc.edu.cn

Abstract: Given the inherent volatility and intermittency of photovoltaic power generation, enhancing the precision of photovoltaic power predictions becomes imperative to ensure the stability of power systems and to elevate power quality. This article introduces an intelligent photovoltaic power prediction model based on the Extreme Learning Machine (ELM) with the Adaptive Spiral Dung Beetle Optimization (ASDBO) algorithm. The model aims to accurately predict photovoltaic power generation under multi-factor correlation conditions, including environmental temperature and solar irradiance. The computational efficiency in high-dimensional data feature conditions is enhanced by using the Pearson correlation analysis to determine the state input of the ELM. To address local optimization challenges in traditional Dung Beetle Optimization (DBO) algorithms, a spiral search strategy is implemented during the dung beetle reproduction and foraging stages, expanding the exploration capabilities. Additionally, during the dung beetle theft stage, dynamic adaptive weights update the optimal food competition position, and the levy flight strategy ensures search randomness. By balancing convergence accuracy and search diversity, the proposed algorithm achieves global optimization. Furthermore, eight benchmark functions are chosen for performance testing to validate the effectiveness of the ASDBO algorithm. By optimizing the input weights and implicit thresholds of the ELM through the ASDBO algorithm, a prediction model is established. Short-term prediction experiments for photovoltaic power generation are conducted under different weather conditions. The selected experimental results demonstrate an average prediction accuracy exceeding 93%, highlighting the effectiveness and superiority of the proposed methodology for photovoltaic power prediction.

Keywords: photovoltaic power generation; extreme learning machine; dung beetle optimization; correlation analysis; power prediction



Citation: Zhang, Y.; Li, T.; Ma, T.; Yang, D.; Sun, X. Short-Term Photovoltaic Power Prediction Based on Extreme Learning Machine with Improved Dung Beetle Optimization Algorithm. *Energies* **2024**, *17*, 960. <https://doi.org/10.3390/en17040960>

Academic Editor: Philippe Leclère

Received: 14 January 2024

Revised: 14 February 2024

Accepted: 17 February 2024

Published: 19 February 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Currently, many countries around the world are facing a challenge to increase energy demand while achieving environmental sustainability. In this context, to accelerate the development and popularization of renewable energy provides an effective solution to addressing global energy crisis and environmental issues. Solar power, as an abundant and environmentally friendly source of renewable energy, plays a significant role in achieving the efficient utilization of clean energy through photovoltaic (PV) power generation, contributing to the sustainability of energy development [1]. Nonetheless, the inherent randomness of sunlight and the rhythmic day–night cycles introduce fluctuations and intermittence in PV power generation. Moreover, various factors such as weather and environmental conditions impact the output of photovoltaic systems, contributing to the

significant unpredictability of power generation when large-scale PV power is integrated into the grid [2,3]. Consequently, this unpredictability has a negative impact on the grid in terms of planning. The accurate prediction of PV power generation that takes into consideration the impact of multi-factor coupled time-varying characteristics is effective in improving the stability of the grid, enhancing its ability to consume PV power, and improving the operating efficiency of PV power plants [4]. Therefore, it has attracted extensive attention for research. Depending on the exact time scale, PV power prediction is classified into short-term forecasting, medium-term forecasting, and long-term forecasting [5]. Short-term forecasting plays a crucial role in enhancing the reliability of the power system. However, it also faces challenges, such as the difficulty in accurately predicting weather changes and the need for higher precision and timeliness to meet the rapid scheduling and response demands of the power system. Furthermore, PV systems themselves exhibit complex nonlinear characteristics, making it difficult to accurately predict their output using simple mathematical models. Simultaneously, inaccurate predictions can jeopardize the stability of the power grid, complicating the efficient alignment of electricity supply and demand within the system. This situation may result in surplus power generation and avoidable carbon emissions, contributing to adverse environmental consequences. Consequently, addressing this issue has emerged as a focal point in current research.

Accurate short-term PV power prediction is crucial for optimizing the lifespan of storage devices such as batteries and for efficiently managing the production, delivery, and storage systems of the power grid on a daily/hourly basis. Currently, the commonly used methods of short-term PV forecasting include physical forecasting methods [6,7], statistical analysis methods, and machine learning methods [8]. Among them, the physical forecasting method involves the calculation of real-time cloud images of PV power generation and the conversion of solar energy into electricity. Mandal et al. proposed to achieve ultra-short-term PV power forecasting by simulating the transmission of solar irradiance and the generation of photovoltaic component power [9]. Monteiro et al. used the data of the meteorological forecast in combination with the geographical location of PV power plants to forecast photovoltaic power [10]. The above-mentioned method necessitates precise location data for the PV power station, along with numerical meteorological forecasts. Nevertheless, it exhibits a certain vulnerability to interference in the face of intricate weather conditions, resulting in a degree of inaccuracy in its predictions. The statistical analysis method mainly involves the counting, analysis, and processing of historical data collected from photovoltaic power stations, including Markov chains [11], Grey system theory [12], linear regression [13], etc. Sperati et al. reduced the error in prediction by constructing a probability density function for PV power prediction under different weather conditions [14]. Feng et al. analyzed the state transition matrix of the Markov chain, showing the different transition trends of solar energy at different intervals [11]. Then, power prediction models were established for different time periods, with good prediction results achieved in a grid connection. After dividing the past power sequence seasonally, Ding et al. applied the Grey theory model to reconstruct the original power data, thus reducing the randomness of the original data and improving the accuracy of power prediction [12]. However, the prediction results are highly dependent on the data, and the modeling process is relatively complex and sensitive to model parameters [8]. This leads to a significant bias in the statistical analysis model.

At present, the commonly used methods of machine learning include Support Vector Machines (SVMs) [15], neural networks [16], Extreme Learning Machines (ELM) [17], etc. Meng proposed a method in which a correlation analysis was performed to select the factors closely correlated with PV power as inputs before a prediction model was constructed using an improved SVM to enhance the accuracy of prediction [18]. Mellit et al. proposed to reduce the error in PV forecasting through a PV forecasting model based on an artificial neural network (ANN), in which such meteorological data as the irradiance, temperature, and atmospheric pressure were taken as inputs [19]. Chen et al. adopted the Long Short-Term Memory (LSTM) neural network to make PV power prediction. This method enables

the short-term prediction of PV power generation, thus reducing the Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) of the prediction results [20]. The above-mentioned prediction model takes meteorological and weather factors into account, effectively reducing prediction errors. However, given its relatively intricate modeling process, it usually demands multiple iterations to converge to the optimal solution, leading to a slower training speed. Moreover, the quality of the prediction results is significantly influenced by the selection of model parameters.

Compared to traditional neural networks, the ELM presents advantages in rapid learning and a concise model structure. It simplifies the training process by randomly setting input weights and hidden thresholds, avoiding the need for continuous adjustments. This reduction in complexity effectively addresses common issues, such as lengthy training periods [21,22]. However, optimizing the model parameters is crucial due to the significant impact of the initial random parameters on the prediction outcomes in the ELM model [23]. Currently, scholars primarily employ intelligent algorithms to optimize hyperparameters. Common algorithms include Grey Wolf Optimization (GWO), War Strategy Optimization (WSO), the Coati Optimization Algorithm (COA), the Sparrow Search Algorithm (SSA), the Crow Search Algorithm (CSA), etc. In [24], the SSA was used to enhance the PV power prediction with the improved Back Propagation (BP) neural network. In [25], the ICSEA was employed to optimize the hyperparameters of the Least Squares Support Vector Machine for short-term PV power prediction. However, the above algorithms still tend to converge to local optimal solutions during the search phase.

With the ongoing development of swarm intelligent optimization algorithms, new algorithms are continually emerging. In 2022, Xue et al. introduced the Dung Beetle Optimization (DBO) algorithm and successfully applied it to various engineering design problems [26]. Inspired by the natural behaviors of dung beetles, such as rolling dung balls, foraging, and breeding, the DBO algorithm optimally allocates diverse survival tasks to individuals based on the division of labor within the dung beetle population. This results in rapid convergence and high-precision solutions compared to traditional algorithms. In [27], the DBO algorithm was applied to wireless sensor networks, introducing a DV-Hop localization algorithm optimized based on the DBO algorithm. In [28], the DBO algorithm was utilized to optimize the BP neural network for predicting the debonding failure of beams. In [29], the Halton sequence was employed exclusively to initialize the positions of individuals within the population, aiming to control the initial spatial distribution of the population. However, the mentioned literature either utilized the original algorithm, or the improvement only exhibited positive effects during the initial stages of optimization. Nevertheless, there still exists an imbalance between global exploration and local exploitation capabilities, making it susceptible to local optima [30].

To comprehensively address the challenges outlined, this study employs Adaptive Spiral Dung Beetle Optimization (ASDBO) to optimize the hyperparameters of the Extreme Learning Machine (ELM) for enhancing the accuracy of PV power prediction. The overarching goal is to mitigate potential energy wastage stemming from inaccurate forecasts, thereby contributing to the reduction in environmental burdens and advancing sustainable development in clean energy. In assessing the accuracy of the proposed models, encompassing the BP, LSTM, GRU, ELM, DBO-ELM, and ASDBO-ELM, various conditions were considered for forecasting PV power. The experimental findings underscore the high accuracy achieved by the model introduced in this paper, particularly in short-term PV power prediction. The contributions of this study are outlined as follows:

- The DBO algorithm has been improved. Introducing a spiral search during dung beetle breeding and foraging altered the search pattern, enhancing the exploratory capabilities. In the stealing stage, a combined dynamic weighting strategy and Levy flight mechanism balanced the search diversity and convergence accuracy, preventing local optima.
- The discussion extensively explored dynamic factors influencing PV power. Employing the Pearson correlation coefficient for feature selection not only ensures a

more accurate alignment with real-world scenarios in specific experiments but also contributes to the advancement of the application and development of clean energy.

- To further improve the predictive performance of the ELM, ASDBO is employed to finely adjust the initial weights and thresholds. This approach eradicates randomness in parameter settings, ensuring subsequent sequence predictions are more dependable and precise.
- The ASDBO-ELM model was introduced to forecast short-term PV power under various conditions. The experimental results show that this study has the potential to enhance the accuracy of PV power predictions, thereby mitigating the energy wastage caused by inaccurate forecasts.

2. Theoretical Analysis

2.1. Extreme Learning Machine

The Extreme Learning Machine, as an innovative feedforward single-hidden-layer neural network, stands out for its capability to randomly generate biases and input weights for the hidden layer. This eliminates the necessity for configuring additional parameters. Specifying the number of hidden layer neurons during training allows for achieving a globally optimal solution, alleviating typical issues encountered in traditional neural networks such as slow training and overfitting. The network structure of the ELM is illustrated in Figure 1.

