Heliyon 11 (2025) e41892

Contents lists available at ScienceDirect

Heliyon

journal homepage: www.cell.com/heliyon

Research article

Enhanced extreme learning machine via competitive learning SSA (CL-SSA) for load capacity factor prediction

Nuriddin Tahir S Luoka *, Wagdi M.S. Khalifa

University of Mediterranean Karpasia, Turkey

ARTICLE INFO

Keywords:
SSA
Load capacity factor
Extreme learning training
Convergence precision
Global optimization

ABSTRACT

Extreme Learning Machine (ELM) is known for its fast training speed and simplicity of implementation; however, it suffers from certain limitations, including sensitivity to random initialization and inadequate weight optimization, which can result in suboptimal accuracy and precision. This study introduces an enhanced Competitive Learning Salp Swarm Algorithm (CLSSA), which integrates the Salp Swarm Algorithm (SSA) with Competitive Swarm Optimization (CSO) to improve the exploitation capabilities of the traditional CSO. The goal is to address the limitations of traditional ELM by optimizing the weights and biases of the network more effectively, thereby improving the precision and convergence speed of ELM. The research first evaluates the efficiency of the improvement made to the CLSSA optimizer in comparison with various optimization methods, using CEC 2015 benchmark functions to demonstrate the effectiveness of the proposed improvements. The results show that CLSSA outperforms other optimizers in 86 % of the CEC 2015 functions, underscoring its superior optimization capabilities. Furthermore, the study assesses the effectiveness of the CLSSA-enhanced ELM (ELM-CLSSA) in predicting the load capacity factor. The findings reveal that the hybrid ELM-CLSSA framework significantly outperforms both alternative approaches and the traditional ELM framework in terms of training and prediction accuracy, achieving an impressive accuracy rate of 97%. The algorithm’s rapid convergence, high precision, and ability to avoid local optima make it a promising solution for complex problems, such as load capacity factor prediction, which is critical for environmentally sustainable initiatives. In addition, the feature analysis conducted by ELM-CLSSA provides valuable insights into the key variables influencing load capacity factor prediction, highlighting the importance of factors such as coal energy, economic growth, technological innovation, and biomass. This study advocates for the use of the ELM-CLSSA framework to improve the precision and reliability of load capacity factor prediction, offering a valuable tool for scientists and policymakers in their efforts to promote ecological conservation and combat climate change.

1. Introduction

As humanity continues to strive for advancement, the condition of the Earth is deteriorating steadily, with significant consequences for its sustainability [1]. Humans’ impact on the Earth’s climate has a growing potential to activate important tipping points, which would accelerate global warming and result in severe and permanent repercussions [2,3]. Despite global efforts, such as the 2015 Paris

* Corresponding author.
E-mail addresses: luokanuriddin@gmail.com (N. Tahir S Luoka), wagdi.kalifa@akun.edu.tr (W.M.S. Khalifa).

https://doi.org/10.1016/j.heliyon.2025.e41892
Received 17 June 2024; Received in revised form 9 January 2025; Accepted 9 January 2025
Available online 11 January 2025

Climate Accord, where major economies committed to limiting global warming to below 1.5 °C, most nations are failing to meet their climate goals [4]. These developments underscore the urgent need for efficient resource management and reliable predictive tools to guide environmental policy and energy use. Among these analytical tools is the ecological footprint, developed by Wackernagel and Rees, which quantifies the environmental impact of human activities on the atmosphere, oceans, and terrestrial ecosystems in terms of global hectares (gha) [5,6]. While the ecological footprint provides a measure of the environmental consequences of natural resource consumption, Siche et al. contend that an exclusive focus on ecological balance does not constitute an optimal approach [7]. To address this limitation, a more robust evaluation can be achieved by calculating the ratio of biocapacity to ecological footprint (biocapacity/ecological footprint), a metric known as the load capacity factor. This ratio serves as a key criterion for assessing the sustainability of current environmental conditions. A load capacity factor value of "1″ indicates sustainability, whereas values below "1″ indicate an unsustainable ecological situation [7]. A key advantage of LCF is its ability to integrate both biocapacity and ecological footprint into a single metric, providing a comprehensive assessment of ecological sustainability [8]. However, the accuracy of LCF predictions depends on the development of robust computational models capable of handling complex and nonlinear environmental data.

In recent years, data-driven machine learning models have become increasingly relevant for predictive tasks, demonstrating exceptional capabilities in modelling nonlinear data series [9]. One such technique is the ELM, an innovative approach specifically designed as a single-layer feed-forward neural networks [10]. A defining feature of ELM is the stochastic initialization of input weights and hidden biases without iterative adjustments during the learning process. This allows for the rapid calculation of optimal output weights based solely on the predefined network structure [11]. ELM has been successfully applied across a diverse range of predictive tasks, including forecasting of wind power generation [12], daily streamflow prediction [13], weather forecasting [14], financial market analysis [15], and energy demand prediction [16]. However, due to the random initialization of its parameters, ELM exhibits fast training speed and implementation simplicity. It suffers from certain drawbacks, such as sensitivity to random initialization and a lack of proper weight optimization, which results in low accuracy. To address these limitations, the literature proposes the use of advanced optimization algorithms to fine-tune ELM's parameters, thereby enhancing its predictive performance [17].

Notably, Wang et al. proposed an Improved Hunter-Prey Optimization (IHPO) algorithm-based ELM to improve the accuracy of short-term wind power predictions [18]. This approach incorporates adaptive inertia weights and an improved population initialization within the HPO algorithm to accelerate convergence and enhance global search capabilities. The optimized parameters subsequently refine the weights and bias of the ELM, resulting in more accurate wind power forecasts. Similarly, El Bourakadi et al. investigate solar energy integration, underscoring the critical role of photovoltaic (PV) power prediction in addressing uncertainties associated with weather data [19]. They introduced a novel model that combines stacked Bidirectional Long-Short Term Memory (BiLSTM) networks with an enhanced ELM. In this framework, BiLSTM is utilized to predict weather variables that influence PV output, while the ELM forecasts PV power based on these predictions. While this model demonstrates superior performance in PV power prediction compared to traditional algorithms, its complexity may pose challenges in real-time applications, requiring substantial computational resources. Zang et al. proposed a hybrid model combining an enhanced Particle Swarm Optimization (PSO) algorithm with ELM alongside the Principal Component Analysis (PCA) [20]. This model, tested on a hot strip rolling dataset, outperforms three other models in accuracy. The method effectively addresses nonlinear challenges associated with the data, making it ideal for parameter prediction and optimization in the steel industry. However, the specific applicability of this model to non-industrial contexts remains unexplored, limiting its generalizability.

Zhu et al. introduced a hybrid PSO-ELM model to accurately predict daily reference EvapoTranspiration (ETo) in Northwest China's arid region [21]. The advantages of the PSO-ELM model included its superior accuracy in estimating ETo under various input combinations, particularly when utilizing radiation data. The introduced PSO-ELM model surpasses traditional approaches in ETo estimation accuracy, demonstrating its effectiveness. However, its performance may decline when solely relying on limited temperature data, as indicated by the moderate outcomes observed with temperature-based models. Anushkannan et al. utilized the Ant-lion Optimizer (ALO) to enhance the ELM by optimizing its input weights and hidden-layer node offsets [22]. The strength of this approach lies in the optimization of ELM's input weights and hidden-layer node offsets through ALO, resulting in a significant enhancement in detection accuracy, achieving approximately 98.8 %. This performance surpasses that of both the standalone ALO and ELM models, showcasing the effectiveness of the hybrid methodology. However, a limitation of this study is the reliance on MRI images for classification, which may not account for variations in image quality. Liu et al. introduced a new method for short-term prediction of photovoltaic (PV) power output using an Improved Chicken Swarm Optimizer (ICSO) and ELM [23]. Their approach employs a correlation coefficient method to determine input variables and improves the convergence of the CSO for optimizing the weights and thresholds of the ELM model. The results indicate that the ICSO-ELM model outperforms traditional models, achieving a Mean Absolute Percentage Error (MAPE) of 3.08 % and a Root Mean Square error (RMSE) of 5.54 %. However, the authors acknowledge that extreme weather conditions like haze, ice, and snow were not accounted for, suggesting that these can affect the model's robustness and reliability.

Recent research efforts have demonstrated substantial advancements in enhancing the performance of ELM models across various domains, as evidenced by the application of hybrid optimization techniques such as IHPO, ALO, ICSO, and PSO-based ELM in recent literature. These methodologies have successfully addressed challenges like convergence speed, global search capability, and parameter tuning, leading to improved predictive accuracy in several domains. However, these approaches also exhibit notable limitations. For instance, the computational complexity of models like BiLSTM-ELM poses challenges for real-time applications, while others, such as the ICSO-ELM and PSO-ELM models, struggle with robustness under extreme conditions or with limited data inputs. Moreover, domain-specific applicability often restricts the generalizability of these models, as highlighted in the PSO-ELM's performance on industrial datasets. Similarly, the reliance on high-quality input data, as seen in the ALO-ELM model, underscores a vulnerability to variations in data quality.

Despite the advancements demonstrated by the various hybrid ELM models across different domains, a significant gap persists in their application for predicting Load Capacity Factor (LCF). Given the critical role of LCF in evaluating environmental sustainability and optimizing resource management, the lack of research specifically addressing the capabilities of optimally tuned machine learning techniques, such as ELM, to accurately forecast LCF represents a significant gap in our understanding. By not exploring ELM's potential, we overlook valuable opportunities to refine sustainability assessments and improve predictive accuracy. Enhancing our understanding of ELM in this context could lead to more effective strategies for sustainable resource utilization and management, ultimately contributing to more informed environmental policies and practices. In response to the critical gap in LCF forecasting, this study introduces an enhanced CLSSA-ELM algorithm, leveraging the capabilities of the Competitive Learning Salp Swarm Algorithm (CLSSA) developed by Qaraad et al. [24]. This innovative approach directly addresses the inherent limitations of the traditional ELM, which is often hindered by the random initialization of its weights and biases [25]. Such random initialization not only compromises the model's predictive accuracy but also undermines the reliability of LCF prediction, which is vital for effective environmental sustainability assessments and resource management. By systematically optimizing these parameters, the proposed CLSSA-ELM framework significantly enhances the robustness and precision of LCF predictions, thereby offering a more reliable tool for decision-making in sustainable resource utilization.

The remainder of this paper is structured as follows: Section two discusses the CLSSA, ELM, and the proposed CLSSA-ELM framework. Section three presents the experiments and discussion, including a comparison of CLSSA with established optimizers and an analysis of load capacity factor prediction; section four discusses contributions and observations. Section five presents the uncertainties within the model. Finally, Section six concludes the paper.

2. Methodology

2.1. Salp Swarm Algorithm

The Salp Swarm algorithm (SSA) is a newly developed swarm optimization technique that mimics the collective movement of salps in water during food hunting [26]. In SSA, individuals are classified as leaders or followers. The leader guides the chain by finding food sources, updating positions, and directing the group. Followers mimic the leader to locate optimal food sources (F) within the search area.

- **Initialization**

SSA begins by initializing a swarm X, which consists of n salps, arranged in a matrix. This matrix is defined in Eq. (1) [24].

$$X_i = \begin{bmatrix} x_1^1 & x_2^1 & \cdots & x_d^1 \\ x_1^2 & \ddots & \ddots & x_d^2 \\ \vdots & \ddots & \ddots & \vdots \\ x_1^n & x_2^1 & \cdots & x_d^{nd} \end{bmatrix} \tag{1}$$

Where x_i^j represents the position of the *i-th* salp in the *j-th* dimension.

- **Leader's Position Update**

The leader Salp's position is updated using Eq. (2)

$$x_i^1 = \begin{cases} F_i + c_1((ub_i - lb_i)c_2 + lb_i)\ c_3 \geq 0.5 \\ F_i - c_1((ub_i - lb_i)c_2 + lb_i)\ c_3 \langle 0.5 \end{cases} \tag{2}$$

x_i^1 is the position of the leader chain salp in the *jth* dimension. F_i denotes the food position in the *j-th* dimension; ub_i and lb_i represent the upper and lower bounds of salp positions, respectively; $c2$ and $c3$ are random integers ranging from 0 to 1, as described in the original SSA paper by Mirjalili et al. [26]. This equation guides the leader salps toward the optimal solution by balancing exploration and exploitation.

