Journal of Ambient Intelligence and Humanized Computing (2023) 14:10473–10491
https://doi.org/10.1007/s12652-022-03703-5

ORIGINAL RESEARCH

A new elite opposite sparrow search algorithm-based optimized LightGBM approach for fault diagnosis

Qicheng Fang[1,2] · Bo Shen[1,2] · Jiankai Xue[1,2]

Received: 12 June 2021 / Accepted: 10 January 2022 / Published online: 23 January 2022

Abstract
In this paper, a new fault diagnosis approach based on elite opposite sparrow search algorithm (EOSSA) optimized LightGBM is proposed. It is necessary to extract appropriate features when dealing with high-dimensional data. Since the distribution of the high-dimensional data is not always approximately subject to a normal distribution, it will cause errors when it is approximated to normal distribution for feature extraction. The dimension reduction algorithms based on Euclidean distance often ignore the change of data distribution. To address this problem, cam locally linear discriminate embedding (CLLDE) based on cam weighted distance is proposed, which can improve the performance dealing with the deformed data of locally linear discriminate embedding (LLDE). The performance of CLLDE is better than LLDE on the iris dataset. It is important to establish a classifier with optimized hyper-parameters for fault identification. Sparrow search algorithm (SSA) is a novel optimization algorithm, which has achieved good results in many applications, but its optimization ability and convergence speed still need to be improved. Elite opposite sparrow search algorithm (EOSSA) is proposed by introducing elite opposite learning strategy and orifice imaging opposite learning strategy into SSA. The optimization results on benchmark functions show that EOSSA converges faster and has better optimization ability compared with the other five algorithms. EOSSA is used to optimize the hyper-parameters of LightGBM to train a classifier that can obtain a better fault recognition rate. Finally, the effectiveness of the proposed fault diagnosis approach is verified on Tennessee Eastman (TE) process dataset. Experiment results demonstrate that the EOSSA-LightGBM-based approach is superior to other algorithms.

Keywords Fault diagnosis · Feature extraction · High-dimensional data · Intelligent optimization · Opposite learning strategy

1 Introduction

Fault diagnosis and detection are mostly data-driven nowadays. Although data acquisition becomes easier, how to deal with a large amount of high-dimensional data has become a difficult problem. There is a lot of information that can be used in high-dimensional data, but most of them are redundant, and a large amount of data increases the computational complexity, which may lead to the curse of dimensionality.

A common feature extraction method to deal with high-dimensional data is dimension reduction. The traditional linear dimension reduction methods include principal component analysis (PCA) (Duchene and Leclercq 1988; Wold et al 1987), linear discriminant analysis (LDA) (Duda et al 2001; Etemad and Chellapa 1997) and others. Later, Schölkopf et al. (1997) proposed kernel principal component analysis (KPCA) to deal with nonlinear data, which achieved good results. With the further development of manifold learning, there is a better choice to deal with nonlinear data.

Manifold learning is mainly divided into local information preserving methods and global information preserving methods. Locally linear embedding (LLE) (Roweis and Saul 2000) is a classical manifold learning method based on local information, which can generate an implicit function to map data from high-dimensional space to low-dimensional space. Nonlinear methods for preserving global information mainly include multi-dimensional scaling (MDS) (Cox and Cox

✉ Bo Shen
bo.shen@dhu.edu.cn

[1] College of Information Science and Technology, Donghua University, Shanghai, China

[2] Engineering Research Center of Digitalized Textile and Fashion Technology, Ministry of Education, Shanghai, China

1994), isometric mapping (ISOMAP) (Law and Jain 2006; Tenenbaum et al 2000) and others. ISOMAP proves that it is feasible and effective to change the measurement distance between data points. Li et al. (2008) combines maximum margin criterion (MMC) (Li et al 2006) with LLE to propose locally linear discriminant embedding (LLDE). LLDE uses the category attributes of data, and it makes the data of different categories farther apart and similar data closer to each other after dimension reduction. Besides, the algorithm avoids the small sample size (SSS) problem (Zheng et al 2004).

As mentioned above, using category attributes is more conducive to classification. Because of the attraction or repulsion between data samples (Zhou and Chen 2006), the distribution of data may be deformed. Cam weighted distance considers the scale of data distribution and the direction of deformation, which can well measure deformed data. When building a neighborhood with cam weighted distance, the samples with larger density will be given smaller weight and larger weight for the smaller ones (Zhou and Chen 2006). The fault data should be paid more attention in fault diagnosis, while cam weighted distance can give more attention to fault data. Owing to the way that LLDE calculates distance, LLDE has some limitations when dealing with deformed data. To address this problem, cam locally linear discriminant embedding (CLLDE) is proposed based on cam weighted distance.

It is necessary to classify the extracted data to realize fault identification after feature extraction. Zhang et al. (2018) used robust LLE to extract features, then used support vector machine (SVM) to identify mechanical faults. Fan et al. (2019) used convolutional neural network (CNN) for feature extraction, and then used LightGBM for mechanical fault prediction. LightGBM (Ke et al 2017) is a gradient lifting algorithm based on the decision tree proposed by Microsoft. In this algorithm, the histogram is introduced to generate the decision tree. In the sampling process, gradient-based one side sampling (GOSS) and exclusive feature bundling (EFB) are used to merge independent features. In addition, the enhancement to the leaf-wise growth strategy improved the computation efficiency of the algorithm.

The influence of hyper-parameters should be considered when training a classifier. LightGBM has some hyper-parameters that determine the final classification effect. Thus, it is important to select the appropriate hyper-parameters for the classifier.