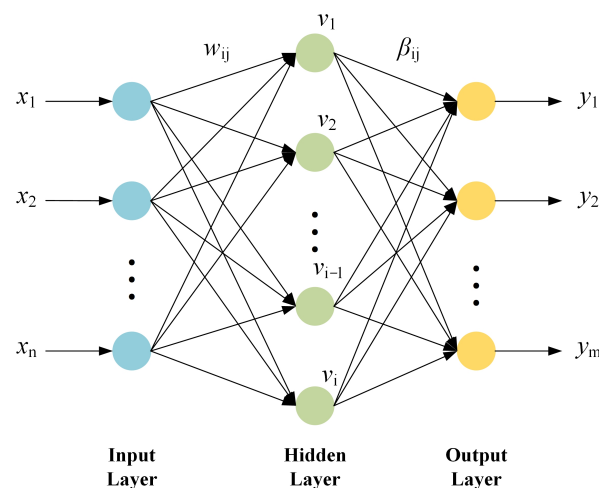


Figure 1. Structure chart of ELM.

In Figure 1, the input layer consists of n neuron nodes ($x_i \sim x_n$), corresponding to n input variables. In the hidden layer, there are k neuron nodes. The output layer consists of m neuron nodes ($y_i \sim y_m$), corresponding to m output variables. w_{ij} represents the weight from the input layer to the hidden layer; $v_1 \sim v_i$ represent the threshold of the hidden layer nodes; and β_{ij} represents the weight from the hidden layer to the output layer.

Let there be M samples, where $x_i = [x_{i1}, x_{i2}, \dots, x_{in}]^T$, $y_i = [y_{i1}, y_{i2}, \dots, y_{im}]^T$, x_i is an n -dimensional input vector, and y_i is an m -dimensional output vector. The number of hidden layer nodes is denoted as k , and the activation function is denoted as $G(\cdot)$. In the prediction model used in this article, the Sigmoid function is taken as the activation function, as shown in Equation (1), to compute the hidden layer output matrix $H_{w,v,x}$.

$$G(x) = \frac{1}{1 + e^{-x}} \quad (1)$$

The output obtained through the training of the ELM neural network is expressed as follows:

$$y_j = \sum_{i=1}^k \beta_i G(w_i \cdot x_j + v_i), j = 1, 2, \dots, M \quad (2)$$

where w_i represents the weights from the input layer to the hidden layer; v_i indicates the thresholds of the neurons in the hidden layer; β_i represents the weights between the hidden layer and the output layer; and $G(w_i \cdot x_j + v_i)$ denotes the activation function.

If the output of the network is presented as $Y = [y_1, y_2, \dots, y_M]^{m \times M}$, Equation (2) can be changed into the matrix form:

$$H_{w,v,x} \beta = Y \quad (3)$$

where β represents the $k \times m$ dimensional output weight matrix, and $H_{w,v,x}$ refers to the output matrix of the implied layer. This can be expressed as follows:

$$H_{w,v,x} = \begin{bmatrix} G(w_1 x_1 + v_1) & \cdots & G(w_k x_1 + v_k) \\ G(w_1 x_2 + v_1) & \cdots & G(w_k x_2 + v_k) \\ \vdots & & \vdots \\ G(w_1 x_M + v_1) & \cdots & G(w_k x_M + v_k) \end{bmatrix} \quad (4)$$

$$\beta = \begin{bmatrix} \beta_1^T \\ \beta_2^T \\ \vdots \\ \beta_k^T \end{bmatrix}_{k \times m} \quad Y = \begin{bmatrix} y_1^T \\ y_2^T \\ \vdots \\ y_M^T \end{bmatrix} \quad (5)$$

where k represents the number of hidden layer nodes, and M denotes the number of training samples. According to the definition of the generalized inverse matrix, Equation (6) is used to obtain the result.

$$\beta^* = H^+ Y \quad (6)$$

where H^+ represents the Moore–Penrose generalized inverse of the output matrix of the hidden layer.

2.2. Dung Beetle Optimization

The Dung Beetle Optimizer is inspired by the rolling, dancing, foraging, breeding, and stealing behaviors of dung beetles. Five different updating rules are designed to assist in finding high-quality solutions. Each dung beetle group consists of four types of agent beetles: rolling dung beetles, breeding beetles, foraging beetles, and stealing beetles.

2.2.1. Dung Beetle Ball Rolling

In the DBO, the dung beetles need to move in a specific direction within an optimal range. The rolling dung beetle relies on the Sun for navigation, but the brightness of the sunlight affects the path determined by the dung beetle. During scrolling, the mathematical model used to update the position of the dung beetle is expressed as follows:

$$x_i(t+1) = x_i(t) + \alpha \times k \times x_i(t-1) + b \times \Delta x \quad (7)$$

$$\Delta x = |x_i(t) - X^w|$$

where t indicates the count of the current iterations; $x_i^{(t)}$ denotes in terms of the position of the i th dung beetle in the population at the t th permutation; k refers to the deflection coefficient, and $k \in (0, 0.2]$; b is a constant, and $b \in (0, 1)$; α represents the extent of the deviation from the value of either -1 or 1 , with 1 indicating no deviation from the original direction and -1 indicating otherwise; and X^w means the worst global position in the present species. Δx is used to simulate the changes in the intensity of the sunlight.

When the individual is unable to continue moving, it repositions itself by dancing to find a new route. To simulate the dancing behavior, the DBO algorithm adopts the tangent function to determine a new direction of scrolling. At this point, the mathematical model used to update the position of the dung beetle is expressed as follows:

$$x_i(t+1) = x_i(t) + \tan(\theta)|x_i(t) - x_i(t-1)| \quad (8)$$

where θ represents the deflection angle, and $\theta \in [0, \pi]$. When $\theta = 0, \pi/2$, and π , no change occurs in where the dung beetles are positioned.

2.2.2. Dung Beetle Breeding

In nature, dung balls are rolled to a safe location and hidden by dung beetles. To reproduce more safely, dung beetles need to select a suitable oviposition site. The DBO algorithm is used to model the oviposition area for dung beetles, which is defined as follows:

$$\begin{aligned} Lb^* &= \max(X^* \times (1 - R), Lb) \\ Ub^* &= \min(X^* \times (1 - R), Ub) \end{aligned} \quad (9)$$

where X^* represents the current best position; Lb^* indicates the lower limit of the spawning area; and Ub^* denotes the upper limit of the spawning area. The lower and upper limits of the optimization problem are represented by Lb and Ub , respectively. $R = 1 - t/T_{max}$, where T_{max} refers to the maximum number of iterations.

Once the oviposition area is determined, the dung beetles lay eggs within this region. Each female dung beetle lays only one egg per cycle. In each iteration cycle, the position of the laid eggs changes constantly, and their mathematical model is expressed as follows:

$$B_i(t+1) = X^* + b_1 \times (B_i(t) - Lb^*) + b_2 \times (B_i(t) - Ub^*) \quad (10)$$

where $B_i(t+1)$ represents the location of the i th brood ball at the t th iteration; b_1 and b_2 represent two independent random vectors that contain D components each; and D is referred to as the dimension parameters of the optimization problem.

2.2.3. Dung Beetle Foraging

Adult dung beetles emerge from the ground to forage. The mathematical models used to determine the boundary of the optimal foraging area and to update the individual position are expressed as follows:

$$\begin{aligned} Lb^b &= \max(X^b \times (1 - R), Lb) \\ Ub^b &= \min(X^b \times (1 - R), Ub) \end{aligned} \quad (11)$$

where X^b represents the local best positions of the previous population. The other parameters are defined in Equation (9).

$$x_i(t+1) = x_i(t) + C_1 \times (x_i(t) - Lb^b) + C_2 \times (x_i(t) - Ub^b) \quad (12)$$

where C_1 represents a random number that conforms to normal distribution, and C_2 refers to a random vector falling into the range of (0, 1).

2.2.4. Dung Beetle Stealing

There is a type of dung beetle known as the thief beetle, as it steals dung balls from other beetles. According to Equation (11), X^b is the optimal source of dung. Therefore, it is assumed that the area surrounding X^b is the best location to compete for food. The

mathematical model used to update the position of the thief beetles during the iteration process is expressed as follows:

$$x_i(t+1) = x^b + S \times g \times \left(|x_i(t) - X^*| + |x_i(t) - X^b| \right) \quad (13)$$

where g represents a vector of dimension D that is randomly chosen, conforming to normal distribution, and S is a constant.

3. Proposed Method

3.1. Adaptive Spiral Dung Beetle Optimization

Currently, the DBO algorithm has been applied to address various issues related to engineering design, and its effectiveness and feasibility have been validated. However, the algorithm still has some limitations, including poor global search capability and a tendency to prematurely converge to local optima. To address these drawbacks, this paper proposes an enhanced Dung Beetle Optimization strategy.

3.1.1. Path Diversity

The reproduction of dung beetles is updated in real time with the position of the scrolling ball. Due to the continuous update on the position of the reproductive dung beetles, the foraging area of the dung beetles changes as well, thus leading to a single mode of the behavior position update. Inspired by the rotational operation of the Whale Optimization Algorithm [31,32], the spiral search strategy is introduced, which allows the dung beetles to conduct a search in a spiral form in space. This expands the reproductive and foraging areas, enhances the individual ability to explore unknown regions, and provides a more flexible path for the position update. Therefore, the capability of the local search is improved for both behaviors. The reproductive and foraging formulas updated after the introduction of the spiral position-updating strategy are presented as Equations (14) and (15), respectively.

$$\begin{cases} B_i(t+1) = X^* + e^{zl} \cdot \cos(2\pi l) \cdot b_1 \cdot (B_i(t) - Lb^*) + e^{zl} \cdot \cos(2\pi l) \cdot b_2 \cdot (B_i(t) - Ub^*) \\ z = e^{k \cos(\frac{\pi t}{\max t})} \end{cases} \quad (14)$$

$$\begin{cases} x_i(t+1) = e^{zl} \cdot \cos(2\pi l) \cdot x_i(t) + C_1 \times (x_i(t) - Lb^b) + C_2 \times (x_i(t) - Ub^b) \\ z = e^{k \cos(\frac{\pi t}{\max t})} \end{cases} \quad (15)$$

where the parameter z , which varies with the number of iterations, is composed of an exponential function based on e . The coefficient of variation, denoted as k , is set to $k = 5$, which ensures that the algorithm operates within an appropriate range. Also, l is a uniformly distributed random number in the range of $[-1, 1]$.

As a crucial parameter used in this strategy, the spiral parameter z is not supposed to be a fixed value because this would lead to the monotonous methods of the reproductive and foraging position update for the dung beetles, thus trapping the strategy in local optima. Therefore, z is a dynamic variable used to adjust the size and amplitude of the spiral curve, which enhances the capabilities of exploration for both behaviors in unknown regions. As dung beetles engage in reproductive and foraging activities with positions varying in a spiral pattern, their search area expands, resulting in the identification of more high-quality solutions. With each iteration, progressively superior values are acquired, thereby enhancing the algorithm's search capabilities to some extent.