- **Control Parameter**

The parameter c_1 also described in the original SSA paper by Mirjalili et al. is crucial for balancing exploration and exploitation and is computed as shown in Eq. (3) [26]

$$c_1 = 2e^{-\left(\frac{4t}{T}\right)^2} \tag{3}$$

Here, t represents the current iteration, whereas T represents the maximum number of iterations. As iterations proceed, c_1 reduces, allowing the swarm to transition from exploration to fine-tuned exploitation of the search space.

- **Followers' Position Update**

The positions of the followers in the salps chain are updated according to Eq. (4)

$$x_i^j = \frac{1}{2}k^* \, dt^2 + s_0{}^*dt \tag{4}$$

For every value of i greater than or equal to 2, x_i^j represents the location of the *jth* salp in the *ith* dimension. The variable dt indicates the duration and s_0 denotes the starting speed, with $k = \frac{s_{\text{final}}}{s_0}$ where $s_{\text{final}} = \frac{x - x_0}{t}$. Given that $s_0 = 0$, the equation simplifies to the following form Eq. (5)

$$x_i^j = \frac{1}{2}\left(x_i^j + x_i^{j-1}\right) \tag{5}$$

This equation reflects how the followers salps update their positions based on the average position of their previous and current location, effectively smoothing their movement toward the leader.

2.2. Competitive Swarm Optimization algorithm

The Competitive Swarm Optimizer (CSO) differs from many traditional algorithms, as individuals (particles) acquire information from randomly selected competitors rather than the best global or individual positions [27]. Initially, during each cycle, all the particles are arbitrarily divided into two equal groups. Competition arises among the particles within every group. The particle that possesses the highest fitness value after the competition is considered best and promptly progresses to the subsequent iteration [24]. The losing particles adjust their velocity and position based on Eq. (6) and Eq. (7)

- **Velocity Update**

$$V_l^{t+1} = R_1^t V_l^t + R_2^t\left(X_w^t - X_l^t\right) + \phi R_3^t\left(X^{-t} - X_l^t\right) \tag{6}$$

```
Algorithm 1. Steps of CSO

Initialize position X_i^t and X_i^t according to upper and lower bounds.
Set t = 0
while (ending condition is not reached):
    for each particle P_i^t:
        Evaluate the new X_i^t and record it as fitness.
        if (Fitness < best_fitness) then:
            update the Best_position and Best_fitness.
        end if
    end for
    P^{t+1} ← ∅
    for each swarm in (number of Search Agents / 2)
            Randomly select particles P_{r1}^t and P_{r2}^t from P^t
             If f(X_{r1}^t) is better than f(X_{r2}^t) then
                P_w^t ← P_{r1}^t, P_l^t ← P_{r2}^t
            else
                P_w^t ← P_{r2}^t, P_l^t ← P_{r1}^t
           end if
            for each dimension
                V_l^{t+1} = R_1^t V_l^t + R_2^t (X_w^t − X_l^t) + φR_3^t (X^{−t} − X_l^t)
                X_l^{t+1} = X_l^t + V_l^{t+1}
                P^{t+1} ← P^{t+1} ∪ {P_w^t, P_l^{t+1}}
                P^t ← P^t \ {P_{r1}^t, P_{r2}^t}
            end for
      end for
    t ← t + 1
End while
```

Fig. 1. Pseudo-code of CSO.

Where V_l^{t+1} is the updated velocity of the losing particle; R_1^t, R_2^t, and R_3^t are random vectors from the interval $[0, 1]^n$ at iteration t. These random vectors were defined by Ran Cheng and Yaochu Jin in their development of the CSO [27]. X_w^t is the position of the winner particle while X_l^t is the position of the loser particle. ϕ is the control parameter that influences the mean location X^{-t} of the current swarm at iteration t.

- **Position Update**

$$X_l^{t+1} = X_l^t + V_I^{t+1} \tag{7}$$

Where X_l^{t+1} is the updated velocity of the losing particle in the next iteration. These updates ensure that the losing particles adjust their positions relative to both the winner's position and the swarm's overall mean position, balancing exploration and exploitation in the search space [28]. The CSO pseudo-code given in Fig. 1, demonstrates that implementing CSO is straightforward; yet, two factors may cause reduced search effectiveness. Initially, the particles are arbitrarily paired in the competition, potentially impacting overall exploration. Furthermore, CSO exclusively updates the most unfavourable particles while disregarding the successful ones, which hinders the potential for local exploitation. Also, SSA is hindered by its low convergence rate and inclination to be stuck in local optima. Thus, Qaraad et al. integrate SSA and CSO into a unified method to enhance the overall efficiency of the CSO algorithm [24]. This modification resulted in a better rate of convergence and an increased rate of search space exploration.

2.3. Extreme learning machine

ELMs, as described by Huang et al., are networks of neurons that consist of three distinct layers: the input, the hidden, and the output layer [29]. ELM distinguishes itself from Back Propagation Neural Network (BPNN) by utilizing solely one hidden layer. ELM algorithms do not require network errors to be propagated backwards for weight updates in order to improve the network's accuracy [30]. Fig. 2 visually represents the network structure of ELM. Let W_1 be an l x m-matrix with random starting values distributed uniformly over the boundaries of $[-1, 1]$. The starting values of the l x 1 matrix have an even distribution in the interval [0, 1]. It should be noted that m indicates the total number of neurons in the input layer, n corresponds to the number of neurons in the output layer, and l is the number of neurons in the hidden layer. One essential part of ELM involves selecting an activation rule, also known as an activation function, denoted as $F(x)$. The outcome of the hidden layer is calculated as follows in Eq. (8).

$$H = F(W_1 \cdot P + B) \tag{8}$$

Here, W_1 represents the interconnecting weight matrix of neurons linking the input layer and the hidden layer(randomly initialized), B represents the bias vector at the hidden layer (randomly initialized), W_2 denote the interconnecting weight matrix of neurons connecting the hidden layer to the output layer, and P denotes the input data. The connecting matrix of neurons from the hidden to the output layer is denoted by W_2 is determined using Eq. (9).

$$W_2 = H^{-1} \times T \tag{9}$$

Where T denote the target data, H^{-1} is the Moore-Penrose pseudoinverse of the matrix H, H denote the hidden layer's output matrix,

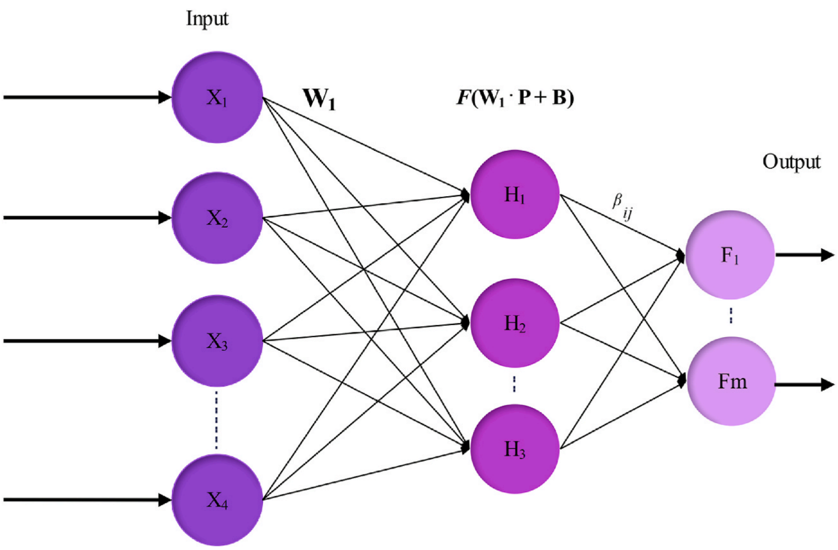


Fig. 2. Extreme learning machines.

the output Y of the ELM is calculated as follows in Eq. (10).

$$Y = H \times W_2 \tag{10}$$

In ELM, the interconnecting vector between the hidden and output layers is computed. While matrix such as W_1 and B are initialized with arbitral values. The ELM method eliminates errors in neurons in the output layer and focuses on generating weights for the output that minimizes the error. The error minimization objective is mathematically modelled in Eq. (11).

$$min \parallel H \times W_2 - T \parallel^2 \tag{11}$$

The objective function or cost function is defined in Eq. (11). The objective is to find the optimal set of weights w_1 and B that minimizes this error term. The more this error is reduced, the closer the predicted output $H \times W_2$ is to the actual target T.

2.4. Proposed CLSSA-ELM FrameWork

From the literature reviewed in the introduction, the ELM is well-established. Specifically, ELM suffers from low prediction accuracy due to the inherent method employed to adjust the network's weights and biases. The primary objective of this research is to address this limitation by developing a hybrid model that integrates ELM with an enhanced optimization algorithm. This algorithm is designed to effectively adjust the weights and biases of the network, thereby improving prediction accuracy.

The proposed enhanced optimizer, termed Competitive Learning-based Salp Swarm Algorithm (CLSSA), builds upon the classical Competitive Swarm Optimization (CSO) by incorporating Salp Swarm Algorithm (SSA) position update mechanism in Eq. (2). This enhancement results in a more robust and effective optimizer for complex optimization problems, including the optimization of ELM parameters. The innovation introduced by CLSSA involves an exploitative position update strategy from SSA, the traditional CSO divides the population into two groups: "winners" and "losers." This division is achieved by ranking individuals based on the quality of their solutions, where the best-performing individuals (winners) have lower fitness values in a minimization problem, and the worst-performing individuals (losers) have higher fitness values. The competitive learning strategy aims to enhance the quality of solutions over iterations. Specifically, the worst-performing individuals update their positions by learning from the best-performing individuals, using the position update strategy in the traditional CSO, as expressed in Eq. (6) and Eq. (7). The traditional CSO does not update the winners. In CLSSA, the best-performing individuals are updated using a new exploitative position update strategy from SSA, as shown in Eq. (2). Fig. 3 presents the flowchart of the CLSSA iteration process, illustrating how the mechanism is implemented. By introducing two distinct learning strategies within the population, CLSSA enhances the quality of solutions by promoting both exploration and exploitation, ultimately leading to improved optimization performance.

The flowchart of the new hybrid CLSSA-ELM is given in Fig. 3. This research focuses on optimizing the settings W_1 and B of the ELM algorithm using the CLSSA in order to enhance the reliability of ELM by improving the accuracy of the conventional ELM. The conventional ELM has respectable generalization capability; however, the random initialization of weights and biases can lead to less accurate results. Therefore, the input weight W_1 and the bias B of the hidden layer neurons in the ELM network are regarded as the individuals represented by a vector in the CLSSA optimizer. Each individual in the population is represented as a vector containing the corresponding values of the weights and biases of the ELM network. The CLSSA optimization algorithm adjusts these weights and biases iteratively until their optimal values are achieved. Initially, the weights and biases for each individual are randomly initialized within the search space, which is bounded between -1 and 1.

The evaluation of each individual is based on the MSE (Error minimization function) of the ELM model. When the weights and biases of the ELM are set to the values specified by a given individual, the MSE serves as an indicator of that individual's performance in

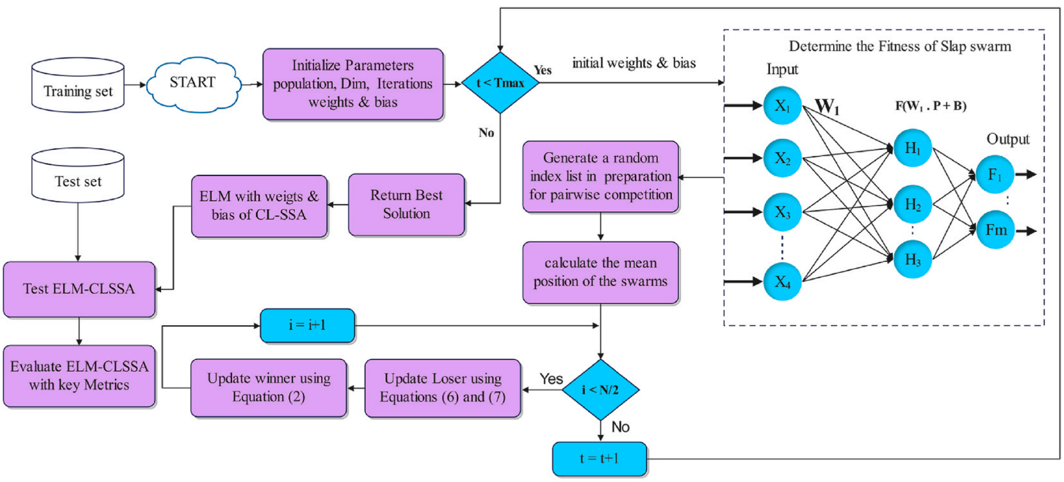


Fig. 3. CLSSA-ELM flowchart.

optimizing the model. In other words, the MSE determines the fitness of each individual, measuring how accurately the weights and biases align with the optimal solution. Following this evaluation, the population is divided into "winners" and "losers" based on their fitness values. The weights and biases of the winners are updated using Eq. (2), while the adjustments for the losers follow the strategies outlined in Eqs. (6) and (7). The process of CLSSA continuously adjusting the weights and biases is controlled by a predefined number of iterations. After these iterations, the individual with the optimal weight and bias is returned by the CLSSA algorithm, and these values are assigned to the ELM model. To assess the performance of the optimized CLSSA-ELM model, it is then tested on new and unseen data, referred to as the test data, to evaluate its accuracy. The CLSSA algorithm is particularly well-suited for optimizing the ELM model due to the introduction of the SSA exploitation strategy, which enhances convergence accuracy by guiding the population toward optimal weights and biases. Additionally, the improved exploration capability of CLSSA allows for efficient searching within the problem space, increasing the likelihood of finding an optimal solution.