Swarm intelligence optimization algorithm is widely applied in engineering and plays an important role in the adjustment of model parameters. Zeng et al. (2017) applied the improved particle swarm optimization (PSO) to address the short-term load forecasting problem. Liu et al. (2019) used an improved PSO and *K*-means algorithm to solve the clustering problem of the emergency patient. You et al. (2018) use BP neural network optimized by hybrid PSO to study the electro-hydraulic control system. Zeng et al. (2018) proposed a switching delayed PSO to optimize the hyper-parameters of SVM to implement medical diagnosis. Pathana et al. (2021) show that an optimization algorithm to optimize CNN to identify COVID-19 patients by X-ray image of the lung.

Sparrow search algorithm (SSA) (Xue and Shen 2020) is a novel swarm intelligence algorithm, which has been applied in many fields and achieved good results, such as (Xing et al 2021; Zhang and Ding 2021; Zhu and Yousefi 2021). Xing et al. (2021) apply SSA in fault diagnosis of the wheelset-bearing system. Zhang et al. (2021) show that the chaotic mapping strategy to improve SSA to optimize the stochastic configuration network. Zhu et al. (2021) proposed an adaptive SSA to address the parameters identification problem. However, the search efficiency and the convergence speed of SSA are needed to be improved. To improve the ability of the algorithm to find the optimal solution and accelerate the convergence speed of the algorithm, the elite opposite sparrow search algorithm (EOSSA) is proposed based on the elite opposite learning strategy and orifice imaging opposite learning strategy. EOSSA is used to optimize LightGBM to find the most suitable hyper-parameters to identify fault data. The diagram of the proposed fault diagnosis approach is shown in Fig. 1.

The main contributions of this paper are summarized as follows. CLLDE algorithm is proposed based on cam weighted distance to improve the limitations when dealing with the deformed data of LLDE. EOSSA is proposed

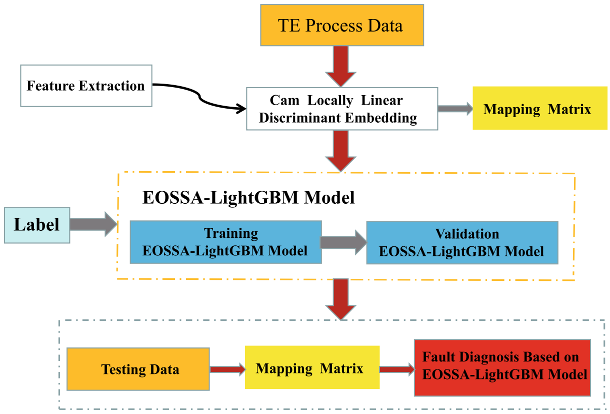

TE Process Data
Feature Extraction
Cam Locally Linear Discriminant Embedding
Mapping Matrix
EOSSA-LightGBM Model
Label
Training EOSSA-LightGBM Model
Validation EOSSA-LightGBM Model
Testing Data
Mapping Matrix
Fault Diagnosis Based on EOSSA-LightGBM Model

Fig. 1 Diagram of the proposed framework based on the EOSSA-LightGBM approach

based on elite opposite learning strategy and orifice imaging opposite learning strategy, which improves the ability to find the optimal solution and accelerate the convergence speed compared with SSA. SSA and EOSSA are used to optimize the hyper-parameters of LightGBM for the first time, and a new fault diagnosis approach based on EOSSA optimized LightGBM is proposed. Aimed at the problem that the fault data may not be approximated to normal distribution for similarity measurement. An approach based on CLLDE and EOSSA-LightGBM is proposed to address the diagnosis problem of the deformed fault data.

The rest of this paper is organized as follows. In Sect. 2, CLLDE is proposed based on cam weighted distance, and its effectiveness is verified on the iris dataset. In Sect. 3, EOSSA is proposed by introducing the elite opposite learning strategy and orifice imaging opposite learning strategy into SSA. The establishment process of EOSSA-LightGBM and the fault diagnosis experiment are introduced in Sect. 4. The conclusion of the whole paper is in Sect. 5.

2 Cam locally linear discriminate embedding

2.1 Cam weighted distance

Let d dimensional vector Z obey standard normal distribution $N(0, I)$, and its probability density function is

$$f(z) = \frac{1}{(2\pi)^{\frac{d}{2}}} \exp\left(-\frac{1}{2}z^T z\right). \tag{1}$$

Random vector X can be defined by the transformation (Zhou and Chen 2006)

$$X = \left(a + b\frac{z^T \tau}{z}\right)z. \tag{2}$$

where z denotes the original well distributed dataset. $a > b \geqslant 0$ reflects the scale of data distribution and the deformation direction in a certain direction. τ is a normalized vector that describes the deformation orientation. $||z|| = \sqrt{z^T z}$, X is the deformation distribution, and a and b are the deformation parameters (Pan et al 2009). The deformed data can be changed into normal distribution by transforming $z = X/(a + b\cos\theta)$, then the definition of cam weighted distance is obtained.

Let $x_0 \in R^d$ be the center of cam distribution $D_d(a, b, \tau)$. The cam weighted distance from point $x \in R^d$ to x_0 can be defined as

$$\mathrm{Dist}\left(x_0, x\right) = \frac{|x - x_0|}{a + b\frac{\left(x-x_0\right)^T \tau}{x-x_0}}. \tag{3}$$

Lemma 1 *(Zhou and Chen 2006) If a random vector $X = D_d(a, b, \tau)$, then $E(X) = c_1 b\tau$ and $E(\|x\|) = c_2 a$, where c_1 and c_2 are constants and can be expressed as*

$$c_1 = \sqrt{2} \times \frac{\Gamma\left(\frac{d+1}{2}\right)}{\Gamma\left(\frac{d}{2}\right)\cdot d}, \tag{4}$$

$$c_2 = \sqrt{2} \times \frac{\Gamma\left(\frac{d+1}{2}\right)}{\Gamma\left(\frac{d}{2}\right)}, \tag{5}$$

where d is the dimensionality of the random vector X. Γ represents Gamma function $\Gamma(m) = \int_0^\infty t^{m-1}e^{-t}dt(m > 0)$.