3.1.2. Dynamic Update on Positions

As can be seen clearly from Equation (13), X^b and its vicinity represent the best location to compete for food. Concerning the thief dung beetles, they update their positions continuously to find the optimal location to compete with others for food. However, when there are individuals conducting a search through the method shown in Equation (13),

they tend to linger around the local optima, which causes the failure in achieving a better performance in local optimization. Therefore, the introduction of the weight improves the individual position, enabling the convergence toward those better search areas.

$$x_i(t+1) = X^b + S \times g \times \left(|x_i(t) - X^*| + |x_i(t) - \omega X^b| \right) \quad (16)$$

$$\omega = \frac{e^{2 \times (1 - \frac{t}{\max t})} - e^{-2 \times (1 - \frac{t}{\max t})}}{e^{2 \times (1 - \frac{t}{\max t})} + e^{-2 \times (1 - \frac{t}{\max t})}}$$

where S is a constant; t denotes the count of the current iterations; and g indicates a random vector that conforms to normal distribution.

Introducing a real-time weight adjustment for the behavior and position of the thief dung beetles enables the algorithm to operate in different modes, enhancing the search flexibility. With an increasing number of iterations, the thief individuals continuously update their positions toward the optimal location, competing for food. Consequently, the algorithm experiences an improved convergence speed.

Although the introduction of dynamic weights to the thief dung beetle individuals is effective in improving the performance in convergence, it is still possible for the algorithm to get trapped in local optima when facing high-dimensional problems. Levy flight refers to a random walk with a probability distribution of step lengths with heavy tails, which means there is a relatively high probability that large steps are taken during the random walk process [33].

For this reason, another update of individual positions is performed using the Levy flight after adaptive weights are introduced into the formula for the thief dung beetles. This improvement is advantageous in allowing for both small and occasional large step movements, which prevents the dung beetle individuals from taking hold of one position repeatedly. This enhances the randomness of the solutions found by the algorithm and enables it to get out of local optima.

The position-updating formula used to introduce the Levy flight is expressed as follows:

$$x_i(t+1) = \text{levy}(\lambda) \cdot X^b + S \times g \times \left(|x_i(t) - X^*| + |x_i(t) - \omega X^b| \right) \quad (17)$$

where $\text{levy}(\lambda)$ represents the random step length and conforms to the Levy distribution, which satisfies the conditions that $\text{Levy} \sim u = t^{-\lambda}$ and $1 < \lambda \leq 3$.

Given the intricacies associated with the Levy distribution, the Mantegna algorithm is frequently employed for simulation purposes [34]. The formula used to calculate the step size is expressed as follows:

$$S = \frac{u}{|v|^{1/\beta}} \quad (18)$$

where u and v conform to normal distribution, $u \sim N(0, \sigma_u^2)$ and $v \sim N(0, \sigma_v^2)$.

$$\sigma_u = \left\{ \frac{\Gamma(1 + \beta) \sin\left(\frac{\beta\pi}{2}\right)}{\Gamma\left(\frac{1+\beta}{2}\right) \beta \times 2^{\frac{\beta-1}{2}}} \right\}^{\frac{1}{\beta}} \quad (19)$$

$$\sigma_v = 1$$

where β usually takes the value in the range of $[0, 2]$ and it is set to 1.5 in this context.

With the incorporation of the Levy flight mechanism, dung beetles exhibit increased flexibility in this stage, enabling them to guide other individuals toward better positions. Consequently, the amalgamation of the Levy flight mechanism and weight strategy brings balance to the algorithm, resulting in a noteworthy enhancement in each solution.

3.1.3. Performance Analysis of the Algorithm

To verify the optimization performance of the ASDBO algorithm, 8 widely used benchmark functions are selected in this study for the evaluation of performance, as shown in Table 1. Among them, $f_1 \sim f_5$ are unimodal functions, the focus of which is on testing the convergence speed and optimization accuracy of the algorithm. $f_6 \sim f_8$ are multimodal functions, the focus of which is on reflecting the ability of the algorithm to get rid of local optima. The performance was evaluated on the 8 benchmark functions listed in the table for dimensions $d = 30$, $d = 50$, and $d = 100$. A comparison was performed against the standard DBO algorithm. Table 2 shows the comparison of the optimization performance for each algorithm, and Figure 2 shows the convergence of the functions of each algorithm. To ensure the objectivity in the experiments, all the tests were conducted independently, with the total number of iterations T set to 500 and the population size N set to 30. To enhance the reliability of the experimental results and remove the impact of chance events, the minimum value, average value, and standard deviation of each algorithm were statistically analyzed. These metrics reflect the optimization performance and stability of each algorithm. The average value represents the accuracy of the optimization, while the standard deviation reflects the robustness of the algorithm.

The improvement in the convergence speed and accuracy of the ASDBO algorithm compared to the DBO algorithm is evident from Figure 2. The global optimum refers to the optimal value attained by the objective function across the entire domain. For the functions $f_1 \sim f_4$, ASDBO demonstrates optimal optimization performance, consistently achieving stable global optimal solutions. In the case of f_6 and f_7 , it consistently identifies the optimal values in each test, while for f_5 , it converges to an optimization accuracy close to 0. Additionally, as shown in Table 2, for the standard deviation test results, except for f_4 , f_5 , and f_3 in the case of dimension 100, the standard deviation of the ASDBO algorithm is 0, indicating its strong robustness. The experimental results confirm that the ASDBO algorithm outperforms the DBO algorithm in the success rate of optimization, whether in low-dimensional or high-dimensional scenarios. The graphical representation also reveals that, for some test functions, ASDBO converges to the global optimum earlier in the iteration process, while the standard DBO algorithm exhibits stagnation in the fitness curve at later stages, indicating difficulty in escaping local optima. Therefore, the ASDBO algorithm significantly excels in the overall convergence speed compared to the standard DBO algorithm. The introduction of multiple strategies allows the algorithm to break through limited search patterns during the optimization process, enabling a more flexible and meticulous search. Consequently, the convergence performance of the algorithm is enhanced.

Table 1. Benchmark functions.

	Function Expression	Interval	Min
Sphere	$f_1(x) = \sum_{i=1}^n x_i^2$	$[-100, 100]$	0
Schwefel2.22	$f_2(x) = \sum_{i=1}^n x_i + \prod_{i=1}^d x_i $	$[-10, 10]$	0
Schwefel1.2	$f_3(x) = \sum_{i=1}^d \left(\sum_{j=1}^i x_j \right)^2$	$[-100, 100]$	0
Schwefel2.1	$f_4(x) = \max_i \{ x_i , 1 \leq i \leq d\}$	$[-100, 100]$	0
Quartic	$f_5(x) = \sum_{i=1}^d ix_i^4 + \text{random}[0, 1)$	$[-1.28, 1.28]$	0
Rastrigin	$f_6(x) = \sum_{i=1}^d [x_i^2 - 10 \cos(2\pi x_i) + 10]$	$[-5.18, 5.12]$	0
Ackley	$f_7(x) = -20 \exp \left(-0.2 \sqrt{\frac{1}{d} \sum_{i=1}^d x_i^2} \right) - \exp \left(\frac{1}{d} \sqrt{\sum_{i=1}^d \cos(2\pi x_i)} \right) + e + 20$	$[-32, 32]$	0
Griewank	$f_8(x) = \frac{1}{4000} \sum_{i=1}^d x_i^2 - \prod_{i=1}^d \cos \left(\frac{x_i}{\sqrt{i}} \right) + 1$	$[-600, 600]$	0

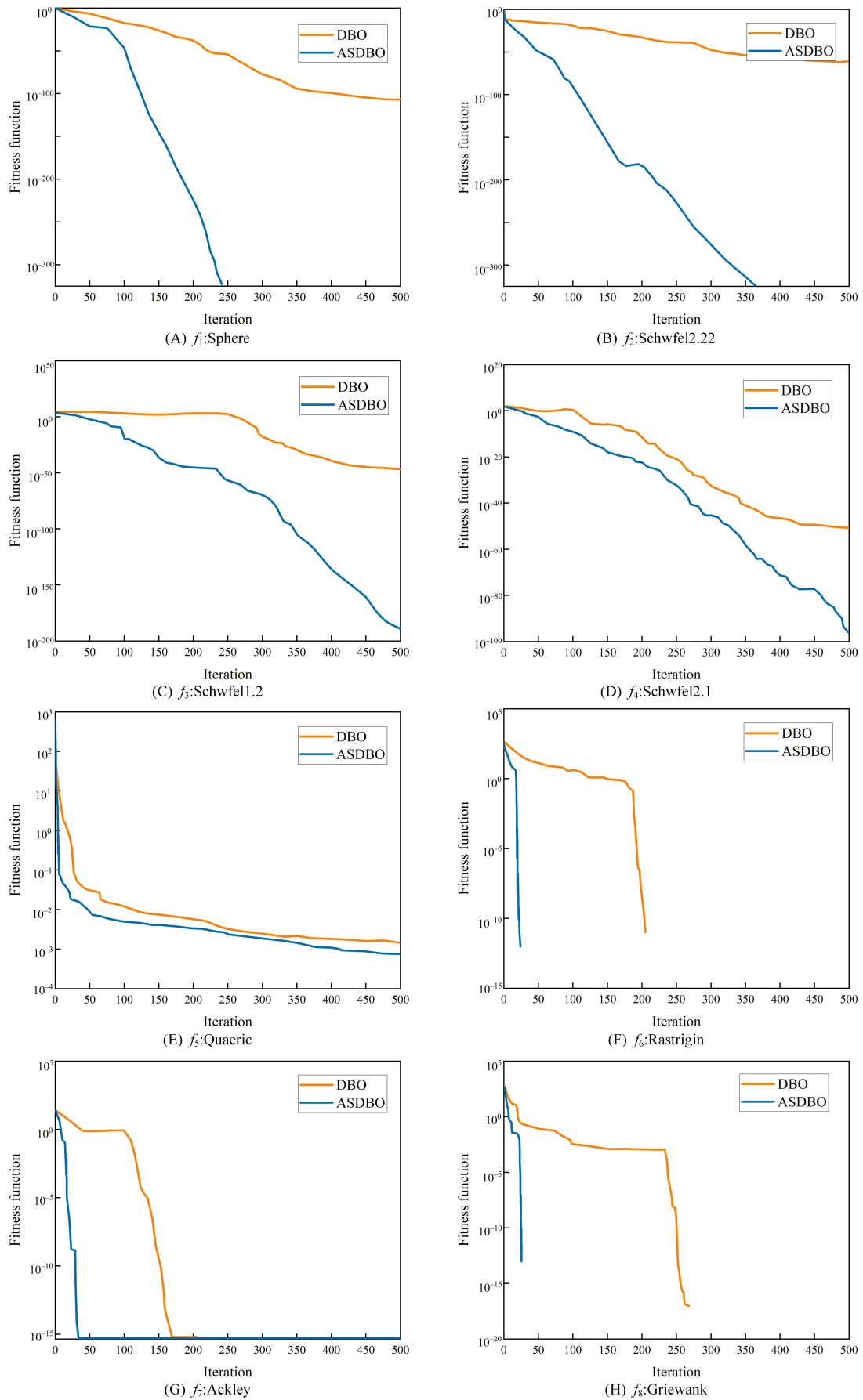


Figure 2. Comparison of the convergence performance of the test functions of the two algorithms.