3. Experiments and discussion

3.1. Comparison of CLSSA with established optimizers

This study performs experimentation analyses utilizing the CEC2015 test suite to validate the capability of CLSSA. The suite consists of single-modal (F1-F2), multimodal (F3-F5), hybrid (F6-F8), and composite functions (F9-F15). Unimodal problems have only a single ideal solution. However, these complex mathematical problems, which are often sphere-shaped or valley-shaped, can be utilized to showcase the method's capacity for exploitation. Multimodal problems provide a greater number of locally ideal solutions, which may be utilized to demonstrate the technique's capacity for exploration. Hybrid and composite methods are effective in thoroughly assessing the technique's effectiveness across several dimensions. For a comprehensive interpretation of CEC2015. The problems and their formulations are presented in Ref. [31]. In the experiment, every optimizer assigns a value of 30 to the size of N (population size) and a value of 1000 as the maximum number of iterations T. The method is executed individually thirty times, and subsequently, the average and deviation are calculated for the purpose of comparative analysis. The corresponding optimizer settings are listed in Table 1, as described in their respective literature. The techniques utilized in this study include the Exponential Differential Optimizer (EDO) [32], Grey Wolf Optimizer (GWO) [33], Harris Hawk Optimizer (HHO) [34], and Salp Swarm Algorithm (SSA) [35] and Competitive Swarm Optimizer [27]

3.2. Statistical results and non-parametric analysis

The results of the statistical analysis are presented in Table 2, which demonstrates the superior performance of the CLSSA across multiple test functions. Specifically, CLSSA exhibits exceptional proficiency in exploiting functions F1 and F2 when compared to other optimization methods. While both the traditional SSA and CSO produced competitive results overall, the fusion of SSA exploitation techniques with CSO traditional position update mechanism encouraged local search around the current positions, thereby yielding more accurate solutions. Table 2 also highlights CLSSA's strong capability for exploring multimodal functions, particularly those in the F3-F5 range. Although other algorithms, such as EDO, GWO, HHO, SSA, and CSO performed well on these functions, CLSSA demonstrated an enhanced and balanced exploration strategy. This improvement can be attributed to the ability of CLSSA to generate new positions for suboptimal individuals when the best individuals become trapped in locally optimal solutions. The random variables introduced in Eq. (6) allow the worst-performing individuals to explore new regions of the search space, thereby increasing the likelihood of escaping local optima. In the case of the more complex hybrid and composite problems (F7-F15), CLSSA also exhibited superior efficiency, as evidenced by the mean fitness values reported for functions F7, F8, F9, F10, F11, F12, F14, and F15. This demonstrates that CLSSA strikes an optimal balance between exploration and exploitation, preventing excessive emphasis on either, which can impede convergence in challenging optimization problems. The synergy between SSA and CSO in CLSSA enhances both exploration and exploitation capabilities, allowing it to leverage the strengths of both algorithms for improved overall performance. Moreover, the standard deviation (STD) values in Table 2 reveal the strong consistency of CLSSA across a wide range of problems. The GWO optimizer performed particularly well on F12, surpassing other methods in that instance and producing competitive results comparable to CLSSA on F6.

To statistically evaluate the significance of CLSSA's performance improvements compared to other optimizers, two widely used tests were conducted: the Friedman Ranking (FR) and Wilcoxon Rank-Sum (WRS) tests. The results of the FR, as shown in Table 2, indicate that CLSSA ranks first among the evaluated optimizers, demonstrating its statistical superiority. Traditional SSA and CSO

Table 1
Algorithm parameters.

Algorithms	Parameter setting
CLSSA	$\phi = 0.3$
SSA	$c1 = [2/e, 2]$
GWO	$a_0 = 2$
HHO	$E_0 = [2, 0]$
EDO	Switch Parameter = 0.5
CSO	$\varphi = 0.5$

Table 2
Results of CLSSA and compared Optimizer on CEC 2015.

		CLSSA	EDO	GWO	HHO	SSA	CSO
F1	Mean	**4.70E+4**	1.67E+10	1.12E+9	4.41E+10	9.88E+4	2.59E+9
	Std	**7.90E+4**	4.59E+9	9.76E+8	7.92E+9	2.44E+5	1.26E+9
F2	Mean	**3.42E+4**	5.17E+4	3.79E+4	6.07E+4	4.42E+4	1.10E+5
	Std	**1.73E+3**	9.70E+3	8.18E+3	1.22E+4	1.62E+4	3.37E+4
F3	Mean	**3.16E+2**	3.38E+2	3.17E+2	3.41E+2	3.20E+2	3.32E+2
	Std	4.95	**2.34**	3.66	2.49	3.10	3.17
F4	Mean	**3.94E+3**	8.46E+3	5.66E+3	6.52E+3	3.95E+3	7.54E+3
	Std	**3.83E+2**	8.13E+2	2.04E+3	7.30E+2	5.65E+2	5.24E+2
F5	Mean	**5.00E+2**	5.04E+2	5.03E+2	5.02E+2	5.01E+2	5.03E+2
	Std	**2.19E-1**	5.94E-1	4.55E-1	5.12E-1	2.52E-1	4.95E-1
F6	Mean	**6.01E+2**	6.03E+2	**6.01E+2**	6.05E+2	**6.01E+2**	6.01E+2
	Std	**1.17E-1**	7.24E-1	1.49E-1	3.51E-1	1.29E-1	4.27E-1
F7	Mean	**7.01E+2**	7.43E+2	7.02E+2	7.71E+2	**7.01E+2**	7.04E+2
	Std	4.01E-1	8.75	2.61	1.52E+1	**3.07E-1**	4.56
F8	Mean	**8.08E+2**	4.12E+5	1.39E+3	3.28E+6	8.12E+2	5.53E+4
	Std	**2.43**	2.37E+5	8.45E+2	1.87E+6	4.92	3.68E+4
F9	Mean	**9.12E+2**	9.14E+2	9.13E+2	9.13E+2	9.13E+2	9.13E+2
	Std	4.09E-1	**1.88E-1**	5.13E-1	3.49E-1	4.94E-1	3.38E-1
F10	Mean	**7.24E+5**	1.29E+7	1.06E+6	4.65E+7	8.62E+5	1.11E+7
	Std	**4.33E+5**	7.76E+6	8.00E+5	2.13E+7	5.48E+5	2.89E+6
F11	Mean	**2.72E+3**	2.52E+7	3.68E+3	2.50E+7	4.00E+3	5.16E+5
	Std	**2.63E+3**	1.53E+7	3.66E+3	2.46E+7	3.63E+3	3.51E+5
F12	Mean	**3.46E+3**	1.45E+10	3.62E+3	2.52E+10	4.08E+3	1.51E+9
	Std	9.05E+2	6.61E+7	**7.66E+2**	2.79E+9	1.20E+3	2.75E+7
F13	Mean	1.60E+3	1.87E+3	**1.57E+3**	2.12E+3	1.60E+3	1.72E+3
	Std	2.84E+1	7.28E+1	**9.92**	2.51E+2	3.21E+1	1.10E+2
F14	Mean	**2.20E+3**	7.63E+3	2.54E+3	6.13E+3	2.28E+3	3.83E+3
	Std	**1.97E+2**	1.00E+3	2.52E+2	2.68E+3	2.91E+2	8.76E+2
F15	Mean	**2.27E+3**	3.04E+3	2.87E+3	2.85E+3	2.33E+3	2.82E+3
	Std	1.18E+2	2.04E+2	**3.24E+1**	5.97E+1	1.10E+2	1.74E+2
	Friedman Mean	1.7133	5.2377	2.6156	5.1044	2.0800	4.2289
	Friedman Rank	1	6	3	5	2	4
	Wilcoxon P-Value	–	1.6532e-06	2.4720e-05	2.1491e-08	4.1746e-03	5.1357e-04

ranked second and fourth, respectively. The WRS test, conducted at a significance level of 0.05, supports these findings. A p-value of less than 0.05 indicates that CLSSA's performance is significantly superior to that of the other optimizers. As shown in Table 2, the p-values confirm that CLSSA achieves a substantial performance boost compared to the competing techniques. In conclusion, the results of this study clearly indicate that CLSSA provides a marked improvement over other algorithms in terms of both exploration and exploitation capabilities, consistency, and statistical significance. This positions CLSSA as a powerful and effective tool for addressing complex optimization problems.

3.3. Convergence Curves and box analysis

To provide a comprehensive analysis of the performance of the competing optimizers, convergence charts for all fifteen problems (F1-F15) are displayed in Fig. 4. In these charts, the y-axis represents the average solutions obtained across all iterations, while the x-axis represents the number of iterations. The convergence graphs offer a clear depiction of the optimization process for each algorithm, highlighting their ability to balance exploration and exploitation effectively. The results show that optimizers such as EDO, HHO, GWO, SSA, and CSO exhibit lower efficiency, as they tend to suffer from an imbalance between the exploration and exploitation stages. This imbalance often causes these algorithms to converge prematurely or stagnate, leading to suboptimal solutions. In contrast, the convergence trajectories of CLSSA indicate that the incorporation of an exploitative position update strategy from SSA significantly enhances the search mechanism, allowing for a more balanced transition between exploration and exploitation.

The contrasting techniques in many cases, as shown in the graphs, tend to settle on suboptimal solutions, demonstrating the limitations of their search strategies. However, CLSSA consistently outperforms these methods, demonstrating superior convergence speed and accuracy. This is particularly evident in the results for problems F1, F2, F3, F5, F9, F10, F11, F12, F14, and F15, where CLSSA not only reaches better solutions but also does so more rapidly than the other comparative optimizers. The convergence performance of CLSSA further substantiates its improved accuracy compared to SSA and CSO, emphasizing the benefits of integrating the SSA exploitation method into the traditional CSO framework. This enhancement enables CLSSA to maintain robust exploration while efficiently refining solutions in the exploitation phase, resulting in superior outcomes across a wide range of optimization problems.

Fig. 5 presents the box plots of the compared optimizers applied to the CEC 2015 (F1-F15) benchmark problems. In conjunction with the data displayed in Table 2 and the box plots in Fig. 5, it is evident that CLSSA exhibited the most effective performance and demonstrated the highest level of consistency in converging toward optimal solutions across the majority of the evaluated problems.