According to Lemma 1, for arbitrary x in dataset, we can suppose that it is the center of the cam distribution. The points around it are subject to cam distribution, and its k-nearest neighbors $X_i = [x_{i1}, x_{i2}, \ldots, x_{ik}]$ can be transformed as $V_i = [v_{i1}, v_{i2}, \ldots, v_{ij}]$, where $v_{ij} = x_{ij} - x_i$, $j = 1, 2, \ldots, k$. The central vector $\hat{G}_i$ and the average vector length $\hat{L}_i$ can be presented as follows:

$$\hat{G}_i = \frac{1}{k}\sum_{j=1}^{k} v_{ij}, \tag{6}$$

$$\hat{L}_i = \frac{1}{k}\sum_{j=1}^{k} \|v_{ij}\|. \tag{7}$$

$\hat{G}_i$ and the mean value of v_{ij} can be used to estimate $E(X)$. Besides, $\hat{L}_i$ and the mean value of $\|v_{ij}\|$ can be used to estimate $E(\|X\|)$. The parameter estimation can be obtained by Lemma 1:

$$\hat{a}_i = \frac{\hat{L}_i}{c_2}, \tag{8}$$

$$\hat{b}_i = \frac{\|G_i\|}{c_1}, \tag{9}$$

$$\hat{\tau}_i = \frac{\hat{G}_i}{\|G_i\|}. \tag{10}$$

2.2 Cam locally linear discriminate embedding

In practical applications, most high-dimensional data are not normally distributed. For fault data, it is often different from normal data in the process of acquisition due to attraction, repulsion, and other factors. These changes can not be ignored. Under this condition, it assumes that the data approximately is subject to the standard normal distribution will produce large errors. The original Euclidean distance calculation method based on normal distribution is no longer applicable. It can transform the deformed data into normal distribution to use cam weighted distance, which can eliminate the influence of data deformation and better describe the similarity (Zhou and Chen 2006). Considering this condition, cam weighted distance is used to improve the nearest neighbor distance calculation method of LLDE to make it more suitable for fault data diagnosis. Cam Locally Linear Discriminate Embedding (CLLDE) uses cam weighted distance to calculate the distance between the samples, and other calculation processes are similar with LLDE.

Changing the calculation method of the distance that used in LLDE between the nearest neighbor points can change the direction of the algorithm to select the nearest neighbor points. Instead of taking the nearest neighbor points as the selection criteria, the calculation method will consider the points in all directions of the center point to select the nearest neighbor points.

Based on the cam distance, combined with LLE and MMC, CLLDE is proposed to extract feature. LLE is a nonlinear dimension reduction method. Let $X = \{X_1, X_2, \dots, X_n\}$, and $X_i \in R^D$ denotes n points in D dimensional space, where $i = 1, 2, \dots, n$. The mapping of high-dimensional data in low-dimensional space can be recorded as $Y = \{Y_1, Y_2, \dots, Y_n\}$, $Y_i \in R^d$, where $i = 1, 2, \dots, n$. The d represents the dimensionality of low-dimensional space.

LLE achieves the goal through three steps (Lei et al 2010). Firstly, kNN algorithm or ε-ball criterion is used in LLE to find the nearest neighbors. LLE is to find the best reconstruction weight matrix in the second step, and to minimize the local reconstruction error of x_i can realize it. It can be described as

$$\phi(W) = \arg\min_{w} \sum_{i=1}^{N} \|x_i - \sum_{j=1}^{k} w_{ij} x_j\|^2, \tag{11}$$

where N is the amount of all data points, x_i denotes the i-th data point, x_j denotes the j-th nearest neighbor, and w_{ij} represents the weight coefficient of the i-th data point to the j-th nearest neighbor.

Let N_i^k be the set of neighbor points of x_i. The x_i is reconstructed by its neighbor points, if x_j is the neighbor point of x_i, then $w_{ij} \neq 0$; on the contrary, $w_{ij} = 0$. In addition, it should meet $\sum_{j=1}^{k} w_{ij} = 1$ to have a better data distribution after dimensionality reduction. The restriction of weight W can be written as

$$\begin{cases} w_{ij} = 0, & \text{if } x_j \notin N_{i=1}^k, \\ \sum_{j=1}^{k} w_{ij} = 1, & \text{if } x_j \in N_{i=1}^k. \end{cases} \tag{12}$$

In the third step, the reconstruction matrix W can be used to calculate the optimal embedding matrix Y after dimension reduction. This step can be described as

$$\begin{aligned} \phi(Y) &= \arg\min_{y} \sum_{i=1}^{N} \|y_i - \sum_{j=1}^{k} w_{ij} y_j\|^2 \\ &= \arg\min_{y} \operatorname{tr}\{Y^T (I - W)^T (I - W) Y\}. \end{aligned} \tag{13}$$

Let $M = (I - W)^T (I - W)$, and (13) can be transformed as $\arg\min_{y} \operatorname{tr}\{Y^T M Y\}$, where $M = \left[M_{ij}\right]_{n \times n}$. To avoid degenerate solutions, it constrains the embedding vectors to have unit covariance with which satisfy $\frac{1}{N} Y^T Y = I$. According to Rayleigh-Ritz theorem (Li et al 2008), the smallest nonzero eigenvector of the sparse matrix M is the solution of (13).