3.2. Establishment of the ASDBO-ELM Prediction Model

Improving the predictive accuracy of short-term PV power forecasting models is crucial for achieving economic dispatch and advancing the development of clean energy generation technologies. However, inappropriate values set for the input weights and hidden thresholds in the ELM can lead to a decline in predictive performance. These randomly generated initial values directly impact the final prediction results. Therefore, this study employs the ASDBO algorithm to optimize the parameters of the ELM model. During the optimization process, each dung beetle's position represents a set of ELM parameters, and the input weights and hidden thresholds are optimized through 5-fold cross-validation and the ASDBO algorithm. Simultaneously, the fitness function is defined as the Mean Square Error (MSE) obtained from 5-fold cross-validation on the training set:

$$e_{MSE} = \frac{1}{N} \sum_{i=1}^N (y_a(i) - y_b(i))^2 \quad (20)$$

where $y_a(i)$ represents the real value; $y_b(i)$ represents predicted value.

The Process Structure Chart is shown in Figure 3.

- The original PV dataset is partitioned into training and testing sets, followed by a correlation analysis on the samples to identify the input feature vectors.
- The parameters of the ASDBO algorithm are initialized, including the number of dung beetles, the maximum number of iterations, and the individual positions of the dung beetles.
- Based on the calculated objective function values for each individual dung beetle in the population, we obtain the global optimum position X^{best} corresponding to the minimum objective function value and the worst position X^{worst} corresponding to the maximum objective function value.
- If the individual belongs to the rolling dung beetle category, the next behavior of this dung beetle is determined, whether it continues rolling or switches to dancing, through a probabilistic method. This process is then used to ascertain the current local optimal position.
- The positions of the other three sub-populations of dung beetles are updated. If an individual belongs to the breeding beetles category, its position is updated according to Equation (14). If the individual belongs to the foraging beetles category, its position is updated according to Equation (15). If the individual belongs to the stealing beetles category, its position is updated according to Equation (16).
- Based on the positional parameters of each dung beetle, the fitness of each individual is calculated by performing 5-fold cross-validation on the training sample set. The fitness value is determined by calculating the Mean Square Error (MSE) of the prediction results using Formula (20).
- It is determined if the iteration termination condition is met and output the optimal model value. Subsequently, we retrain the model with the optimized parameters, predict on the test dataset, and analyze and evaluate the results.

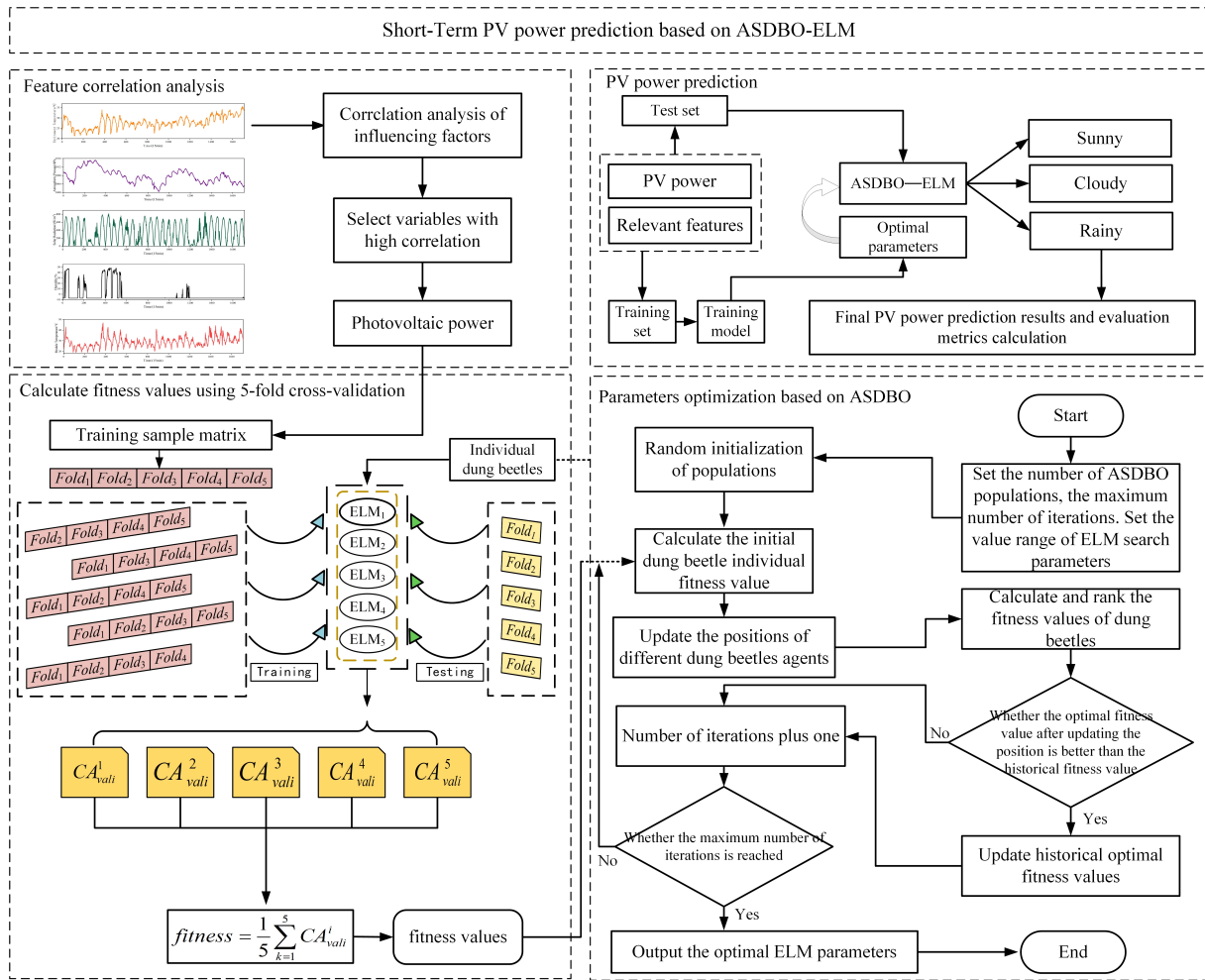


Figure 3. Process Structure Chart.

4. Influential Dynamic Factors on PV Power

4.1. Analysis of the Impact of Different Weather

The output of PV power is greatly influenced by weather conditions, with varying degrees of impact on PV power generation under different weather conditions. Therefore, PV power output exhibits significant fluctuations. In this paper, the research data are based on the existing data from a PV power station in a specific region of Jiangsu, along with the corresponding meteorological elements (i.e., the solar radiation, atmospheric pressure, environment temperature, relative humidity, and PV module temperature). The data points were sampled at 15 min intervals to accurately capture the operational variations. Given the absence of power output at night, the analysis centers on the time period from 05:00 to 19:00, considering the daylight hours at the PV power station. As an example, the data from May 2017 were selected, including the clear sky power data on 7 May, overcast sky power data on 5 May, and rainy sky power data on 12 May, for further analysis.

As shown in Figure 4, the power output curve of the sunny spell shows relatively insignificant fluctuations, while the fluctuations become more significant during cloudy and rainy spells. During the sunny spell, which lasts from 5:00 to 11:00, the PV power output increases with the rise in the sunlight intensity. At around 12:00, the PV power generation peaks. In the hours after 13:00, the sunlight intensity decreases gradually, as does the output of PV power.

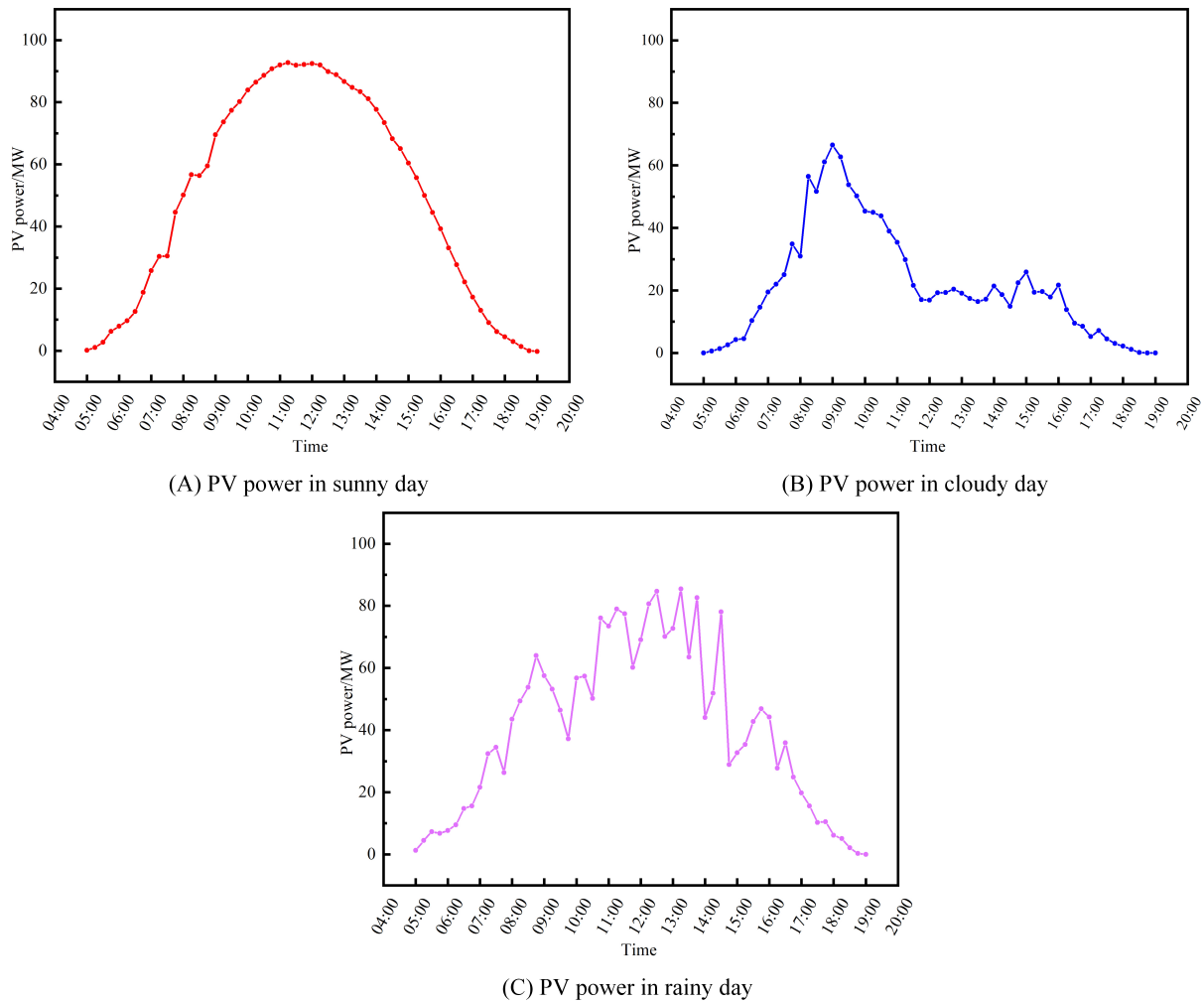


Figure 4. Photovoltaic power curve.