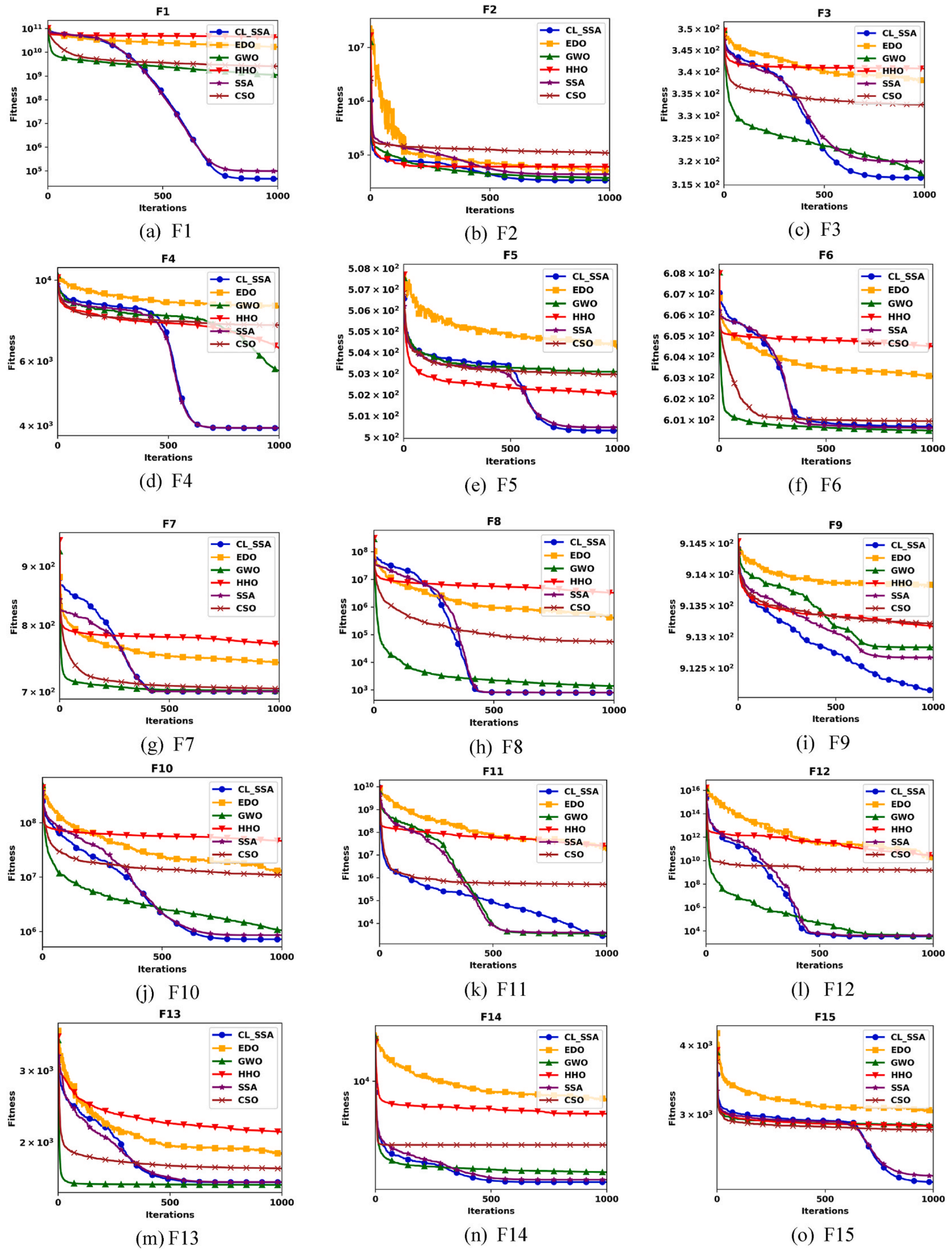
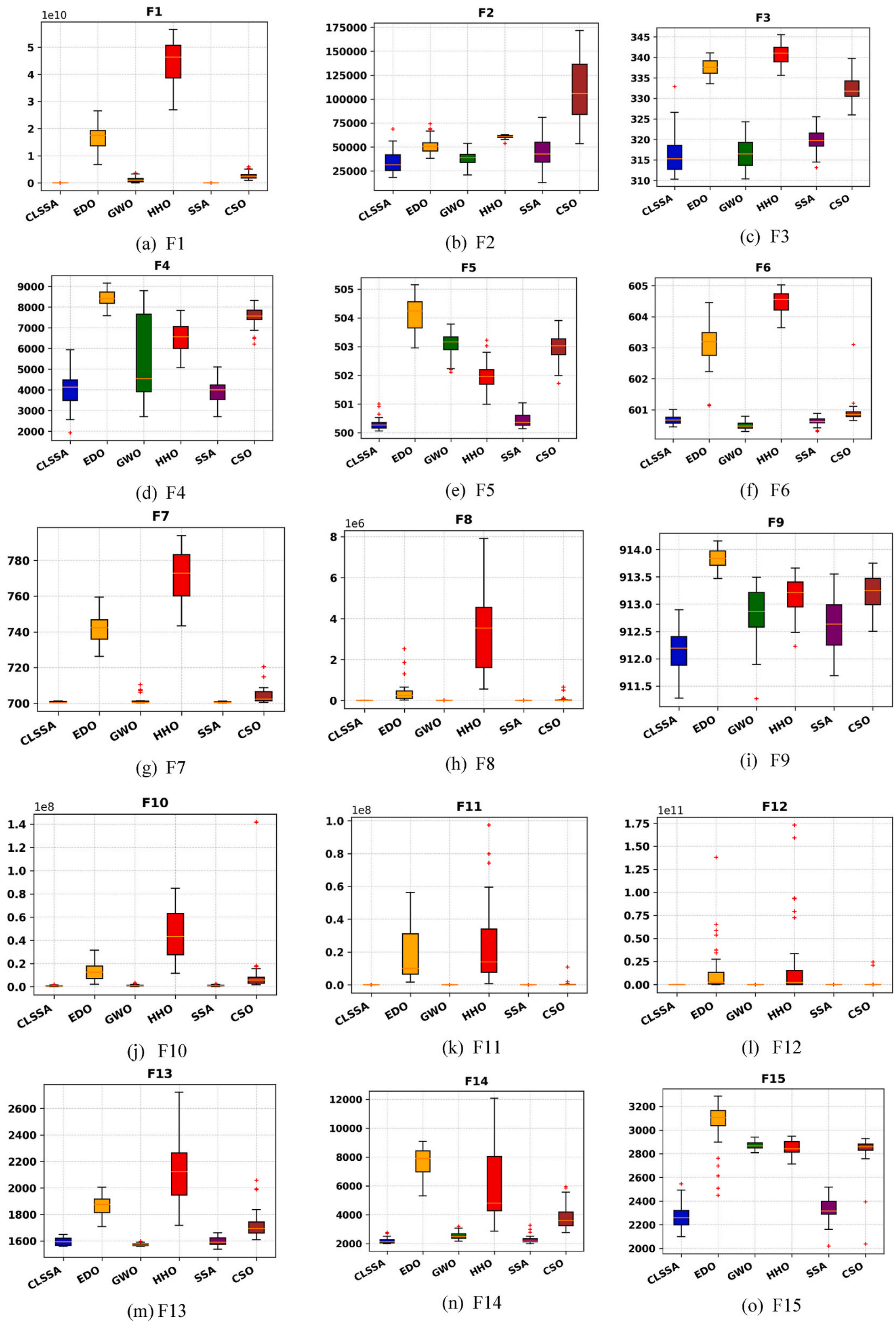


Fig. 4. Convergence Curve of CLSSA and other Optimizers on CEC 2015 functions (F1-F15).

The box plots illustrate the dispersion of the most optimal outcomes achieved by the algorithms over 30 independent runs, providing valuable insights into their performance stability. In the box plots, the red "+" symbols represent anomalies or outliers data points that deviate significantly from the expected distribution. The orange line within each plot marks the median of the distribution, serving as an indicator of the central tendency of the results. The length of the boxes, which encapsulate the interquartile range, reflects the steadiness and regularity of the algorithms in finding suitable solutions. A shorter box height suggests higher consistency and reliability in solution quality. As depicted in Fig. 5, CLSSA exhibits relatively small box heights for a significant portion of the problems, highlighting its dependable performance in identifying optimal or near-optimal solutions. This consistency underscores the



(a) F1 (b) F2 (c) F3 (d) F4 (e) F5 (f) F6 (g) F7 (h) F8 (i) F9 (j) F10 (k) F11 (l) F12 (m) F13 (n) F14 (o) F15

Fig. 5. Box Plot of CLSSA and other Optimizers on CEC 2015 functions (F1-F15).

algorithm's robustness in maintaining a balanced exploration-exploitation process across multiple runs, minimizing the variability in its results. Furthermore, CLSSA's ability to converge toward optimal solutions with minimal variance, as evidenced by the compact box plots, contrasts with the more variable performances of other optimizers, whose larger box heights and frequent outliers suggest less reliability in achieving consistent results.

3.4. Exploration and Exploitation Analysis

Balancing exploration and exploitation is essential for effectively navigating the solution space, as it allows the algorithm to discover diverse potential solutions while simultaneously refining the best-known solutions to achieve optimal performance [36]. To gain deeper insights into the dynamics of exploitation and exploration in CLSSA while it seeks the optimal solution, Fig. 6 offers a visual representation of these two processes within the proposed algorithm using functions F1, F3, F6, F7, F8, and F10. Each image in the figure includes two distinct lines: the blue line represents the optimizer's exploration phase, where it searches for new potential solutions, while the red line illustrates the exploitation phase, where it refines and improves the most promising solutions identified thus far. At the initial stages, the recommended CLSSA algorithm places a strong emphasis on exploration, focusing on surveying a wide range of possible solutions while minimizing its efforts on exploitation. This phase allows CLSSA to broadly investigate the search space and avoid premature convergence to suboptimal solutions. However, as the algorithm progresses through the iterations for most of the selected problems, there is a noticeable and timely shift towards exploitation. In the latter stages, the focus shifts predominantly to refining and optimizing the best-found solutions using the SSA operators, enabling CLSSA to capitalize on its earlier exploration efforts.

This strategic transition from exploration to exploitation allows CLSSA to achieve an effective balance between these two critical phases of the optimization process. By beginning with robust exploration and then gradually transitioning to focused exploitation, CLSSA ensures a comprehensive search of the problem space while maintaining the capability to fine-tune solutions as the algorithm converges. The balance between exploration and exploitation, as depicted in Fig. 6, provides key insights into the superior performance of CLSSA in solving complex optimization problems. This well-managed equilibrium allows CLSSA to avoid the pitfalls of excessive exploration, which can slow convergence, and over-exploitation, which can lead to premature convergence on suboptimal solutions. Ultimately, this balance contributes to the algorithm's ability to deliver consistent and high-quality solutions across a wide range of optimization tasks.

3.5. Computation time complexity

Table 3 depicts the average runtime of all optimizers on CEC2015 functions. From Table 3 and it is evident that the CLSSA algorithm exhibits a lower computational overhead compared to traditional SSA and CSO. Specifically, the average runtime of CLSSA (15.27 s) is significantly faster than SSA (22.70 s) and CSO (45.06 s), while being competitive with other swarm-based and meta-heuristic optimizers like EDO and HHO. EDO exhibited the least computational runtime amidst the compared optimizer. The superior computational efficiency of CLSSA compared to the traditional CSO and SSA can be attributed to the following. CLSSA integrates a simplified update mechanism compared to both SSA and CSO. While CSO employs a velocity-based search mechanism similar to PSO, which can result in high computational costs due to the frequent velocity updates and the large number of iterations required to achieve optimal solutions and SSA is known for its strong exploration capabilities but often faces challenges in exploitation, leading to

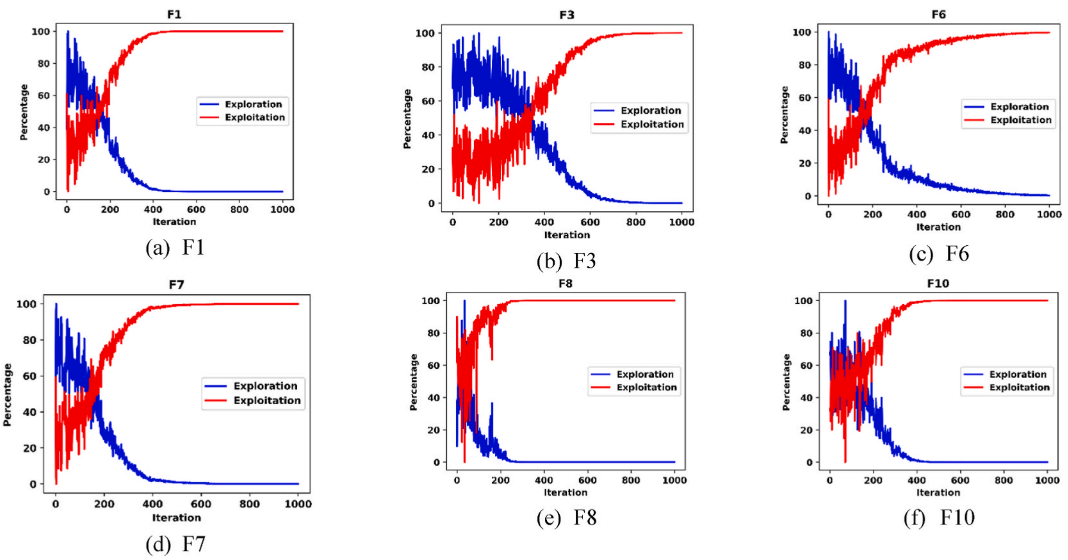


Fig. 6. Exploration and exploitation analysis of CLSSA on CEC 2015 functions.