Li et al. (2008) linearize LLE by introducing $Y = V^T X$ into LLE to address this problem, and (13) can be rewritten as

$$\Phi(Y) = \arg\min_{y} \operatorname{tr}\{V^T X M X^T V\}. \tag{14}$$

The maximum margin criterion (MMC) restricts the mapping relationship, and its purpose is to maximize the distance between each class of samples. It can use the category attributes of data to improve the separability of data. S_w is the within-class scatter matrix. S_b is the between-class scatter matrix. They can be calculated by Li et al (2006)

$$S_w = \sum_{i=1}^{c} \sum_{j=1}^{n_i} \left(x_j^i - m_i\right)\left(x_j^i - m_i\right)^T, \tag{15}$$

$$S_b = \sum_{i=1}^{c} n_i \left(m_i - m\right)\left(m_i - m\right)^T, \tag{16}$$

where c is the number of classes, m is the mean vector of all the samples, m_i represents the average vector of the i-th class, n_i is the number of samples in the i-th class, and x_j^i is the j-th sample in the i-th class.

The objective function of MMC under projection matrix W can be described as

$$J = \mathrm{tr}\{W^T\left(S_b - S_w\right)W\}. \tag{17}$$

The class relationship between samples is considered in MMC. It can obtain the optimal linear discriminant mapping and maintain the internal geometric structure of neighbors. The linearized LLE obtains the manifold structure of high-dimensional data, which ensures the scalability of the data. To minimize the reconstruction error and maximize the distance between classes, a multi-objective optimization problem can be obtained by combining (14) and (17), and they have the same mapping matrix in this case. The multi-objective optimization problem can be represented as

$$\begin{cases} \min\ \mathrm{tr}\{V^T XMX^T V\}, \\ \max\ \mathrm{tr}\{V^T\left(S_b - S_w\right)V\}. \end{cases} \tag{18}$$

Since (14) and (17) have the same mapping matrix, a single objective optimization problem can be constructed by making a difference to solve the mapping matrix V. The single objective optimization problem can be written as

$$\min\ \mathrm{tr}\{V^T\left(XMX^T - \left(S_b - S_w\right)\right)V\}, \tag{19}$$

$$\text{s.t. } V^T XX^T V = I. \tag{20}$$

The problem can be solved by the Lagrange multiplier method, and the form of (19) can be transformed into

$$\left(XMX^T - \left(S_b - S_w\right)\right)V = \lambda XX^T V, \tag{21}$$

where λ_i is the generalized eigenvalue of $(XMX^T - (S_b - S_w))$ and XX^T. V is the corresponding generalized eigenvector.

Therefore, the minimum value of the objective function (19) can be required as long as the generalized eigenvector V of (21) is obtained. V is composed of eigenvectors corresponding to the first d smallest eigenvalues of the generalized spectral decomposition, where d is the target dimension to be reduced to.

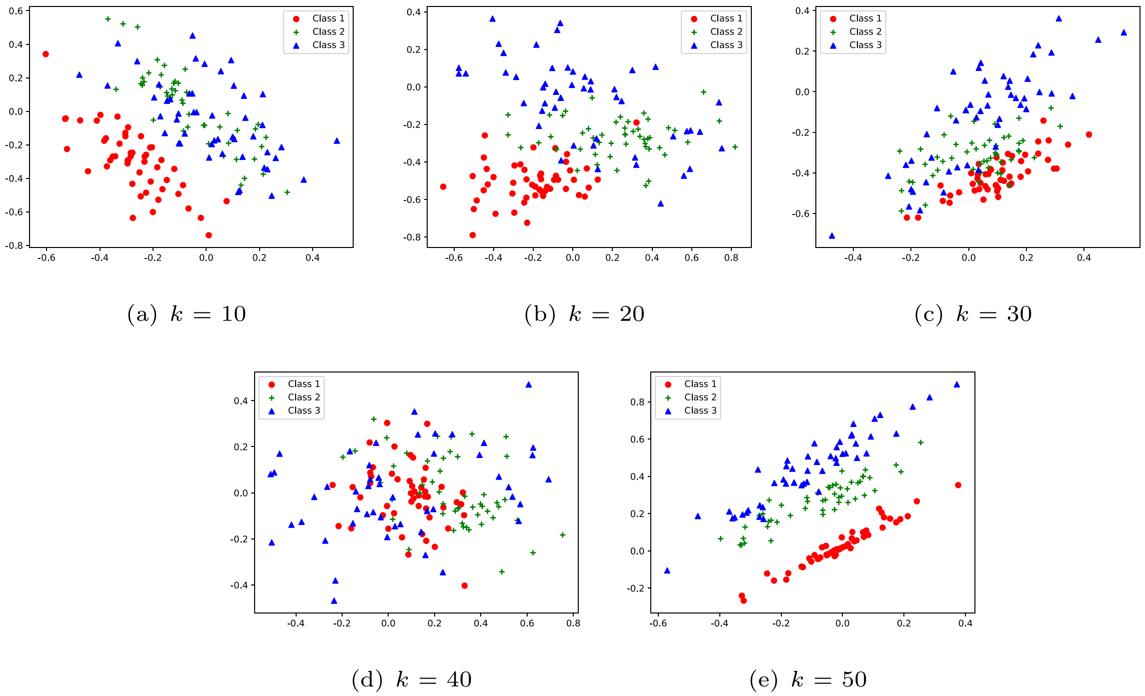
After changing the distance of selecting the nearest neighbor, mapping matrix Y can be obtained by (21) and $Y = V^T X$, which can better reflect the actual distribution of the deformed data. The main steps of CLLDE are summarized in Algorithm 1.