4.2. Analysis of the Impact of Different Meteorological Factors

To a significant extent, PV power generation is influenced by various meteorological factors that are inherently uncertain and beyond our control. Factors such as solar irradiance, atmospheric temperature, module temperature, wind pressure, humidity, and more contribute to the dynamic nature of the power output in PV systems. In this experiment, the discussions and analyses center on the solar irradiance, PV module temperature, ambient temperature, atmospheric pressure, relative humidity, and power sequence data.

In the realm of PV power prediction, the complexity arises in selecting prediction features due to the myriad factors affecting PV power and the intricate relationships among them. Therefore, a thorough analysis of the factors influencing PV power becomes crucial before venturing into predictions. In this experiment, the Pearson correlation coefficient method is utilized to scrutinize the correlation between the output power of the selected data and various meteorological factors. A correlation analysis is applicable to identify the input variables closely related to PV power, thus avoiding the interference from irrelevant variables used in the power prediction model and reducing computational complexity. At the same time, the presence of a correlation lays a foundation for the introduction of relevant variables as input into the model. The calculation of the Pearson correlation coefficient is shown in Formula (21).

$$\rho_{x,y} = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^n (y_i - \bar{y})^2}} \quad (21)$$

where $|\rho| \leq 1$ represents the correlation coefficient between variables; x_i and y_i represent the values of two factors for the i -th data point; and \bar{x} and \bar{y} represent the average value of variables x and y , respectively.

If the correlation coefficient ρ is positive, it suggests a positive correlation present between the variables. On the contrary, a negative correlation is indicated when the correlation coefficient is negative. A correlation coefficient of 0 signifies the absence of a relationship between the variables. The strength of the correlation is typically assessed based on the range of the correlation coefficient, as illustrated in Table 3.

Table 3. Relevant degree judgment.

ρ	Degree of Association
0.8~1.0	Exceedingly high association
0.6~0.8	More association
0.4~0.6	Moderate association
0.2~0.4	Less association
0.0~0.2	Exceedingly less or no association

Figure 5 shows a calculation of the correlation coefficients between the power and environmental temperature, solar irradiance, air pressure, PV module temperature, and relative humidity. Among these variables, solar irradiance shows the closest correlation with power, reaching 0.75. According to Table 3, the correlation between these two attributes reaches a clearly significant extent. The correlation coefficient is 0.53 and 0.29 for the environmental temperature and module temperature, respectively, indicating a moderate correlation and a weak correlation, respectively. The relative humidity and air pressure exhibit very weak and even negative correlations. The analysis indicates that the solar irradiance, environmental temperature, and PV module temperature significantly impact PV power, whereas the relative humidity and air pressure exert a minor influence on the power output.

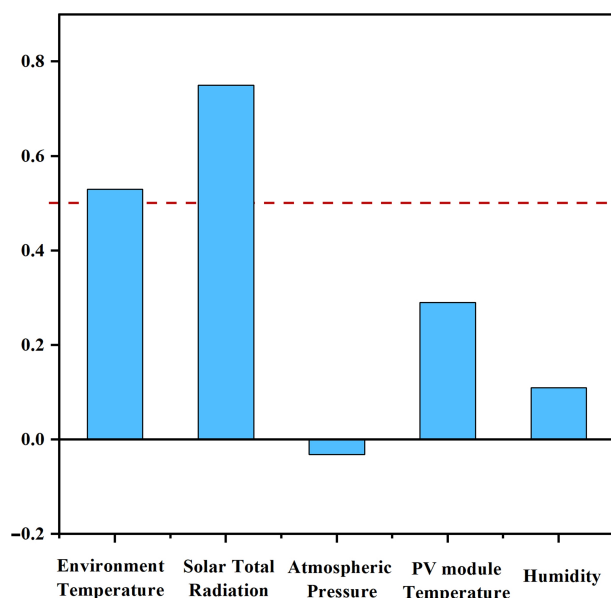


Figure 5. Pearson correlation coefficient graph.

In this chapter, the weather and meteorological factors influencing PV power are investigated by analyzing the patterns of variation in the PV curves under different weather conditions. The solar irradiance, PV module temperature, environmental temperature, and historical PV power are utilized as the input feature vectors. The relationship curves between these variables are shown in Figure 6.

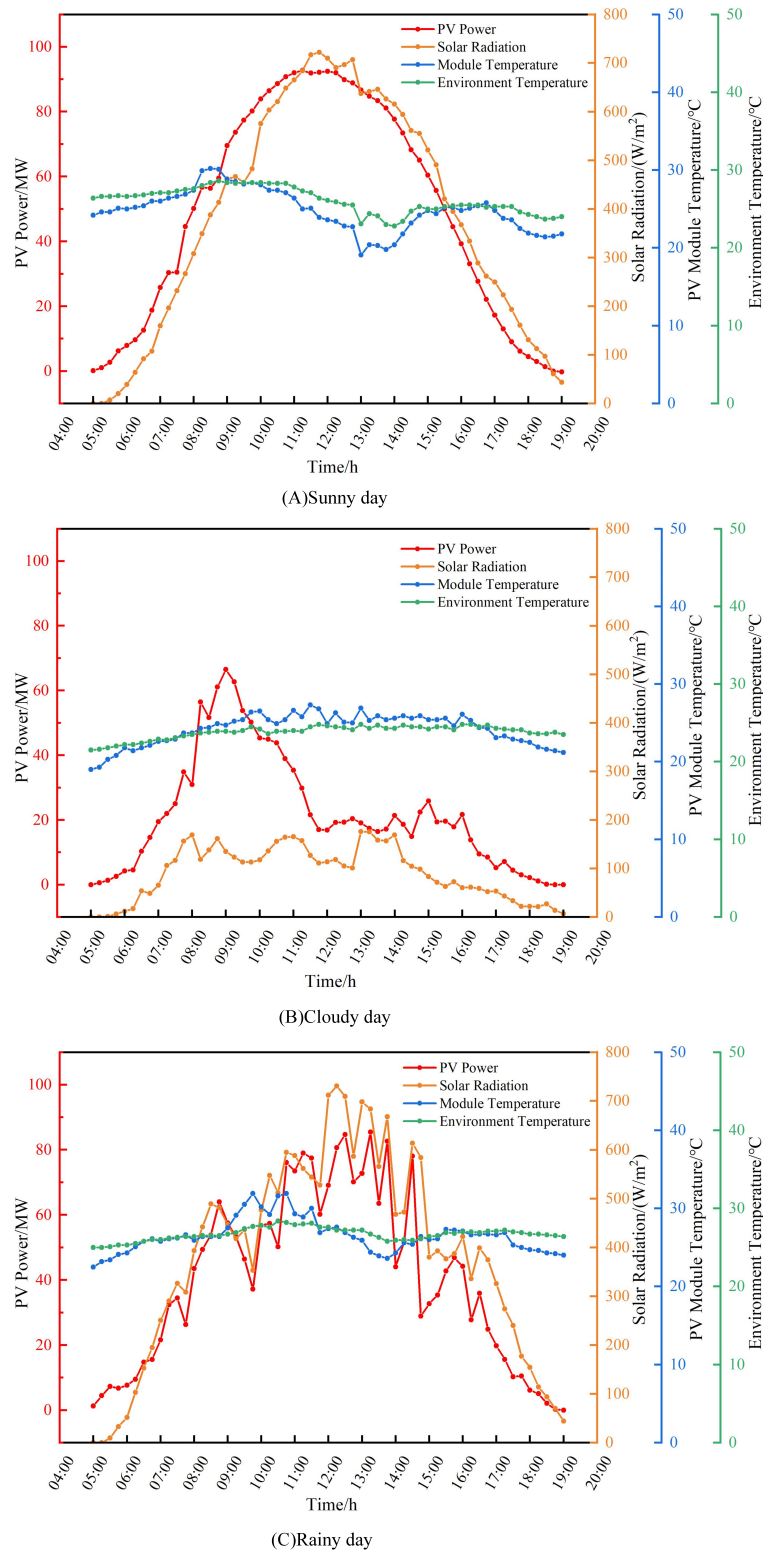


Figure 6. Relationship curve.

5. Case Study

5.1. Data Analysis and Evaluation Metrics

The output power of the PV system varies with different weather conditions and factors that affect the climate. As shown in Figure 4, the PV power curve is stable with slight fluctuations on sunny days, while the power curve fluctuates more significantly on cloudy and rainy days. To validate the predictive performance of the proposed forecasting

model, the PV power generation is predicted under three different conditions: sunny, cloudy, and rainy spells. Take the 2017 data as an example. The sunny day power data are sourced from the samples spanning from 15 May to 30 May, the cloudy day power data are sourced from the samples spanning from 1 June to 15 June, and the rainy day power data are sourced from the samples spanning from June 16th to June 30th. Here, 80% of the samples are treated as the training set, and the remaining 20% are treated as the test set. The number of hidden neurons is set to 50. For the optimization of the hidden layer weights and biases, the ASDBO algorithm is applied. The algorithm is set to have a dung beetle population of 25 and a maximum iteration count of 100. The fitness of each individual is calculated through cross-validation on the training sample set.