Table 3
Average runtime of optimizers on CEC2015 functions.

Optimizer	Average Runtime (Seconds)
CLSSA	15.27
EDO	15.01
GWO	16.48
HHO	16.03
SSA	22.70
CSO	45.06

a slower convergence process. This extended search phase can increase computational time, CLSSA adopts a more straightforward competitive learning strategy from CSO and an exploitation position update mechanism from SSA, with less internal iteration and within the main maximum iteration. This minimizes the amount of computational resources needed per iteration and ensures that each iteration efficiently contributes to improving the solution quality without unnecessary redundancy.

3.6. Load capacity factor prediction

3.6.1. Data

The details of the dataset used in this experiment are detailed in Table 4. The target feature is the Load Capacity Factor, whereas the input features are Economic Growth, Coal Energy, Biomass Energy, Renewable Energy, Natural Resources, and Political Risk. To ensure a rigorous and well-generalized experiment, all the machine learning approaches and hybrid approaches underwent training and testing utilizing quarterly data spanning from 1984 to 2022 from South Africa. The correlation heat map, which displays the measure of dependence among all features, is shown in Fig. 7. The overall dispersal of every factor (Load Capacity Factor, Economic Growth, Coal Energy, Biomass Energy, Renewable Energy, Natural Resources, and Political Risk) and its trend over the years can be observed in Figs. 8 and 9, respectively. Various characteristics may exhibit distinct units and ranges. Scaling is a technique used to guarantee that all features have equal importance in the model. It prevents features with greater ranges from overpowering the learning process. In this research, the process of standardization is employed. This approach standardizes features by removing the mean and scaling to have a variance of one. As a result, the adjusted data will have a mean value of 0 and a standard deviation of 1. The formula for standardization is given in Eq. (12):

$$X_s = \frac{X - \mu}{\sigma} \tag{12}$$

Where X is the original feature value, μ is the mean of the feature, σ is the standard deviation of the feature, and X_s is the scaled data. Variability, complexity and heterogeneity of the dataset can stem from several aspects. The diversity in measurement units and scales increases the need for standardization to prevent features with larger magnitudes from disproportionately influencing the model. The standardization process used in this study (Eq. (12)) ensures that all variables contribute equally to the prediction process by converting them to a standardized scale. The data covers a long period, from 1984 to 2022, and captures fluctuations in technological innovation, energy consumption, and political stability, among other factors. Quarterly data points introduce additional complexity, as many input features, such as energy use or political stability, can show seasonal trends or cyclical behaviors that could influence the load capacity factor. The presence of such temporal trends requires a model capable of adapting to changes over time, ensuring robust predictions across different periods. The correlation heat map reveals the extent of interdependence between the features. Some factors, such as load capacity factor and energy (coal), have strong correlations, while others may have weaker or even nonlinear relationships with the target variable. The complexity of these relationships, as well as the presence of interactions among input variables, necessitates an advanced optimization algorithm to capture these dynamics effectively. CL-SSA contributes to managing the variability and complexity of the dataset in the CLSSA-ELM model through the following methods.

1. CL-SSA optimizes the weights and biases of the ELM, ensuring that the model can effectively learn from the dataset's complexities, such as the wide range of values and the nonlinear relationships between input features and the target variable. By iterating through possible weights and biases, CL-SSA identifies the best-performing configurations that minimize the prediction error.

Table 4
Description of variables.

Variables	Metric	Sourced
Load capacity factor	Hectares per capita	Global footprint
Economic growth	Constant 2015 US$	World Bank Database
Coal energy	Exajoules	British Petroleum database
Biomass energy	Tonnes	Material Flows Database
Natural resources	% of GDP	World Bank Database
Political risk	Index	PRS database
Technological Innovation	Addition of both residents and non-resident patent applications	World Bank Database

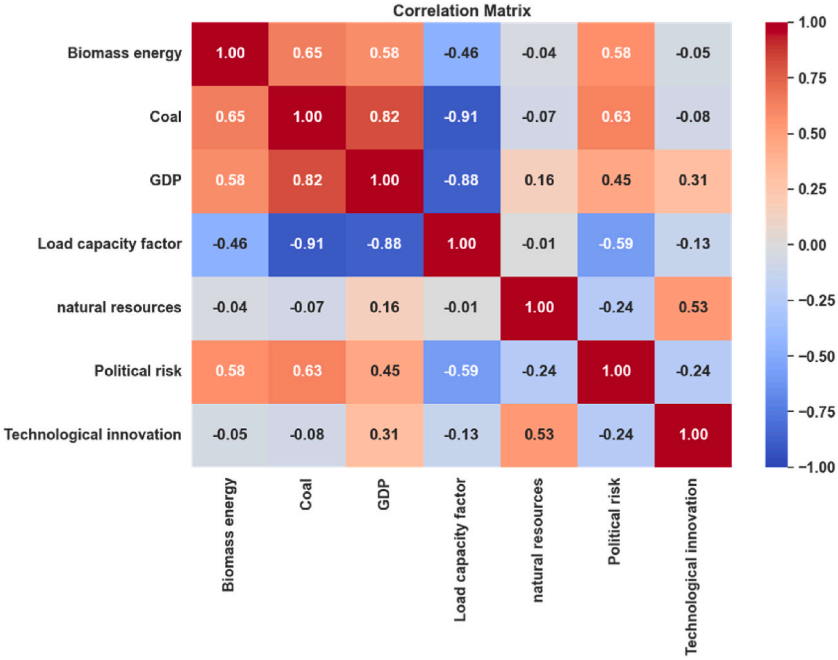


Fig. 7. Correlation heat map.

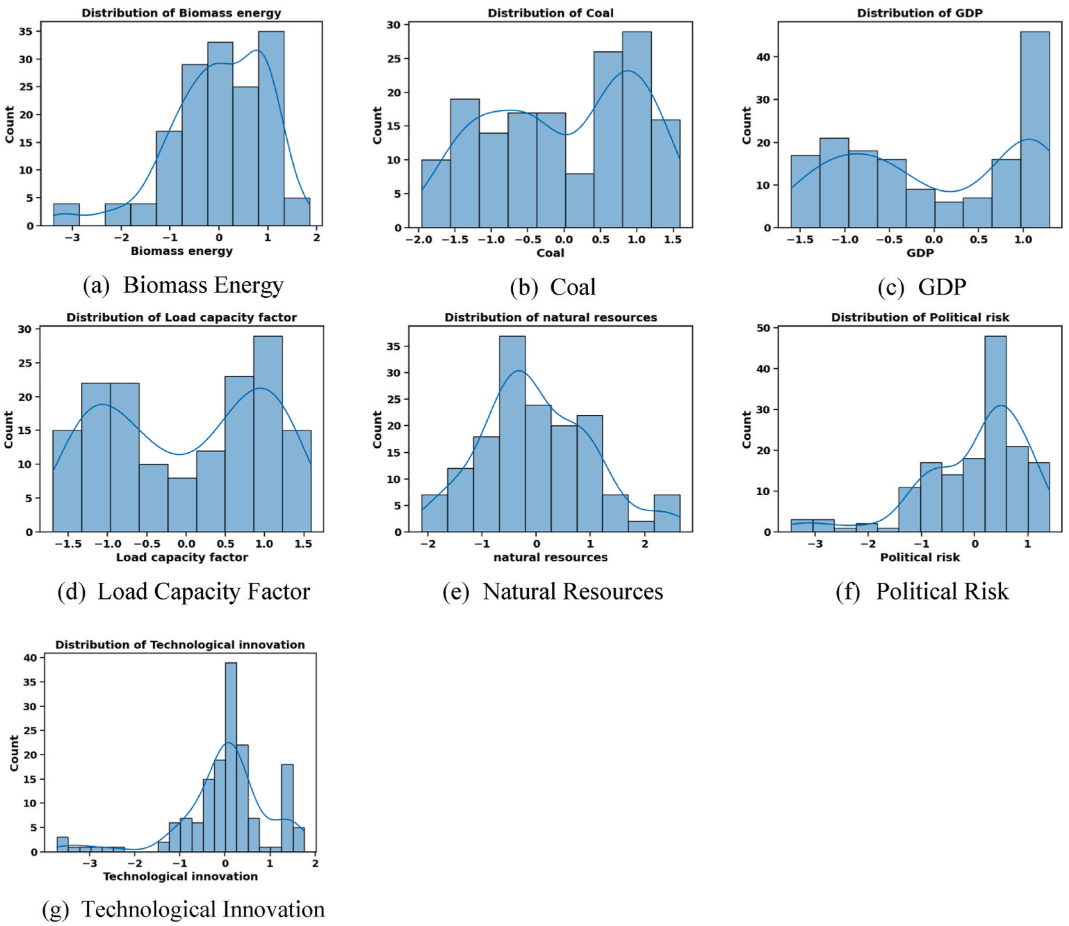
2. Due to the diverse nature of the input features, the optimization problem involves searching through a multidimensional space to identify the best model parameters. CL-SSA excels in exploiting the explored regions of the complex and multidimensional search spaces, leveraging the exploitation position update strategy to refine the search process. The algorithm's balance between exploration and exploitation ensures that it does not get trapped in local minima, enabling it to find the optimal parameters even in highly variable datasets.
3. Non-parametric uncertainties, such as noise in the data, can degrade model performance. CL-SSA mitigates the impact of these uncertainties by performing robust optimization, allowing the ELM to remain resilient even when the dataset is noisy and non-linear. Additionally, by optimizing the weights and biases of the ELM, CL-SSA ensures that the model is less sensitive to variations in input data and can provide more stable predictions.
4. The presence of correlations and non-linear relationships between the input features and the target variable presents a challenge for traditional optimization techniques. CL-SSA, through its iterative and competitive learning approach, can effectively handle such nonlinearities and correlations. It adapts to these complexities by refining the model's parameters in a way that captures the underlying relationships, improving the overall prediction accuracy.

3.6.2. Evaluation metric

Evaluation metrics are crucial in machine learning because they provide a quantitative basis to assess the performance of a model. These metrics help determine how well a model makes predictions and enables comparisons between different models. Without evaluation metrics, it would be challenging to gauge the accuracy, reliability, and overall effectiveness of a model, leading to sub-optimal decisions in model selection and tuning. Therefore, Table 5 gives details of various metrics that are utilized in this work. Evaluation metrics such as R^2, RMSE, MSE, MAE, and NRMSE are vital for assessing the performance of machine learning models. They help in understanding the accuracy, reliability, and overall effectiveness of models, guiding improvements and comparisons to select the best model for a given task. Each metric offers a different perspective on model performance, making it essential to consider multiple metrics for a comprehensive evaluation. Here, N indicates the number of data points, σ represents the deviation of data points, Y_i^{Exp} refers to the ith observed data point, Y_i^{ELM} represent the ith approximated value using any of the ELM model, and $\overline{Y}$ denote the mean of the data.