Algorithm 1 Cam Locally Linear Discriminate Embedding

```
Require: The number of data samples $n$, the number of nearest neighbor points $k$.
Ensure: Eigenvectors matrix $V$, low-dimensional embedding $Y$.
Calculate the $k$ nearest neighbor points by $k$NN algorithm;
for $i = 1 : n$
    for $j = 1 : k$
        Calculate $v_{ij}$ by $v_{ij} = x_{ij} - x_i$;
        Calculate $\hat{G}_i$ and $\hat{L}_i$ by (6) and (7);
        Calculate $a_i$, $b_i$, $\tau_i$ based on (8), (9) and (10);
    end for
end for
for $i = 1 : n$
    Calculate cam weighted distance from $x_i$ to $\forall x_j \in D$, $j \neq i$ by (3);
end for
while $j \neq i$ do
    if Dist$(x_j, x_i)$ < Dist$(x_k, x_i)$
        Select the $x_j$ as the neighbor of $x_i$;
    else
        Select the $x_k$ as the neighbor of $x_i$;
    end if
end while
Calculate $V$ and $Y$ by (14) and (17), and (21);
return $V$ and $Y$.
```

The next part is the performance experiment of CLLDE in comparison with LLDE. The integrated development environment is Python 3.7.6 for all the experimentations, the model of CPU is Intel(R) Core(TM) i7-8750H, and the operating system is Windows 10. The following experiments are in the same environment.

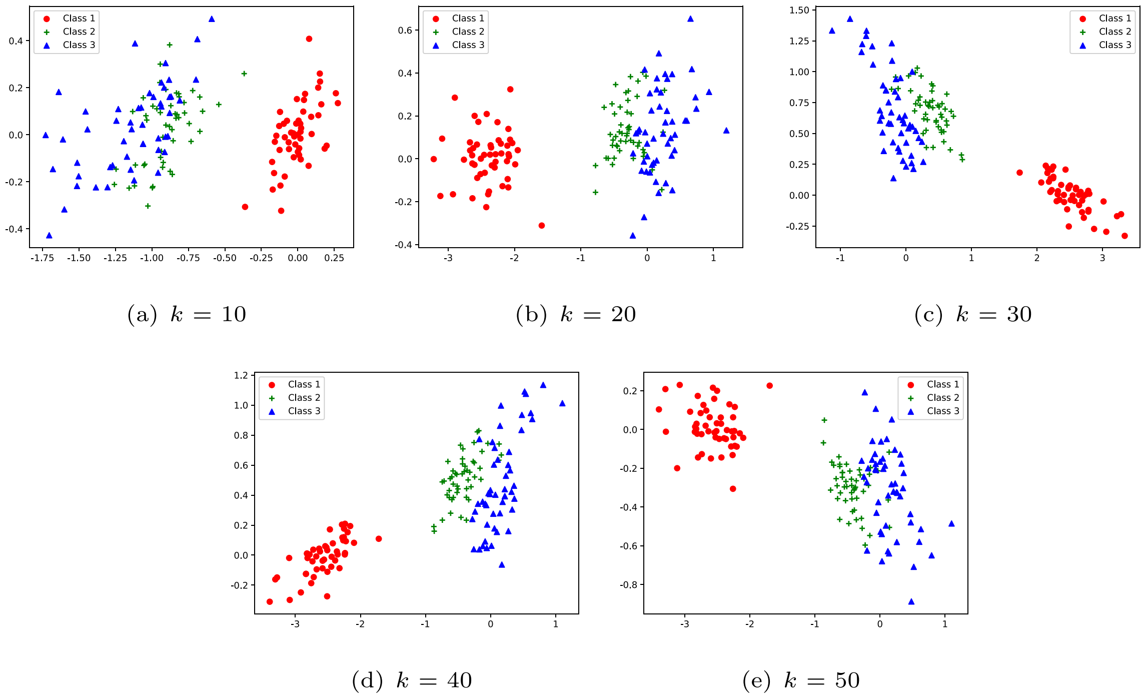
The iris dataset contains 150 records of three categories called iris-setosa, iris-versicolour, and iris-virginica respectively. Each category contains 50 data, and each data contains 4 characteristic variables, which means the sample is 4-dimensional. Figure 2 shows the dimension reduction effect of LLDE on the iris data with 10, 20, 30, 40, and 50 neighbors, while Fig. 3 is the dimension reduction effect of CLLDE on the iris data with 10, 20, 30, 40, and 50 neighbors. Through the comparison, we can conclude that CLLDE achieves a better dimensionality reduction effect on the iris dataset than LLDE.



(a) $k = 10$ (b) $k = 20$ (c) $k = 30$

(d) $k = 40$ (e) $k = 50$

Fig. 2 LLDE dimension reduction effect on iris dataset



(a) $k = 10$ (b) $k = 20$ (c) $k = 30$

(d) $k = 40$ (e) $k = 50$

Fig. 3 CLLDE dimension reduction effect on iris dataset

In summary, the experiments on the iris dataset show that CLLDE can better distinguish different categories of data, and it is more suitable for pattern recognition with labels under the same neighborhood conditions for uneven data or deformed data.

3 Elite opposite sparrowsearch algorithm

3.1 Sparrow search algorithm and elite opposite sparrow search algorithm

SSA is a novel swarm intelligence algorithm inspired by the behaviors of the sparrows. SSA searches for the solution to the optimization problem by imitating the foraging behaviors and anti-predatory behaviors of the sparrows. Firstly, some sparrows are randomly selected as producers in the sparrow population. The main mission of the producers is to find food for the whole population, which determines the direction and the area of foraging. Scroungers will follow the producers to find food, and some scroungers will keep a lookout over the producers and wrest food from the producers. Once the predator is detected, the individuals will send the anti-predatory signal. Finally, sparrows find the most suitable location to forage for the whole group by iterating the location of producers and scroungers for many iterations.