In order to further enhance the predictive performance of our model, this study employed cross-validation to fine-tune the model's hyperparameters and strengthen its robustness. Cross-validation involves dividing the data into multiple training and validation sets, effectively alleviating the potential validation bias arising from data randomness and limited training samples. Specifically, this study adopted a 5-fold cross-validation technique. This method randomly divided the original data into approximately five subsets of roughly equal size. In each iteration, four of these subsets were used for training, while the remaining one served as the validation set. This process was repeated five times, resulting in five independent cycles of model training and validation. The final cross-validation error was computed as the average of the validation results from these iterations. Through this approach, this research not only addressed concerns related to overfitting and underfitting but also fine-tuned the model's hyperparameters to achieve a higher level of predictive accuracy.

Before making predictions, utilizing feature data with original dimensional scales in calculations may introduce adverse impacts resulting from disparate dimensions. To alleviate the influence of varying data scales on prediction outcomes due to multiple factors, it is crucial to normalize both PV data and influencing factor data. Common normalization methods encompass the Min-Max, Mean, and Z-score. The Z-score method, a centering approach, is suitable for situations where the data distribution approximates a normal distribution. However, PV data may manifest varying distributions under different weather conditions, deviating from a strict adherence to a normal distribution. The effectiveness of mean normalization is contingent upon the data distribution, and in instances of significant skewness, the mean may not precisely represent the central tendency, thereby influencing the normalization outcome. Additionally, mean normalization does not rectify the scale differences among distinct features. In scenarios with substantial variations in feature scales, mean normalization may inadequately scale the data.

For PV data, Min-Max normalization proves particularly advantageous when the maximum and minimum values are explicitly defined. This method proficiently scales the data to the [0, 1] range. This strategy contributes to preserving inherent weight relationships, as reflected by the standard deviation. Consequently, it helps mitigate the adverse effects of scale differences among diverse features during model training, as delineated in Equation (22). Finally, denormalization is applied to the predicted output power values to restore their physical significance.

$$y'_i = \frac{y_i - y_{min}}{y_{max} - y_{min}} \quad (22)$$

where y_i represents the original sample data; y'_i indicates the normalized data; y_{max} represents the maximum value in the data; and y_{min} denotes the minimum value in the data.

In order to evaluate the predictive accuracy of the proposed model in this paper against other models, the Mean Absolute Percentage Error (MAPE), Root Mean Square Error (RMSE), and Mean Absolute Error (MAE) are used to evaluate the predictive performance of the models. The coefficient of determination R-squared (R^2) is also used to evaluate the goodness of fit.

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_a(i) - y_b(i)| \quad (23)$$

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{y_a(i) - y_b(i)}{y_a(i)} \right| \times 100\% \quad (24)$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n [y_a(i) - y_b(i)]^2} \quad (25)$$

$$R^2 = 1 - \frac{\sum_{i=1}^n [y_a(i) - y_b(i)]^2}{\sum_{i=1}^n [y_a(i) - \bar{y}(i)]^2} \quad (26)$$

5.2. Model Performance Evaluation

To validate the performance of the proposed predictive model, Figure 7 illustrates the forecast results for different typical days. The graph depicts the relationship between the actual measurement values and those obtained through testing with the ASDBO-ELM model. As shown in Figure 7, the predicted PV power by the proposed ASDBO-ELM model closely aligns with the actual PV power output curve. Additionally, despite significant fluctuations in the actual PV power curve during the overcast and rainy days, the model accurately predicts the PV generation. Based on the correlation distribution of the actual and predicted power in the test samples, the model's predicted PV power matches the actual PV power curve consistently. However, it is noteworthy that, despite some errors, the predicted results achieved by the ASDBO-ELM model closely resemble the actual results in the majority of cases. At the same time, in extremely rare cases, there is a significant discrepancy between the predicted and actual values. This further underscores the robust performance of the model under different weather conditions.

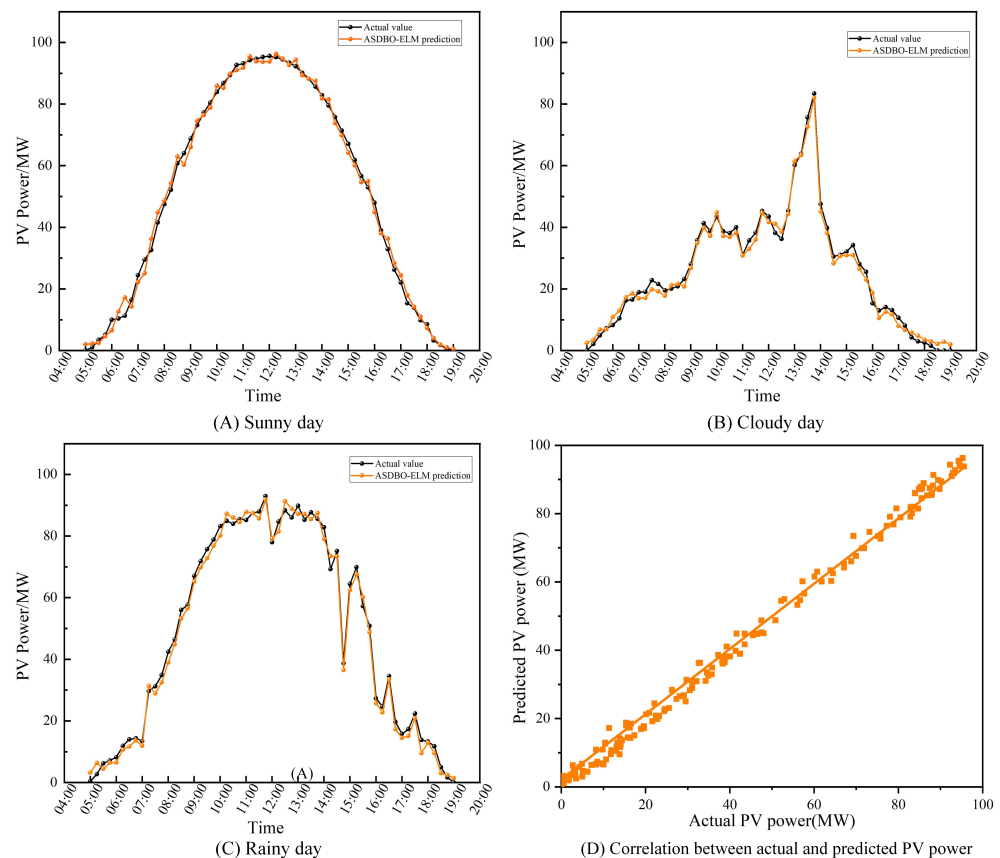


Figure 7. Actual and predicted photovoltaic power generation with correlation.

5.3. Comparative Analysis of Different Prediction Methods

To further validate the exceptional performance and applicability of the ASDBO-ELM prediction model, this paper also compares its forecasting results with those of various weather conditions from the PV power prediction models, including the BP, LSTM, GRU, SVM, ELM, and DBO-ELM.

Figure 8A–C show the PV power generation predictions made on sunny, cloudy, and rainy days, respectively. As shown in Figure 8A, in the PV power prediction curve under sunny conditions, the BP model exhibits significant prediction errors between 08:00 and 14:00, deviating from the actual values toward the end. In contrast, the LSTM, GRU, and ELM models show continuous fluctuations between 6:00–14:00 and 15:00–18:00. Although the DBO-ELM model has some deviations, such as insignificant fluctuations in predictions between 06:00 and 08:00 and 11:00–14:00, inconsistent with actual values, it still demonstrates good predictive performance. In comparison with the other models, the prediction curve of the ASDBO-ELM model is much more consistent with the actual values, showing the most satisfactory predictive performance. Figure 8B illustrates the PV power prediction curve on cloudy days. It can be observed that the BP model exhibits the most significant deviation from the actual values in its prediction curve. The LSTM, GRU, ELM, and DBO-ELM models show substantial fluctuations in different time periods. According to the analysis, the predictive curve of the ASDBO-ELM model continues to well reflect the trend of changes in the actual power values during the cloudy conditions, with the best fitting effect observed especially between 9:00 and 15:00. In cloudy weather, the PV power curve displays considerable randomness, and the predicted values of the model proposed in this paper deviate to some extent from the actual values. However, compared to the other prediction models, the overall prediction error is relatively small and more stable, demonstrating the most significant fitting effect with the actual curve. Figure 8C shows the PV power prediction curve on a rainy day, indicating that the PV power curve exhibits significant fluctuations on rainy days. Overall, the prediction curves of these models capture the fluctuation trends in the actual power curve. The ASDBO-ELM model demonstrates the best predictive performance, effectively mitigating the impact of data fluctuations on accuracy. The analysis indicates that the proposed predictive model exhibits strong adaptability to different weather conditions, showcasing high stability and accuracy.

5.4. Error Analysis and Impact

As shown in Figure 9, a boxplot is constructed using the absolute errors between the predicted results and the actual values as the research object, where the red line represents the median, the blue dots represent the mean, and the red crosses represent the outliers. As can be seen from the boxplot on the sunny day, the ASDBO-ELM model exhibits less significant but more concentrated absolute errors, with fewer outliers. Relative to other models, the upper and lower boundaries of the box are closer, indicating smaller errors and greater stability. Based on the boxplots for the cloudy and rainy days, the prediction model consistently shows shorter box lengths and fewer outliers, attributed to the fluctuations in the PV curves during these weather conditions. In essence, this indicates that absolute errors are confined within a narrower range, highlighting the superior accuracy of the ASDBO-ELM model across diverse weather conditions. In comparison to the alternative models, the ASDBO-ELM model not only exhibits fewer outliers but also attains higher accuracy.

Based on the prediction results, Table 4 presents the evaluation metrics R^2 , MAPE, MAE, and RMSE for the different prediction models. It is noteworthy that the prediction errors of the model increase due to the uncertainty and randomness of the PV power generation curve under cloudy and rainy weather conditions. Compared to sunny weather, the MAE and RMSE obtained under cloudy and rainy weather conditions are relatively larger.