3.6.3. Load capacity prediction, discussion and experiment

In this section of the experiment, we maintain the parameters of each nature-inspired algorithm as previously established in Table 1, while the population size and number of iterations are specified as 30 and 100, respectively. Also, 70 % of the data is used to train each model, while 30 % is used for testing. The parameters fine-tuned for the ELM model by various optimizers are the weights and biases. The upper and lower bounds of the problem space for optimizing the weights and biases of the ELM model are set between −1 and 1. Five optimizers and CLSSA are employed to improve the configurations of the weights and biases of the ELM. The outcomes of this optimization process are presented in Tables 6 and 7. These tables provide four error measurement metrics (RMSE, MSE, MAE, and NRMSE) and one precision statistic (R^2). The statistics demonstrate that the MSE of the CLSSA method for estimating load capacity



(a) Biomass Energy (b) Coal (c) GDP (d) Load Capacity Factor (e) Natural Resources (f) Political Risk (g) Technological Innovation

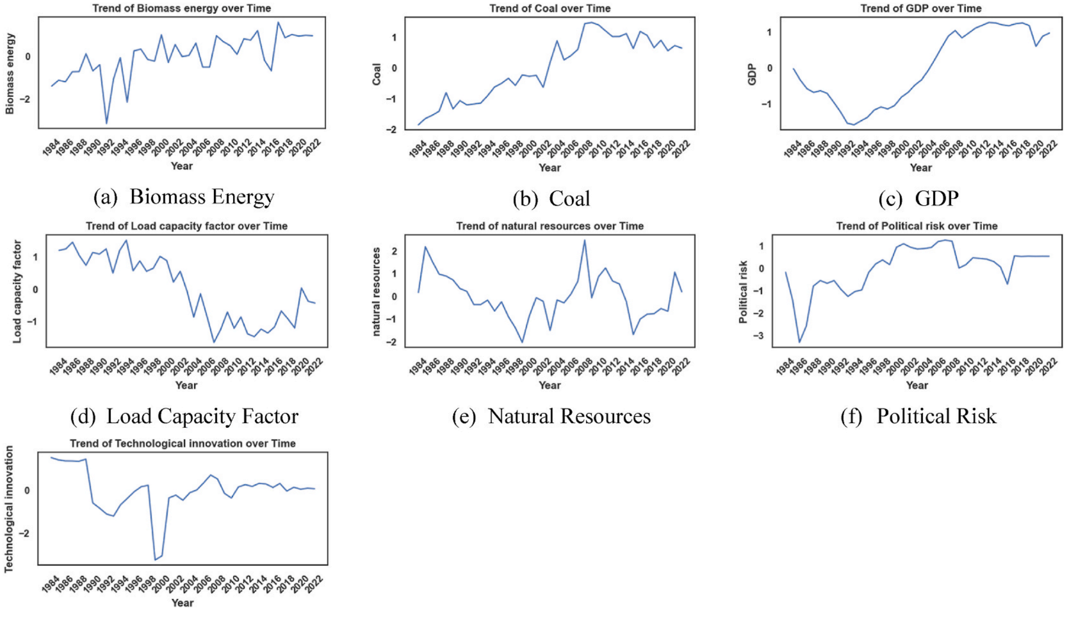
Fig. 8. Data distribution of all features in the dataset.

factor in South Africa attained the lowest values compared to the five other optimizers and the ELM in both training and testing datasets.

The MSE values for CLSSA were 0.00167 and 0.002633, respectively. For the SSA, the MSE values were 0.003715 and 0.003971, CSO reached an MSE of 0.004055 and 0.003781 for training and testing, respectively. The discrepancy in forecasting error of load capacity factor between CLSSA, CSO and SSA substantially differ, this highlights the efficiency of the proposed CLSSA in handling complex optimization problems. The MSE of the nature-inspired method ELM for training is shown in Fig. 10. This graph represents the performance of the algorithm over the specified number of iterations. The CLSSA method demonstrates its resilience in exploring a challenging terrain to achieve the lowest error, thereby improving the model's learning capacity. Furthermore, it is evident that the CLSSA method also exhibits the lowest error across all error assessment criteria. Tables 6 and 7 provide accuracy data for the predictions made by all the methods, allowing for a comprehensive assessment of their accuracy. The statistics clearly demonstrate that the load capacity factor prediction accuracy for CLSSA is superior compared to other techniques. The CLSSA model achieves a training accuracy of 0.981327 and a testing accuracy of 0.97365, indicating that the CLSSA optimizer produces an optimal prediction outcome. The tables further highlight the advantages of using CLSSA.

Fig. 11 presents scatter plots for all compared models (CLSSA-ELM, HHO-ELM, GWO-ELM, EDO-ELM, SSA-ELM, CSO-ELM, and ELM), offering critical insights into the performance and predictive accuracy of the evaluated models. In this plot, when the circles representing different data points are tightly clustered near the best-fit line, it indicates high model accuracy, the closer the points are to the line, the higher the model's precision. An R^2 value of one signifies that the model has perfectly evaluated all data points without any errors. The illustration shows that for every model in the experiment, the data points are closely grouped around the best-fit line in both the training and testing data, leading to satisfactory outcomes for all models. The performance of the hybrid models is assessed by observing their corresponding R^2 scores. The CLSSA model demonstrates an R^2 value of 0.977 for overall load capacity factor predictions, which is the highest among the analyzed optimizers. This indicates that the CLSSA optimizer performs exceptionally well compared to other hybrid models. After carefully analyzing all the model plots, it is evident that the CLSSA plot stands out as the most accurate, while the SSA plot is the second best.

The proper alignment of data points around the red line in these graphs clearly indicates the potential efficacy of these models. Fig. 12 displays a line graph comparing the actual and projected values of all machine learning methods CLSSA-ELM, HHO-ELM, GWO-



(a) Biomass Energy (b) Coal (c) GDP

(d) Load Capacity Factor (e) Natural Resources (f) Political Risk

(g) Technological Innovation

Fig. 9. Trend of features in the dataset over time.

Table 5
Evaluation metric definition.

Metric	Formula	Definition
R^2	$\dfrac{\sum_{i=1}^{N}\left(Y_i^{Exp}-\overline{Y}^{Exp}\right)^2-\sum_{i=1}^{N}\left(Y_i^{Exp}-Y_i^{ELM}\right)^2}{\sum_{i=1}^{N}\left(Y_i^{Exp}-\overline{Y}^{Exp}\right)^2}$	Coefficient of Determination
RMSE	$\sqrt{\Sigma\left(Y_i^{ELM}-Y_i^{Exp}\right)^2/n}$	Root Mean Square Error
MSE	$\frac{1}{N}\sum_{i=1}^{N}\left(Y_i^{ELM}-Y_i^{ExP}\right)^2$	Mean square error
MAE	$\frac{1}{N}\sum_{i=1}^{n}\left\|Y_i^{Exp}-Y_i^{ELM}\right\|$	Mean absolute error
NRMSE	$\dfrac{\sum\left(Y_i^{ELM}-Y_i^{Exp}\right)^2}{\sum\left(Y_i^{Exp}\right)^2}$	Normalized Root Mean Square Error

Table 6
Evaluation metrics results from training hybrid models and ELM.

Model	R^2	RMSE	MSE	MAE	NRMSE
CLSSA-ELM	0.981327	0.040861	0.00167	0.032055	0.137974
HHO-ELM	0.944885	0.070201	0.004928	0.053174	0.24161
GWO-ELM	0.954351	0.063889	0.004082	0.053555	0.218482
EDO-ELM	0.962372	0.058005	0.003365	0.044324	0.197736
SSA-ELM	0.959293	0.060949	0.003715	0.047102	0.205909
CSO-ELM	0.95465	0.063679	0.004055	0.049935	0.218335
ELM	0.936372	0.075428	0.005689	0.060394	0.260675

ELM, EDO-ELM, SSA-ELM, CSO-ELM, and ELM) used in this study, along with a line representing their absolute errors. The plots are divided into training and testing data. The spikes represented by the pink line indicate the magnitude of prediction errors, thereby reflecting the effectiveness of the approaches used. Less amplitude on the pink line signifies minimal forecasting errors, indicating excellent performance. The CLSSA approach stands out in these plots, with few instances of high amplitudes on the error lines compared to others, demonstrating its effectiveness in understanding and connecting complex input variables to the target variable. The predictive capabilities of the CLSSA approach are impressive. In conclusion, the amplitude provides valuable insight into the disparity between predicted and actual values of the hybrid model patterns in both the training and testing datasets.

Table 7
Evaluation metrics results from testing hybrid models and ELM.

Model	R^2	RMSE	MSE	MAE	NRMSE
CLSSA-ELM	0.97365	0.051317	0.002633	0.037183	0.160731
HHO-ELM	0.948236	0.071926	0.005173	0.060799	0.228706
GWO-ELM	0.95444	0.067478	0.004553	0.057669	0.214661
EDO-ELM	0.936234	0.07983	0.006373	0.064645	0.251904
SSA-ELM	0.95798	0.063012	0.003971	0.049383	0.210067
CSO-ELM	0.96217	0.061488	0.003781	0.048743	0.19581
ELM	0.947919	0.072146	0.005205	0.05884	0.233735

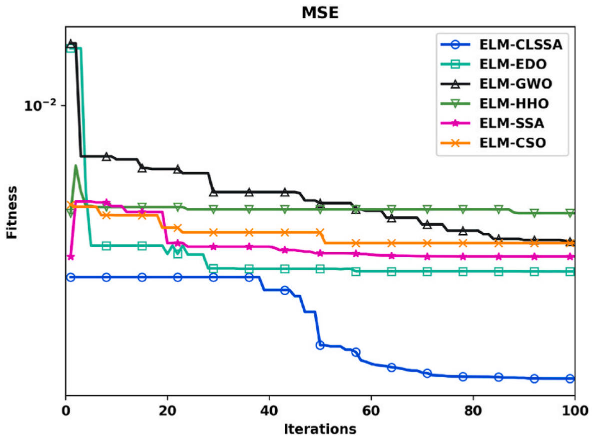


Fig. 10. Error convergence Curve of hybrid models.

4. Contribution and observation

Permutation importance is a technique used to evaluate the significance of features in a machine-learning model. It measures the change in the model's performance when the values of a single feature are randomly shuffled. This change in performance indicates the impact of each feature on predicting the target variable. However, permutation importance does not indicate whether a feature has a positive or negative effect on the target variable, it only measures how much the model's performance changes when the feature is permuted. For instance, if permuting a feature leads to a significant drop in performance, it suggests that the feature is important, but it does not reveal whether the feature is positively or negatively correlated with the target variable. The permutation feature importance scores derived from the model provide a detailed understanding of the factors influencing the load capacity prediction factor in this section in Fig. 13, with the length of each bar indicating their relative impact.

Notably, coal has the highest importance score, highlighting its significant impact on the load capacity factor in this study. Despite the global shift towards cleaner energy sources, coal remains a critical component in many regions, such as South Africa, because of mining or consumption due to its reliability and established infrastructure. This underscores the necessity of balancing environmental concerns with the practicalities of existing energy systems. This finding aligns with other studies showing that coal mining is crucial for South Africa's economy, significantly contributing to GDP, employment, and export revenues. However, it also poses substantial environmental challenges [37]. Following coal, economic growth ranks as the second most impactful factor. Economic growth significantly influences the load capacity factor by driving energy demand and enabling investments in energy infrastructure and technology. As economies expand, energy consumption increases, necessitating higher load capacities. Thus, fostering economic growth can have a direct effect on the energy sector's capacity, reinforcing the interplay between economic growth and energy capacity [38]. Technological innovation ranked third in importance. Innovations in areas such as smart grids, renewable energy technologies, and overall energy efficiency are crucial for enhancing energy production, storage, and distribution capacities. These advancements directly impact load capacity, aligning with previous studies by Refs. [39,40].