The dimension of the sparrow population is $n \times H$, and n is the number of all the sparrows. H represents the dimensionality of variables to be optimized. The position of the i-th sparrows can be described as $S_i = [s_{i,1}, s_{i,2}, \dots, s_{i,q}], i \in [1, n]$, $q \in [1, H]$, and $s_{i,j}$ shows the position of the i-th sparrow having j dimensional variables. The fitness value of the sparrow population can be described as

$$F_S = \begin{bmatrix} f\left(\left[s_{1,1}, s_{1,2}, \dots, s_{1,q}\right]\right) \\ f\left(\left[s_{2,1}, s_{2,2}, \dots, s_{2,q}\right]\right) \\ \vdots \\ f\left(\left[s_{n,1}, s_{n,2}, \dots, s_{n,q}\right]\right) \end{bmatrix}, \tag{22}$$

where n represents the number of all the sparrows of population, and the value of each row in F_S is the fitness value of the individual sparrow.

In SSA, the mission of the producers is to find food for the population, and the main behavior of the scroungers is mainly to follow the producers to get food. Besides, some scroungers may compete with producers for the food, and the scroungers will get the food if they win. Based on the above description, the location of producers is defined as

$$s_{i,j}^{t+1} = \begin{cases} s_{i,j}^{t} \cdot \exp\left(\dfrac{-i}{\beta \cdot T}\right), & A_{\text{alarm}} < J_{th}, \\ s_{i,j}^{t} + Q \cdot \vartheta, & A_{\text{alarm}} \geqslant J_{th}, \end{cases} \tag{23}$$

where t represents the current iteration, $s_{i,j}^{t}$ is the value of the i-th sparrow in the j-th dimension at the t-th iteration, T is the maximum number of iterations, and $\beta \in (0, 1]$ is a random number. $A_{\text{alarm}} \in [0, 1]$ represents the warning value while $J_{th} \in [0.5, 1]$ expresses the safety threshold. Q is a random number that is subject to the normal distribution. ϑ is a $1 \times q$ matrix where all elements are one (Xue and Shen 2020).

The position of the scrounger can be updated by

$$s_{i,j}^{t+1} = \begin{cases} Q \cdot \exp\left(\dfrac{s_{\text{worst}}^{t} - s_{i,j}^{t}}{i^2}\right), & i > n/2, \\ s_{\text{pbest}}^{t+1} + |s_{i,j}^{t} - s_{\text{pbest}}^{t+1}| \cdot \iota^{+} \cdot \vartheta, & i \leqslant n/2, \end{cases} \tag{24}$$

where s_{pbest}^{t+1} represents the best position in the global search space at the t+1 iteration, while s_{worst}^{t} represents the global worst position at t-th iteration. ι is a $1 \times q$ matrix whose elements are randomly assigned 1 or -1, and ι^{+} satisfies $\iota^{+} = \iota^{T}(\iota\iota^{T})^{-1}$.

When the sparrows in the population are aware of the danger, for the sparrows at the edge position, they will fly to a safe place. For the sparrows in the middle of the population, they approach other sparrows in a random walk. According to mentioned above, the mathematical model can be expressed as

$$s_{i,j}^{t+1} = \begin{cases} s_{\text{best}}^{t} + \gamma \cdot \left(s_{i,j}^{t} - s_{\text{best}}^{t}\right), f_i \neq f_g, \\ s_{i,j}^{t} + K \cdot \left(\frac{s_{i,j}^{t} - s_{\text{worst}}^{t}}{|f_i - f_w| + \mu}\right), \quad f_i = f_g, \end{cases} \tag{25}$$

where s_{best}^{t} express the optimal position in the global search space at t-th iteration, and γ is a random number, which subjects to standard normal distribution and controls the step size. $K \in [-1, 1]$ shows the moving direction of sparrow. μ is a very small constant, which is aimed to avoid zero-division-error.

In a swarm intelligence algorithm, the initial solution determines the distribution of the initial population in the solution space. The spatial distribution of the solution will affect the search-ability and convergence efficiency of the algorithm. The high-quality initial population can accelerate the convergence speed of the algorithm, which is more conducive to finding the global optimal solution. Based on mentioned above, the elite opposite learning and orifice imaging opposite learning strategy are introduced into SSA, and the elite opposite sparrow search algorithm (EOSSA) is proposed.

Before introducing EOSSA, it is necessary to explain the opposite point and the elite opposite solution. Let $s = \left[s_1, s_2, \ldots, s_q\right]$ be a point in q dimensional space, where $s_i \in \left[\upsilon_j, \omega_j\right], j = 1, 2, \ldots, q$, υ_j and ω_j represent the lower boundary and the upper boundary of search space, respectively. The opposite point is defined as (Wang et al 2011)

$$s_j^* = \upsilon_j + \omega_j - s_j. \tag{26}$$

Let $s_{i,j}$ be the value of ordinary individual s_i in the j dimension, then the elite opposite solution can be defined as (Sihwail et al 2020)

$$s_{i,j}^e = \rho\left(\upsilon_j + \omega_j\right) - s_{i,j}, \tag{27}$$

where $\rho \in [0, 1]$ is a random number, $\left[\upsilon_j, \omega_j\right]$ is the interval constructed by the elite sparrows, and $s_{i,j}^e \in \left[\upsilon_j, \omega_j\right]$.

The elite opposite strategy can provide more alternative solutions. The dynamic boundary can accelerate the convergence of SSA. EOSSA takes the sparrows with the top $v\%$ of fitness values as the elite solutions. v is a constant that represents the ratio of the elite sparrows. EOSSA can obtain the dynamic boundary of the elite sparrows, and then uses the opposite strategy to obtain the opposite solutions. Compared with the sparrow before updating and the sparrow after updating, if it obtains a better value, it will replace the previous sparrow.