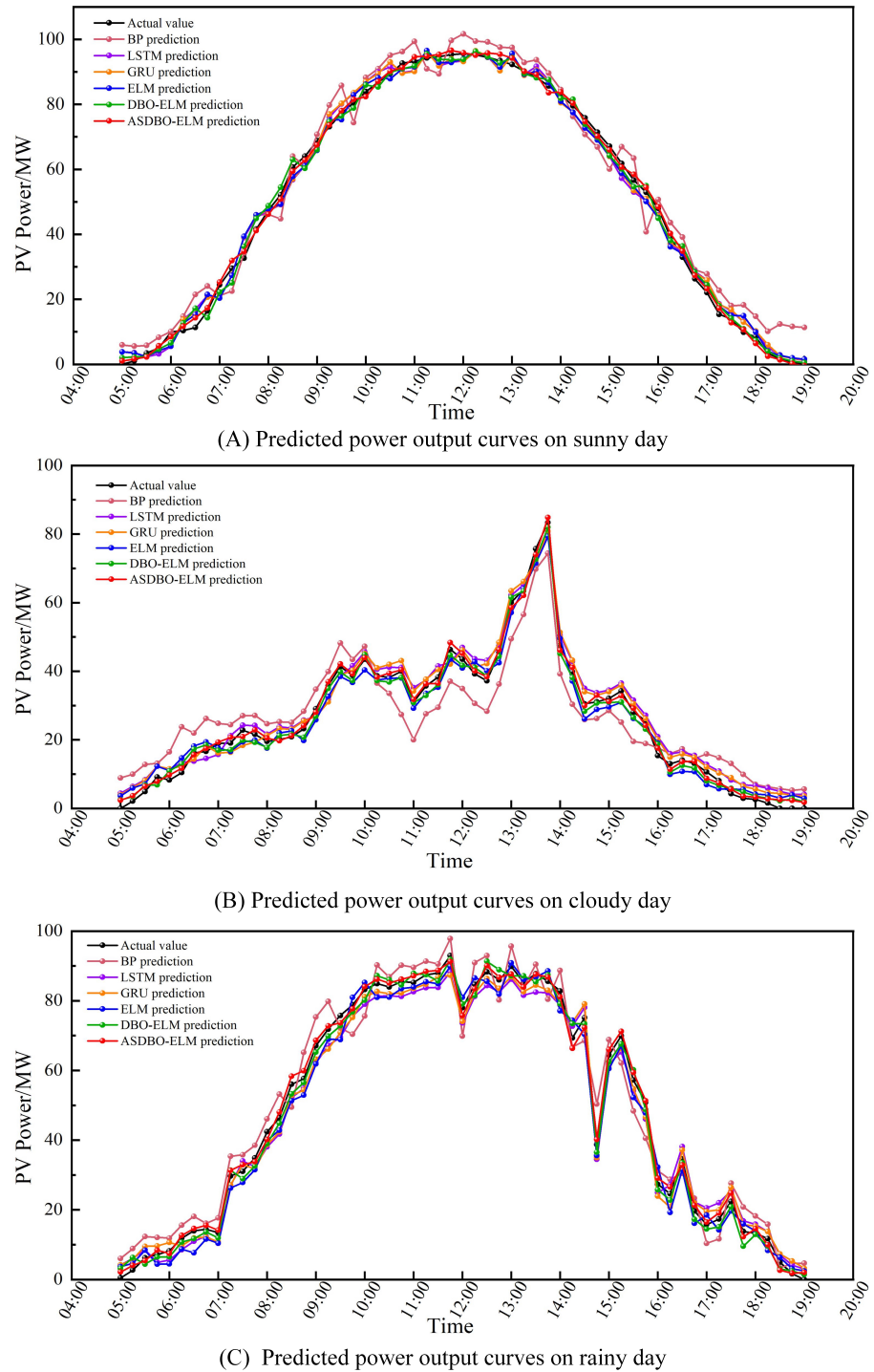


Figure 8. Prediction results of different models.

On sunny days, the ELM model reduces the *MAE* evaluation metric by 43.08%, 27.66%, and 21.33% compared to the BP model, LSTM model, and GRU model, respectively. The ELM model also decreases the *RMSE* evaluation metric by 41.47%, 5.29%, and 2.45% for the respective models. Additionally, in terms of the *MAPE*, the reductions are 27.86%, 5.94%, and 2.06%, and for the R^2 , the improvements are 4.11%, 1.92%, and 1.15%.

On cloudy days, the ELM model reduces the *MAE* evaluation metric by 22.88%, 8.74%, and 4.19% compared to the BP model, LSTM model, and GRU model, respectively. The reductions in the *RMSE* are 26.98%, 10.79%, and 9.36%, while the decreases in the

MAPE are 42.94%, 11.97%, and 2.58%. The improvements in the R^2 are 4.31%, 1.36%, and 1.20%, respectively.

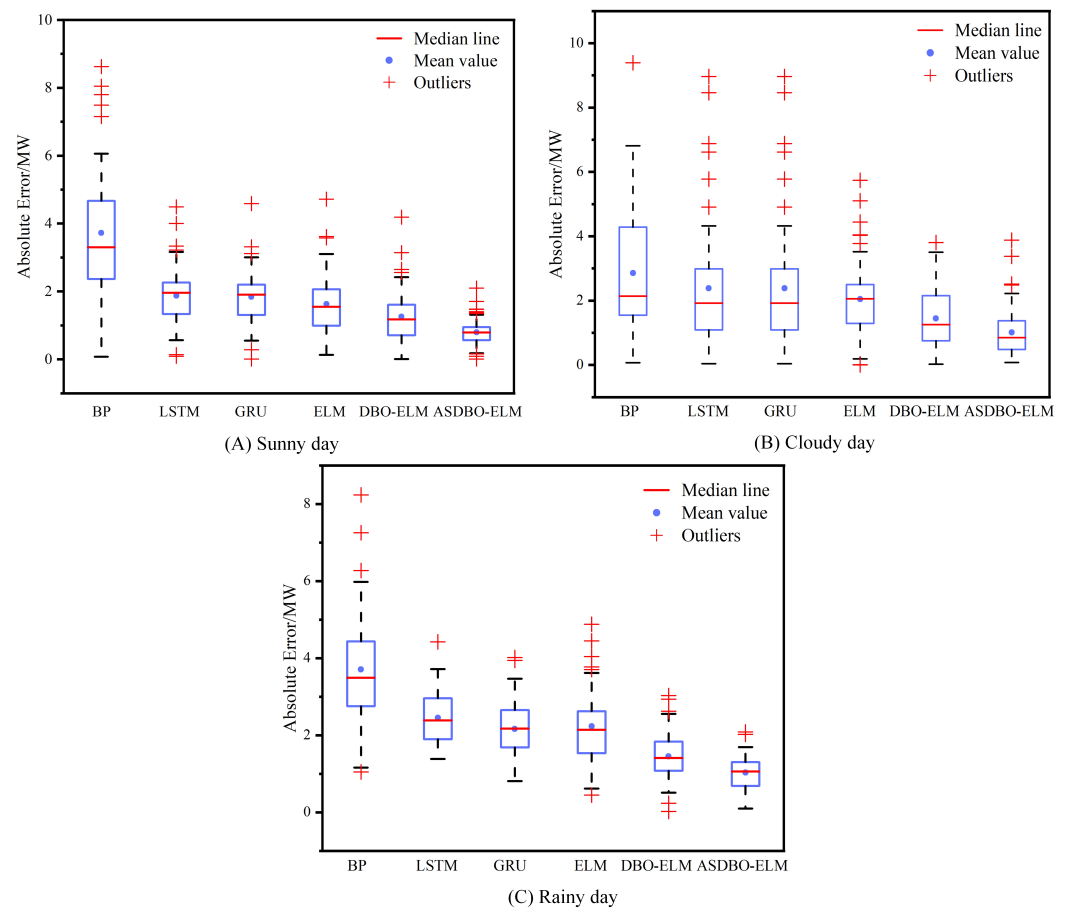


Figure 9. Boxplots of absolute errors of different weather forecasts.

Table 4. Evaluation of the prediction effects of different models.

Weather	Models	Evaluation Index			
		R^2 (%)	MAE	RMSE	MAPE
Sunny	BP	92.37	4.425	6.714	3.351
	LSTM	94.36	3.341	4.534	2.461
	GRU	95.07	3.073	3.431	2.387
	ELM	96.17	2.417	3.347	2.314
	DBO-ELM	98.41	1.741	2.168	2.067
	ASDBO-ELM	98.65	1.543	1.963	1.851
Cloudy	BP	90.87	5.631	7.141	8.653
	LSTM	94.46	4.759	5.863	5.701
	GRU	94.61	4.532	5.546	5.067
	ELM	94.80	4.342	5.394	4.936
	DBO-ELM	95.17	3.963	4.779	4.086
	ASDBO-ELM	97.03	3.615	4.081	3.767
Rainy	BP	91.74	5.741	7.078	9.824
	LSTM	93.56	4.521	5.947	6.671
	GRU	94.61	4.482	5.821	6.831
	ELM	94.76	4.347	6.421	5.627
	DBO-ELM	96.41	3.842	6.041	4.953
	ASDBO-ELM	97.82	3.462	3.446	5.316

During the rainy days, the ELM model reduces the *MAE* evaluation metric by 24.29%, 3.85%, and 2.99% compared to the BP model, LSTM model, and GRU model, respectively. The reductions in the *RMSE* are 21.20%, 6.22%, and 4.20%, and the decreases in the *MAPE* are 32.37%, 3.57%, and 5.97%. The improvements in the R^2 are 3.29%, 1.28%, and 0.16%, respectively.

Despite the further improvement achieved by the ELM model in the accuracy of predictions under different weather conditions, there still exists room for the continued enhancement of precision.

Therefore, the ASDBO-ELM model outperforms the other prediction models through the integration of the ASDBO algorithm with the ELM model and the optimization of the ELM model parameters employing the ASDBO algorithm. In comparison to the alternative models, the *MAE* evaluation metric exhibits a reduction ranging from 11.39% to 63.67% during sunny spells, 7.87% to 35.13% during cloudy spells, and 9.89% to 39.67% during rainy spells. This analysis underscores the ASDBO-ELM model's ability to maintain prediction errors within a narrow range. The *RMSE* evaluation metric shows a decrease from 9.31% to 65.86% during sunny spells, 14.61% to 42.67% during cloudy spells, and 12.40% to 51.34% during rainy spells. The results indicate a high level of prediction stability for the ASDBO-ELM model. The *MAPE* evaluation metric witnesses a decrease of 12.22% to 45.88% during sunny spells, 7.81% to 56.46% during cloudy spells, and 2.40% to 40.18% during rainy spells. This analysis affirms the ASDBO-ELM model's high accuracy in power prediction. The R^2 evaluation metric experiences an increase of 0.24% to 6.80% during sunny spells, 1.74% to 6.56% during cloudy spells, and 1.46% to 6.63% during rainy spells. These findings emphasize the ASDBO-ELM model's exceptional performance in fitting power prediction under three different weather conditions.

In summary, through the analysis of the prediction graphs, error boxplots, and four evaluation metrics, it can be concluded that the ASDBO-ELM model demonstrates exceptionally high accuracy in power prediction under different weather conditions and in the presence of various influencing factors. It can be effectively applied to simulate the trends in photovoltaic generation, ensuring accurate predictions across diverse scenarios.