Biomass energy, with a moderate impact ranking fourth, contributes to the load capacity factor in the dataset by diversifying the energy mix and enhancing energy security. However, its relatively lower impact compared to other factors may be due to scalability challenges and limited resource availability. Nonetheless, biomass remains an important component of a balanced and sustainable energy strategy [41]. Political risk is another significant factor affecting load capacity. The stability and predictability of political environments, along with robust regulatory frameworks and governmental support, are crucial for fostering investment and development in the energy sector. High political risk can create an uncertain investment climate, reducing capacity, whereas stable and supportive political conditions can facilitate higher energy production capabilities. This underscores the critical role of governance and policy-making in securing and enhancing energy infrastructure. Natural resources hold the least importance in this study, although the

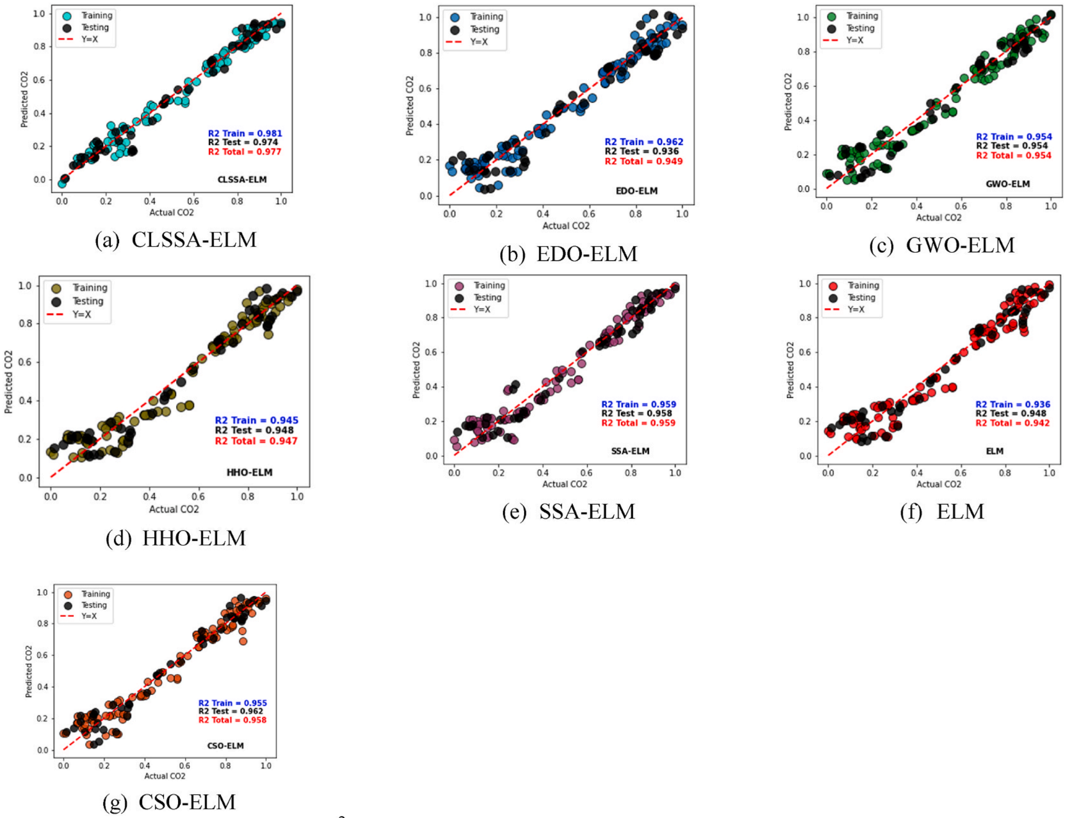


Fig. 11. Actual vs Predicted R^2 Plots of CLSSA-ELM and the Compared Models.

fundamental role of resource availability and quality, such as water, minerals, and fossil fuels, in sustaining high load capacities. Regions rich in natural resources can support more substantial energy production, directly influencing load capacity.

In conclusion, the permutation feature importance scores highlight coal energy, economic growth, technological innovation, and biomass as the top critical factors influencing load capacity factor. These insights provide valuable guidance for policymakers, investors, and energy companies. Policymakers can formulate strategies that balance environmental goals with practical energy needs, investors can prioritize areas with high potential for innovation and resource availability, and energy companies can develop strategic plans to ensure reliable and sustainable energy supplies. This comprehensive analysis enhances the understanding of the factors driving load capacity factor and supports efforts to optimize energy systems for sustainable development. Practical applications of the knowledge gained from the importance score are as follows.

1. Policy Formulation: Governments can focus on creating policies that promote technological advancements and political stability to maximize load capacity.
2. Investment Decisions: Investors can use these insights to prioritize funding in areas with high potential for technological innovation that can promote sustainable energy production.
3. Energy Planning: Energy companies can leverage this information to develop long-term strategies that ensure a reliable and sustainable energy supply.

5. Uncertainties in load capacity prediction

This section provides a detailed exploration of both internal and external uncertainties, with an emphasis on how these factors impact the outcomes of the hybrid CLSSA-ELM framework.

1. Internal Uncertainties: These uncertainties arise from within the model itself, particularly from the inherent variability in its parameters, such as the weights and biases in the ELM. The random initialization of these parameters can lead to different optimization performances, resulting in variations in prediction accuracy. Such internal uncertainties underscore the importance of employing optimization techniques, like CLSSA, which can stabilize the model by optimizing these parameters effectively.
2. External Uncertainties: External uncertainties originate from factors outside the control of the predictive model. These include fluctuations in environmental variables, economic changes, and policy shifts that can influence the load capacity factors. Since

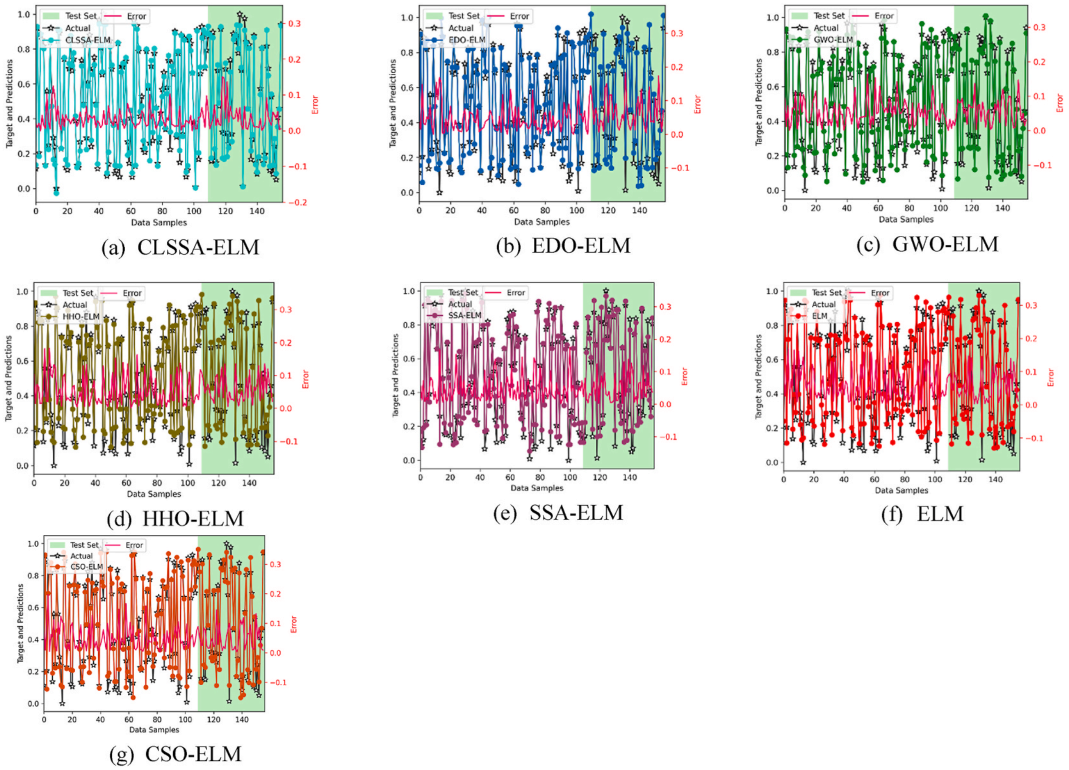


Fig. 12. Actual vs Predicted Absolute Error Plots of CLSSA-ELM and the Compared Models.

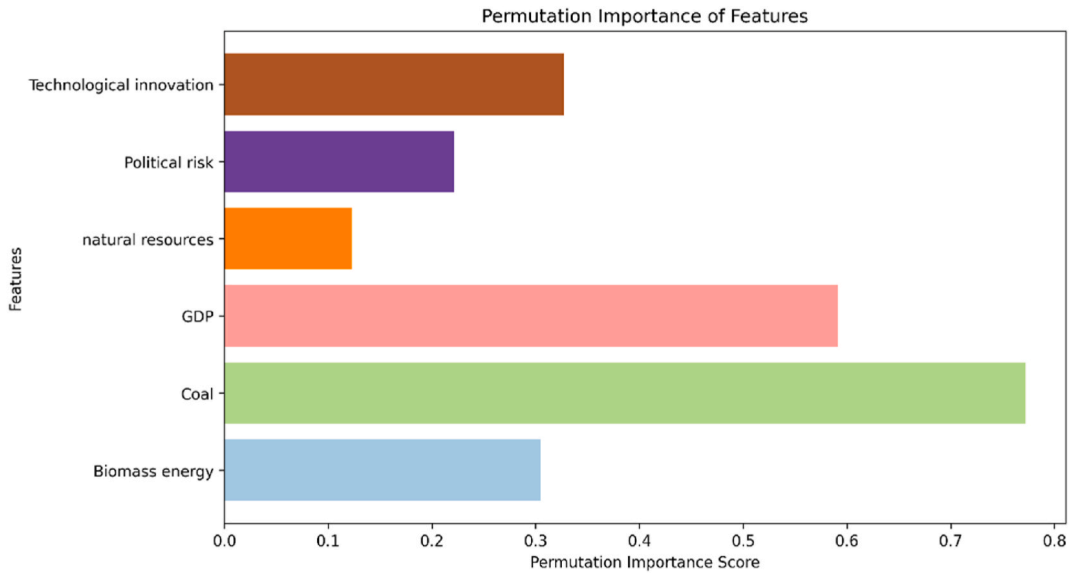


Fig. 13. Permutation importance score.

these variables are often dynamic and unpredictable, they introduce significant variability in the predictions made by the model. A well-designed model must account for these variabilities and complexities in the data set by ensuring the model maintains a robust optimization strategy during the prediction process, as highlighted in the improvement made to the CLSSA algorithm.

3. Parametric Uncertainties: These uncertainties stem from variability in the parameters of the model itself. For instance, the performance of the CLSSA in optimizing the ELM is directly influenced by the variability in its search for optimal weights and biases. When these parameters are uncertain or vary greatly, they introduce risks in the prediction outputs, necessitating a robust and

adaptive optimization approach to mitigate these risks. Effective parameter tuning, such as that provided by CLSSA, is crucial for handling this category of uncertainty.

4. Non-parametric Uncertainties: These uncertainties are not related to the model parameters but rather to the data used for training and testing. Examples include noise in the input data, missing values, and incomplete datasets. Such uncertainties can degrade the model's performance and limit its generalizability. The structure of these uncertainties plays a key role in model reliability, and understanding the nature of the data used (e.g., its variability, complexity, and noise) is vital for developing a robust prediction model.

Uncertainty, both parametric and non-parametric, significantly affects the accuracy and reliability of load capacity predictions. Variations in model parameters, coupled with external changes, can lead to fluctuating predictions, making it difficult to achieve consistently accurate results. The structure of uncertainty, whether originating from noisy data or volatile environmental conditions, can further complicate the prediction process. Understanding and addressing these uncertainties are essential for ensuring that the model delivers reliable forecasts, thereby supporting better decision-making for policymakers and energy planners.

6. Conclusion

The forecasting of load capacity factor has garnered significant attention, leading to the development of a strong study field largely due to its potential to have a detrimental effect on the ecosystem. This work aims to forecast the load capacity factor by using an improved machine learning methodology called ELM hybridized CLSSA. Within this specific framework, the ELM is selected as the primary predictive model for analyzing training data and identifying connections among different load capacity factor parameters. In order to improve the effectiveness of the ELM, a hybridization technique is employed that combines CLSSA with ELM. CLSSA is the fusion of two optimization algorithm, the traditional CSO exploitation phase is improved using the SSA exploitation strategy to improve convergence accuracy. The integration enables efficient adjustment and optimization of the parameters, particularly the weights and biases, for the machine learning model. Subsequently, a comprehensive assessment is conducted to evaluate the effectiveness of the ELM-CLSSA framework. The study's findings reveal that the ELM-CLSSA hybrid framework has remarkable achievements, achieving the highest levels of accuracy, and efficiency in this study. The ELM-CLSSA achieves impressive overall R^2 values, especially achieving 0.977 for load capacity factor prediction.

The metrics of the ELM-CLSSA approach transcend those of other methods, indicating that it has a higher level of accuracy in prediction compared to its rivals. According to the ELM-CLSSA feature significance assessment, coal energy, economic growth, technical innovation, and biomass are identified as the primary factors that contribute the most to the load capacity factor. The effectiveness and feasibility of the ELM-CLSSA approach can be limited by a lack of access to comprehensive and reliable data. Moreover, in terms of limitation, more analysis is required to ascertain the feasibility of implementing the suggested framework in other geographical areas and weather conditions. Finally, it is crucial to acknowledge the resource demands and computational intricacy of the model. Learning and procedures for optimization need substantial computational resources and time. In order to enhance the precision and robustness of prediction models, forthcoming research will focus on broadening the range of data collection to encompass bigger and more diverse datasets from other climatic regions. To enhance understanding and optimize the manner of formulating environmentally beneficial policies, we will also explore approaches to elucidating and deciphering the estimations provided by ELM-CLSSA.