The rest of the population is the ordinary sparrows. For ordinary sparrows, their low fitness is mainly because the search area is far away from the search range of excellent individuals. To expand the search range and have the ability to jump out of the current poor search area, and to increase the diversity of the population, we use the different opposite learning strategies from elite individuals to operate on ordinary sparrows.

Algorithm 2 Elite Opposite Sparrow Search Algorithm

```
Require: The maximum iterations $T$, the number of the sparrows $n$, the lower boundary $B_{ub}$ and the upper boundary $B_{lb}$, random constant $\rho$, the number of elite sparrow $E$, the regulation factor $\varrho$.
Ensure: Fitness value $f_g$, the optimal solution $s_{best}$.
$\upsilon_j^e$ and $\omega_j^e$: the dynamic search boundary of the elite sparrows
$\upsilon_j^{oi}$ and $\omega_j^{oi}$: the dynamic search boundary of the ordinary sparrows
Initialize sparrow population $n$ by using rand(1) $\times (B_{ub} - B_{lb}) + B_{lb}$. Set the number of iteration. Initialize the regulation factor $\varrho$.
while $(t < T)$ do
    Calculate and sort the fitness value of population;
    for $i = 1 : n$
        Update the position of the sparrows by (23), (24), (25);
    end for
    Select the elite sparrows $s_{i,j}^e$;
    $\upsilon_j^e \leftarrow \min\left(s_{i,j}^e, 0\right)$, $w_j^e \leftarrow \max\left(s_{i,j}^e, 0\right)$;
    for $i = 1 : E$
        Update the position of the elite sparrow individual by (27);
        if $fitness\left(s_{i,j}^e\right) < fitness\left(s_{i,j}^t\right)$ then
            $s_{i,new}^{t+1} \leftarrow s_{i,j}^e$;
        else
            $s_{i,new}^{t+1} \leftarrow s_{i,j}^t$;
        end if
    end for
    $\upsilon_j^{oi} \leftarrow \min\left(s_{i,j}^e, 0\right)$, $w_j^{oi} \leftarrow \max\left(s_{i,j}^{oi}, 0\right)$;
    for $i = E + 1 : n$
        Using (31) to update the location of the ordinary sparrows;
        if $fitness\left(s_{i,j}^{oi}\right) < fitness\left(s_{i,j}^t\right)$ then
            $s_{i,new}^{t+1} \leftarrow s_{i,j}^{oi}$;
        else
            $s_{i,new}^{t+1} \leftarrow s_{i,j}^t$;
        end if
    end for
    Update and sort the fitness values of population;
    $t \leftarrow t + 1$;
end while
return $f_g$ and $s_{best}$.
```

Orifice imaging opposite learning is an opposite learning strategy based on optical principle (Zhang et al 2021), which assumes that there is a light source ϖ with a height of h, and the upper and lower boundaries of the coordinate axis are υ_j and ω_j, respectively. The projection point of the light source on the coordinate axis is s. The image that has height $\tilde{h}$ formed by the light source through the orifice is ϖ^*, and its projection on the coordinate axis is s^{oi}, which is called the opposite point of s. The schematic diagram of orifice imaging is shown in Fig. 4.

According to the principle of orifice imaging, the process can be described as

$$\frac{\left(\upsilon_j+\omega_j\right)/2-s}{s^{oi}-\left(\upsilon_j+\omega_j\right)/2}=\frac{h}{\tilde{h}}. \tag{28}$$

Let $h/\tilde{h}=\varrho$, then (28) can be transformed as

$$s^{oi}=\frac{\left(\upsilon_j+\omega_j\right)}{2}+\frac{\left(\upsilon_j+\omega_j\right)}{2\varrho}-\frac{s}{\varrho}. \tag{29}$$

The process is extended to EOSSA, which can be described as

$$\frac{\left(\upsilon_j+\omega_j\right)/2-s_{i,j}^t}{s_{i,j}^{oi}-\left(\upsilon_j+\omega_j\right)/2}=\varrho. \tag{30}$$

$$s_{i,j}^{oi}=\frac{\left(\upsilon_j+\omega_j\right)}{2}+\frac{\left(\upsilon_j+\omega_j\right)}{2\varrho}-\frac{s_{i,j}^t}{\varrho}, \tag{31}$$

where $s_{i,j}^t$ is the j-th dimension of the i-th sparrow individual in the t-th iteration, $s_{i,j}^{oi}$ formed by imaging through orifice represents the opposite point of $s_{i,j}^t$, υ_j and ω_j represent the lower boundary and the upper boundary of search space, respectively, ϱ is regulation factor.

Remark 1 SSA as a swarm intelligence optimization algorithm is essentially a greedy algorithm. SSA may fall into a locally extremum point. The orifice imaging opposite strategy and opposite learning strategy are introduced into SSA to make it have the ability to escape the local extremum point. These strategies can help the algorithm explore more space in a limited time. The opposite strategy will make the sparrow search in the opposite direction in its searching space when the algorithm tends to local optimization. Besides, this approach can increase the diversity of the search population, which is conducive to the algorithm to obtain the optimal solution.

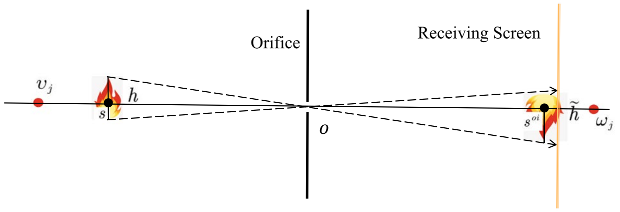


Fig. 4 The schematic diagram of orifice imaging principle

The steps of the EOSSA can be summarized as Algorithm 2.