6. Conclusions

In order to ensure the stable and reliable continuous operation of the power system, accurate forecasting of PV power generation is essential. In this study, an improved Dung Beetle Optimizer was employed to optimize the hyperparameters of the Extreme Learning Machine for precise PV power prediction. The experimental results, as demonstrated in the PV data analysis and model comparison, indicate that the proposed approach significantly enhances the accuracy of PV power prediction. The main contributions of this research are outlined as follows:

- Incorporating different strategies into the DBO algorithm to address its shortcomings, the proposed ASDBO exhibits a superior global search capability compared to traditional DBO.
- Conducting a dynamic analysis of weather and meteorological factors that impact PV power, and selecting highly correlated variables as model inputs through variable correlation analysis, this approach not only reduces the computational costs but also enhances the efficiency of PV prediction under diverse conditions.
- The proposed model showcases formidable predictive capabilities, effectively functioning under diverse weather conditions and environmental scenarios. Through algorithm adjustments, the model ensures adaptability to different environmental factors, maintaining reliable performance across a variety of contexts. The introduction of cross-validation operations further fortifies the model's reliability, confirming its predictive accuracy and enhancing its applicability in real-world environments.
- Accurate short-term photovoltaic predictions are crucial for improving the operational efficiency and management of PV power stations. Serving as guiding tools, these predictions aid decision-makers in identifying optimal power generation resources and

configurations to meet future energy demands. This, in turn, propels the advancement of clean energy.

The accurate prediction of PV power is vital for the grid dispatch department to devise sound plans and ensure a balance between supply and demand. In the future, enhancements to the ELM structure will be explored to further elevate the predictive performance of the ELM model. Moreover, the integration of other emerging deep learning algorithms will be pursued to enhance the predictive capabilities of the hybrid model.

Author Contributions: Y.Z. conceived and completed this paper. T.L. and T.M. supported the writing and the review of this paper. X.S. and D.Y. supported the program coding. Y.Z. supported the building modeling. All the authors contributed to this article and approved the submitted version. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the R&D Program of the Beijing Municipal Education Commission (KM202310015003).

Data Availability Statement: The original contributions presented in this study are included in the article; further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as potential conflicts of interest.

Nomenclature

PV	Photovoltaic
ELM	Extreme Learning Machine
SVM	Support Vector Machine
DBO	Dung Beetle Optimization
ASDBO	Adaptive Spiral Dung Beetle Optimization
w_{ij}	Connection Weight between Hidden Layer and Input Layer
β_{ij}	Connection Weight between Hidden Layer and Output Layer
M	Samples
$G(\bullet)$	Activation Function
X^w	Global Worst Position of DBO
X^b	Global Optimal Position of DBO
X^*	Local Optimal Position of DBO
w	Weight Factor of ASDBO
z	Spiral Parameter of ASDBO
k	Coefficient of Variation of ASDBO
ρ	Correlation Coefficient
MAPE	Mean Absolute Percentage Error
MAE	Mean Absolute Error
RMSE	Root Mean Square Error
R^2	Decision Coefficient

References

- Huang, G.B.; Zhou, H.; Ding, X.; Zhang, R. Extreme learning machine for regression and multiclass classification. *IEEE Trans. Syst. Man Cybern. Part B* **2011**, *42*, 513–529. [[CrossRef](#)] [[PubMed](#)]
- Sens, L.; Neuling, U.; Kaltschmitt, M. Capital expenditure and levelized cost of electricity of photovoltaic plants and wind turbines—Development by 2050. *Renew. Energy* **2022**, *185*, 525–537. [[CrossRef](#)]
- Menezes, E.J.N.; Araújo, A.M.; Da Silva, N.S.B. A review on wind turbine control and its associated methods. *J. Clean. Prod.* **2018**, *174*, 945–953. [[CrossRef](#)]
- Paul, A.; Vaidya, B.P. A Comparison Study of Extrapolation Models and Empirical Relations in Forecasting Solar Wind. *Front. Media SA* **2020**, *7*, 572084.
- Das, U.K.; Tey, K.S.; Seyedmahmoudian, M.; Mekhilef, S.; Idris, M.Y.I.; Van Deventer, W.; Horan, B.; Stojcevski, A. Forecasting of photovoltaic power generation and model optimization: A review. *Renew. Sustain. Energy Rev.* **2018**, *81*, 912–928. [[CrossRef](#)]
- Zazoum, B. Solar photovoltaic power prediction using different machine learning methods. *Energy Rep.* **2022**, *8*, 19–25. [[CrossRef](#)]
- Chai, M.K.; Xia, F.; Zhang, H.; Lu, J.F.; Cui, C.G.; Ma, B. Ultra-short-term prediction of self-identifying photovoltaic based on sky cloud chart. *Power Syst. Technol.* **2021**, *45*, 1023–1031.

8. Wang, K.; Qi, X.; Liu, H. A comparison of day-ahead photovoltaic power forecasting models based on deep learning neural network. *Appl. Energy* **2019**, *251*, 113315. [[CrossRef](#)]
9. Mandal, P.; Madhira, S.T.S.; Haque, A.U.; Meng, J.; Pineda, R.L. Forecasting power output of solar photovoltaic system using wavelet transform and artificial intelligence techniques. *Procedia Comput. Sci.* **2012**, *12*, 332–337. [[CrossRef](#)]
10. Monteiro, C.; Fernandez-Jimenez, L.A.; Ramirez-Rosado, I.J.; Muñoz-Jimenez, A.; Lara-Santillan, P.M. Short-term forecasting models for photovoltaic plants: Analytical versus soft-computing techniques. *Math. Probl. Eng.* **2013**, *2013*, 767284. [[CrossRef](#)]
11. Feng, J.; Wang, Z.; Zhang, P. Forecasting power generation of solar photovoltaic system based on the combination of grey model and weighted Markov chain. *Power Syst. Prot. Control* **2019**, *47*, 55–60.
12. Ding, M.; Liu, Z.; Bi, R.; Zhu, W. Photovoltaic output prediction based on grey system correction-wavelet neural network. *Power Syst. Technol.* **2015**, *39*, 2438–2443.
13. Kudo, M.; Takeuchi, A.; Nozaki, Y.; Endo, H.; Sumita, J. Forecasting electric power generation in a photovoltaic power system for an energy network. *Electr. Eng. Jpn.* **2009**, *167*, 16–23. [[CrossRef](#)]
14. Sperati, S.; Alessandrini, S.; Delle Monache, L. An application of the ECMWF Ensemble Prediction System for short-term solar power forecasting. *Sol. Energy* **2016**, *133*, 437–450. [[CrossRef](#)]
15. González-Castaño, C.; Marulanda, J.; Restrepo, C.; Kouro, S.; Alzate, A.; Rodriguez, J. Hardware-in-the-loop to test an MPPT technique of solar photovoltaic system: A support vector machine approach. *Sustainability* **2021**, *13*, 3000. [[CrossRef](#)]
16. Snieder, E.; Shakir, R.; Khan, U.T. A comprehensive comparison of four input variable selection methods for artificial neural network flow forecasting models. *J. Hydrol.* **2020**, *583*, 124299. [[CrossRef](#)]
17. Han, Y.; Wang, N.; Ma, M.; Zhou, H.; Dai, S.; Zhu, H. A PV power interval forecasting based on seasonal model and nonparametric estimation algorithm. *Sol. Energy* **2019**, *184*, 515–526. [[CrossRef](#)]
18. Xu, M. Short-Term Photovoltaic Power Prediction Based on MA-SVM Method. *Electr. Mach. Control Appl.* **2022**, *49*, 104–111.
19. Mellit, A.; Sağlam, S.; Kalogirou, S.A. Artificial neural network-based model for estimating the produced power of a photovoltaic module. *Renew. Energy* **2013**, *60*, 71–78. [[CrossRef](#)]
20. Chen, H.; Chang, X. Photovoltaic power prediction of LSTM model based on Pearson feature selection. *Energy Rep.* **2021**, *7*, 1047–1054. [[CrossRef](#)]
21. Gong, G.; An, X.; Chen, Z.; Zhang, S.; Wen, Y.; Wu, Q.; Su, C. Model of load forecasting of electric vehicle charging station based on SAE-ELM. *Mod. Electr. Power* **2019**, *36*, 9–15.
22. Chen, J.; Zhang, N.; Liu, G.; Guo, L.; Li, J. Photovoltaic short-term output power forecasting based on EOSSA-ELM. *Renew. Energy Resour.* **2022**, *40*, 890–898.
23. Meng, X.; Xu, A.; Zhao, W.; Wang, H.; Li, C.; Wang, H. A new PV generation power prediction model based on GA-BP neural network with artificial classification of history day. In Proceedings of the 2018 International Conference on Power System Technology (POWERCON), Guangzhou, China, 6–8 November 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 1012–1017.
24. Xie, S.; He, S.; Yan, X.; Zhang, Q. Short-term photovoltaic power prediction based on SSA-BP neural network. *J. Zhejiang Univ. Technol.* **2022**, *50*, 628–633.
25. Wang, R.; Wang, C.; Lu, J. Short-term photovoltaic power prediction based on VMD-ICSA-LSSVM. *Manuf. Autom.* **2023**, *45*, 124–129.
26. Xue, J.; Shen, B. Dung beetle optimizer: A new meta-heuristic algorithm for global optimization. *J. Supercomput.* **2023**, *79*, 7305–7336. [[CrossRef](#)]
27. Pan, Z.; Bu, F. DV-Hop Localization Algorithm Optimized Based on Dung Beetle Optimizer. *J. Electron. Meas. Instrum.* **2023**, *37*, 33–41.
28. Hu, T.; Zhang, H.; Zhou, J. Prediction of the Debonding Failure of Beams Strengthened with FRP through Machine Learning Models. *Buildings* **2023**, *13*, 608. [[CrossRef](#)]
29. Dong, Y.; Yu, Z.; Hu, T.; He, C. Inversion of Rayleigh wave dispersion curve based on improved dung beetle optimizer algorithm. *Pet. Geol. Recovery Effic.* **2023**, *30*, 86–97.
30. Nikoo, M.; Malekabadi, R.A.; Hafeez, G. Estimating the mechanical properties of Heat-Treated woods using Optimization Algorithms-Based ANN. *Measurement* **2023**, *207*, 112354. [[CrossRef](#)]
31. Tao, X.; Guo, W.; Li, Q.; Ren, C.; Liu, R. Multiple scale self-adaptive cooperation mutation strategy-based particle swarm optimization. *Appl. Soft Comput.* **2020**, *89*, 106124. [[CrossRef](#)]
32. Mirjalili, S.; Lewis, A. The whale optimization algorithm. *Adv. Eng. Softw.* **2016**, *95*, 51–67. [[CrossRef](#)]
33. Chegini, S.N.; Bagheri, A.; Najafi, F. PSOSCALF: A new hybrid PSO based on Sine Cosine Algorithm and Levy flight for solving optimization problems. *Appl. Soft Comput.* **2018**, *73*, 697–726. [[CrossRef](#)]
34. Haklı, H.; Uğuz, H. A novel particle swarm optimization algorithm with Levy flight. *Appl. Soft Comput.* **2014**, *23*, 333–345. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.