CRediT authorship contribution statement

Nuriddin Tahir S Luoka: Methodology, Formal analysis, Conceptualization. **Wagdi M.S. Khalifa:** Writing – review & editing, Validation, Methodology, Investigation, Formal analysis.

Data availability statement

Data used for the study as being provided in the study.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] R.M. Petrović, N. Kocić, R.B. Stojanović, The importance of renewable energy sources for sustainable development, Econ. Sustain. Dev. 4 (2) (2020) 15–24, https://doi.org/10.5937/ESD2002016P.
[2] C. Heinze, et al., The quiet crossing of ocean tipping points, Proc. Natl. Acad. Sci. 118 (9) (Mar. 2021) e2008478118, https://doi.org/10.1073/pnas.2008478118.
[3] N. Wunderling, J.F. Donges, J. Kurths, R. Winkelmann, Interacting tipping elements increase risk of climate domino effects under global warming, Earth Syst. Dyn. 12 (2) (Jun. 2021) 601–619, https://doi.org/10.5194/esd-12-601-2021.
[4] The CAT Thermometer." Accessed: June. 13, 2024. [Online]. Available: https://climateactiontracker.org/global/cat-thermometer/.

[5] D. Pandey, M. Agrawal, J.S. Pandey, Carbon footprint: current methods of estimation, Environ. Monit. Assess. 178 (1) (Jul. 2011) 135–160, https://doi.org/10.1007/s10661-010-1678-y.
[6] H.S. Matthews, C.T. Hendrickson, C.L. Weber, The importance of carbon footprint estimation boundaries, Environ. Sci. Technol. 42 (16) (Aug. 2008) 5839–5842, https://doi.org/10.1021/es703112w.
[7] R. Siche, L. Pereira, F. Agostinho, E. Ortega, Convergence of ecological footprint and emergy analysis as a sustainability indicator of countries: Peru as case study, Commun. Nonlinear Sci. Numer. Simul. 15 (10) (Oct. 2010) 3182–3192, https://doi.org/10.1016/j.cnsns.2009.10.027.
[8] U.K. Pata, D. Balsalobre-Lorente, Exploring the impact of tourism and energy consumption on the load capacity factor in Turkey: a novel dynamic ARDL approach, Environ. Sci. Pollut. Res. 29 (9) (Feb. 2022) 13491–13503, https://doi.org/10.1007/s11356-021-16675-4.
[9] C. Liu, B. Sun, C. Zhang, F. Li, A hybrid prediction model for residential electricity consumption using holt-winters and extreme learning machine, Appl. Energy 275 (Oct. 2020) 115383, https://doi.org/10.1016/j.apenergy.2020.115383.
[10] J. Wang, S. Lu, S.-H. Wang, Y.-D. Zhang, A review on extreme learning machine, Multimed. Tools Appl. 81 (29) (Dec. 2022) 41611–41660, https://doi.org/10.1007/s11042-021-11007-7.
[11] W. Sun, J. Sun, Prediction of carbon dioxide emissions based on principal component analysis with regularized extreme learning machine: the case of China, Environ. Eng. Res. 22 (3) (Sep. 2017) 302–311, https://doi.org/10.4491/eer.2016.153.
[12] C. Wan, Z. Xu, P. Pinson, Z.Y. Dong, K.P. Wong, Probabilistic forecasting of wind power generation using extreme learning machine, IEEE Trans. Power Syst. 29 (3) (May 2014) 1033–1044, https://doi.org/10.1109/TPWRS.2013.2287871.
[13] A.R. Lima, A.J. Cannon, W.W. Hsieh, Forecasting daily streamflow using online sequential extreme learning machines, J. Hydrol. 537 (Jun. 2016) 431–443, https://doi.org/10.1016/j.jhydrol.2016.03.017.
[14] Md A. Rizvee, A.R. Arju, Md Al-Hasan, S.M. Tareque, Md Z. Hasan, Weather forecasting for the north-western region of Bangladesh: a machine learning approach, in: 2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT), Jul. 2020, pp. 1–6, https://doi.org/10.1109/ICCCNT49239.2020.9225389.
[15] X. Li, et al., Empirical analysis: stock market prediction via extreme learning machine, Neural Comput. Appl. 27 (1) (Jan. 2016) 67–78, https://doi.org/10.1007/s00521-014-1550-z.
[16] J. Zhang, F. Xu, Y. Zhang, T. Shen, ELM-based driver torque demand prediction and real-time optimal energy management strategy for HEVs, Neural Comput. Appl. 32 (18) (Sep. 2020) 14411–14429, https://doi.org/10.1007/s00521-019-04240-7.
[17] N. Priyadharshini, S. N, B. Hemalatha, C. Sureshkumar, A novel hybrid Extreme Learning Machine and Teaching–Learning-Based Optimization algorithm for skin cancer detection, Healthc. Anal. 3 (Nov. 2023) 100161, https://doi.org/10.1016/j.health.2023.100161.
[18] X. Wang, J. Li, L. Shao, H. Liu, L. Ren, L. Zhu, Short-term wind power prediction by an extreme learning machine based on an improved hunter–prey optimization algorithm, Sustainability 15 (2) (Jan. 2023), https://doi.org/10.3390/su15020991. Art. no. 2.
[19] D. El Bourakadi, H. Ramadan, A. Yahyaouy, J. Boumhidi, A novel solar power prediction model based on stacked BiLSTM deep learning and improved extreme learning machine, Int. J. Inf. Technol. 15 (2) (Feb. 2023) 587–594, https://doi.org/10.1007/s41870-022-01118-1.
[20] Z. Wang, Y. Liu, T. Wang, D. Gong, D. Zhang, Prediction model of hot strip crown based on industrial data and hybrid the PCA-SDWPSO-ELM approach, Soft Comput. 27 (17) (Sep. 2023) 12483–12499, https://doi.org/10.1007/s00500-023-07895-6.
[21] B. Zhu, Y. Feng, D. Gong, S. Jiang, L. Zhao, N. Cui, Hybrid particle swarm optimization with extreme learning machine for daily reference evapotranspiration prediction from limited climatic data, Comput. Electron. Agric. 173 (Jun. 2020) 105430, https://doi.org/10.1016/j.compag.2020.105430.
[22] N.K. Anushkannan, G.H. Balde, D. Suganthi, P.M. Pandian, B. Kaur, K.V.D. Sagar, A novel method for categorizing brain tumors using the hybrid ALO-ELM model, in: 2023 7th International Conference on Trends in Electronics and Informatics (ICOEI), Apr. 2023, pp. 1467–1472, https://doi.org/10.1109/ICOEI56765.2023.10125907.
[23] Z.-F. Liu, L.-L. Li, M.-L. Tseng, M.K. Lim, Prediction short-term photovoltaic power using improved chicken swarm optimizer - extreme learning machine model, J. Clean. Prod. 248 (Mar. 2020) 119272, https://doi.org/10.1016/j.jclepro.2019.119272.
[24] M. Qaraad, A. Aljadania, M. Elhosseini, Large-scale competitive learning-based salp swarm for global optimization and solving constrained mechanical and engineering design problems, Mathematics 11 (6) (Jan. 2023) 6, https://doi.org/10.3390/math11061362.
[25] F. Han, H.-F. Yao, Q.-H. Ling, An improved evolutionary extreme learning machine based on particle swarm optimization, Neurocomputing 116 (Sep. 2013) 87–93, https://doi.org/10.1016/j.neucom.2011.12.062.
[26] S. Mirjalili, A.H. Gandomi, S.Z. Mirjalili, S. Saremi, H. Faris, S.M. Mirjalili, Salp Swarm Algorithm: a bio-inspired optimizer for engineering design problems, Adv. Eng. Softw. 114 (Dec. 2017) 163–191, https://doi.org/10.1016/j.advengsoft.2017.07.002.
[27] Ran Cheng, Yaochu Jin, A competitive swarm optimizer for large scale optimization, IEEE Trans. Cybern. 45 (2) (Feb. 2015) 191–204, https://doi.org/10.1109/TCYB.2014.2322602.
[28] M.-C. Yuen, S.-C. Ng, M.-F. Leung, A competitive mechanism multi-objective particle swarm optimization algorithm and its application to signalized traffic problem, Cybern. Syst. 52 (1) (Oct. 2020) 73–104, https://doi.org/10.1080/01969722.2020.1827795.
[29] G.-B. Huang, Q.-Y. Zhu, C.-K. Siew, Extreme learning machine: a new learning scheme of feedforward neural networks, in: 2004 IEEE International Joint Conference on Neural Networks (IEEE Cat. No.04CH37541), vol. 2, Jul. 2004, pp. 985–990, https://doi.org/10.1109/IJCNN.2004.1380068.
[30] M. Li, K. Li, Q. Qin, A rockburst prediction model based on extreme learning machine with improved Harris Hawks optimization and its application, Tunn. Undergr. Space Technol. 134 (Apr. 2023) 104978, https://doi.org/10.1016/j.tust.2022.104978.
[31] Anita, A. Yadav, AEFA: artificial electric field algorithm for global optimization, Swarm Evol. Comput. 48 (Aug. 2019) 93–108, https://doi.org/10.1016/j.swevo.2019.03.013.
[32] M. Abdel-Basset, D. El-Shahat, M. Jameel, M. Abouhawwash, Exponential distribution optimizer (EDO): a novel math-inspired algorithm for global optimization and engineering problems, Artif. Intell. Rev. 56 (9) (Sep. 2023) 9329–9400, https://doi.org/10.1007/s10462-023-10403-9.
[33] S. Mirjalili, S.M. Mirjalili, A. Lewis, Grey Wolf optimizer, Adv. Eng. Softw. 69 (Mar. 2014) 46–61, https://doi.org/10.1016/j.advengsoft.2013.12.007.
[34] A.A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, H. Chen, Harris hawks optimization: algorithm and applications, Future Gener. Comput. Syst. 97 (Aug. 2019) 849–872, https://doi.org/10.1016/j.future.2019.02.028.
[35] S. Mirjalili, A.H. Gandomi, S.Z. Mirjalili, S. Saremi, H. Faris, S.M. Mirjalili, Salp Swarm Algorithm: a bio-inspired optimizer for engineering design problems, Adv. Eng. Softw. 114 (Dec. 2017) 163–191, https://doi.org/10.1016/j.advengsoft.2017.07.002.
[36] O. Tutsoy, D.E. Barkana, K. Balikci, A novel exploration-exploitation-based adaptive law for intelligent model-free control approaches, IEEE Trans. Cybern. 53 (1) (Jan. 2023) 329–337, https://doi.org/10.1109/TCYB.2021.3091680.
[37] A.S. Hassan, Coal mining and environmental sustainability in South Africa: do institutions matter? Environ. Sci. Pollut. Res. 30 (8) (Feb. 2023) 20431–20449, https://doi.org/10.1007/s11356-022-23585-6.
[38] O.A. Olanrewaju, Analysing impacts responsible for South Africa's energy consumption: LMDI application, in: 2019 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM), Dec. 2019, pp. 1593–1596, https://doi.org/10.1109/IEEM44572.2019.8978872.
[39] A.A. Awosusi, et al., A roadmap toward achieving sustainable environment: evaluating the impact of technological innovation and globalization on load capacity factor, Int. J. Environ. Res. Public. Health 19 (6) (Jan. 2022) 6, https://doi.org/10.3390/ijerph19063288.
[40] X. Liu, V.O. Olanrewaju, E.B. Agyekum, M.F. El-Naggar, M.M. Alrashed, S. Kamel, Determinants of load capacity factor in an emerging economy: the role of green energy consumption and technological innovation, Front. Environ. Sci. 10 (Oct. 2022), https://doi.org/10.3389/fenvs.2022.1028161.
[41] H. Visser, G.A. Thopil, A. Brent, Life cycle cost profitability of biomass power plants in South Africa within the international context, Renew. Energy 139 (Aug. 2019) 9–21, https://doi.org/10.1016/j.renene.2019.02.080.