3.2 The convergence characteristics of the algorithms on benchmark functions

In this part, benchmark functions are used to verify the feasibility and effectiveness of EOSSA. SSA, chaotic SSA (CSSA) (Zhang and Ding 2021), chaos particle swarm optimization(CPSO) (Kennedy and Eberhart 1995; Su et al 2015; Pluhacek et al 2018), disturbance and somersault foraging grey wolf optimizer (DSFGWO) (Mirjalili et al 2014; Wang et al 2021), and Levy-flight based moth-flame optimization (LMFO) (Mirjalili 2015; Suja 2021) are used to compare the convergence characteristics with EOSSA. The initial population number of all algorithms is set to 100. The number of iterations is 1000. The parameters of each algorithm for comparison are set according to (Mirjalili 2015; Xue and Shen 2020). The detailed information of the standard benchmark functions is listed in Table 1 and the optimization results are listed in Table 2. The optimization results are the average value and standard deviation (Std) of 30 experiments. Figure 5 shows the optimization process curves of the algorithms on the unimodal benchmark functions $F_1(x)$, $F_2(x)$, $F_3(x)$, $F_4(x)$, and $F_5(x)$. Figure 6 shows the optimization process curves of the algorithms on the multimodal benchmark functions $F_6(x)$, $F_7(x)$, $F_8(x)$, and $F_9(x)$, while Fig. 7 shows the optimization process curves of the algorithms on the fixed-dimension benchmark functions $F_{10}(x)$ and $F_{11}(x)$.

Remark 2 F_1 to F_5 are the unimodal benchmark functions. This kind of benchmark function has only one extreme point. They can be used to verify the convergence speed, optimization accuracy, and local development ability of the algorithm. F_6 to F_9 are multimodal benchmark functions. This kind of benchmark function has multiple local extremum points, which makes the algorithm extremely easy to fall into the local extremum. They can be used to verify the ability of the algorithm to escape from the local extremum and global exploration ability. F_{10} and F_{11} are the fixed dimension benchmark functions. They can be used to further verify the convergence speed, stability, and convergence accuracy of the algorithm.

From the optimization results, because the opposite strategy provides more alternative solutions for the algorithm and improves the ability to avoid falling into local optimum, the optimization performance of EOSSA is improved compared with SSA. After 30 experiments, the average test result

Table 1 The list of benchmark functions

Functions	Type	Dim	Optimal value
$F_1(x) = \sum_{i=1}^{n} x_i^2$	Unimodal	30	0
$F_2(x) = \sum_{i=1}^{n} \lvert x_i \rvert + \prod_{i=1}^{n} \lvert x_i \rvert$	Unimodal	30	0
$F_3(x) = \max_i\{\lvert x_i \rvert, 1 \le i \le n\}$	Unimodal	30	0
$F_4(x) = \sum_{i=1}^{n-1} \left[100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2\right]$	Unimodal	30	0
$F_5(x) = \sum_{i=1}^{n} (\left[x_i + 0.5\right])^2$	Unimodal	30	0
$F_6(x) = \sum_{i=1}^{n} -x_i \sin(\sqrt{\lvert x_i \rvert})$	Multimodal	30	– 418.9829 ×n
$F_7(x) = \sum_{i=1}^{n} [x_i^2 - 10\cos(2\pi x_i) + 10]$	Multimodal	30	0
$F_8(x) = -20 \exp(-0.2\sqrt{\frac{1}{n} \sum_{i=1}^{n} x_i^2}) - \exp(\frac{1}{n} \sum_{i=1}^{n} \cos(2\pi x_i)) + 20 + e$	Multimodal	30	0
$F_9(x) = \frac{1}{4000} \sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n} \cos(\frac{x_i}{\sqrt{i}}) + 1$	Multimodal	30	0
$F_{10}(x) = 4x_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1x_2 - 4x_2^2 + 4x_2^4$	Fixed-dimension	2	– 1.0316
$F_{11}(x) = [1 - \lvert \frac{\sin[\pi(x_1-2)]\sin[\pi(x_2-2)]}{\pi^2(x_1-2)(x_2-2)} \rvert^5][2 + (x_1 - 7)^2 + 2(x_2 - 7)^2]$	Fixed-dimension	2	0

of EOSSA is better than other algorithms, and Std is also better than other algorithms in most benchmark functions, but the standard deviation is slightly worse than others in a few benchmark functions, such as $F_6(x)$. From the convergence curve, the convergence speed of EOSSA is improved compared with SSA.

To better evaluate and compare the optimization performance of various optimization algorithms, the box plot is used for statistical comparison and analysis. Figure 8 a–k show the statistical results of SSA, EOSSA, CSSA, CPSO, DSFGWO, and LMFO on each benchmark function of this experiment, respectively. From the results of the box plots, it can be concluded that the comprehensive performance of EOSSA on the benchmark functions is better than other algorithms. In summary, the convergence characteristics of EOSSA are better than the other five optimization algorithms on the whole from the above analysis.

4 Experiments of fault diagnosis

The feature extraction accuracy will affect the final recognition rate. Simple and significant features can improve fault recognition accuracy and reduce computational complexity. Manifold learning reduces the redundant dimension through the geometric structure of data. LLE is a classical manifold learning algorithm, which can reveal the internal structure of data. For labeled data, it does not use the category attributes of data. Therefore, LLDE is proposed by combining LLE with MMC. Because of the deformed data distribution, the Euclidean distance in LLDE is replaced by cam weighted distance, and CLLDE is proposed. The previous experiment shows a good effect on the classification problem.

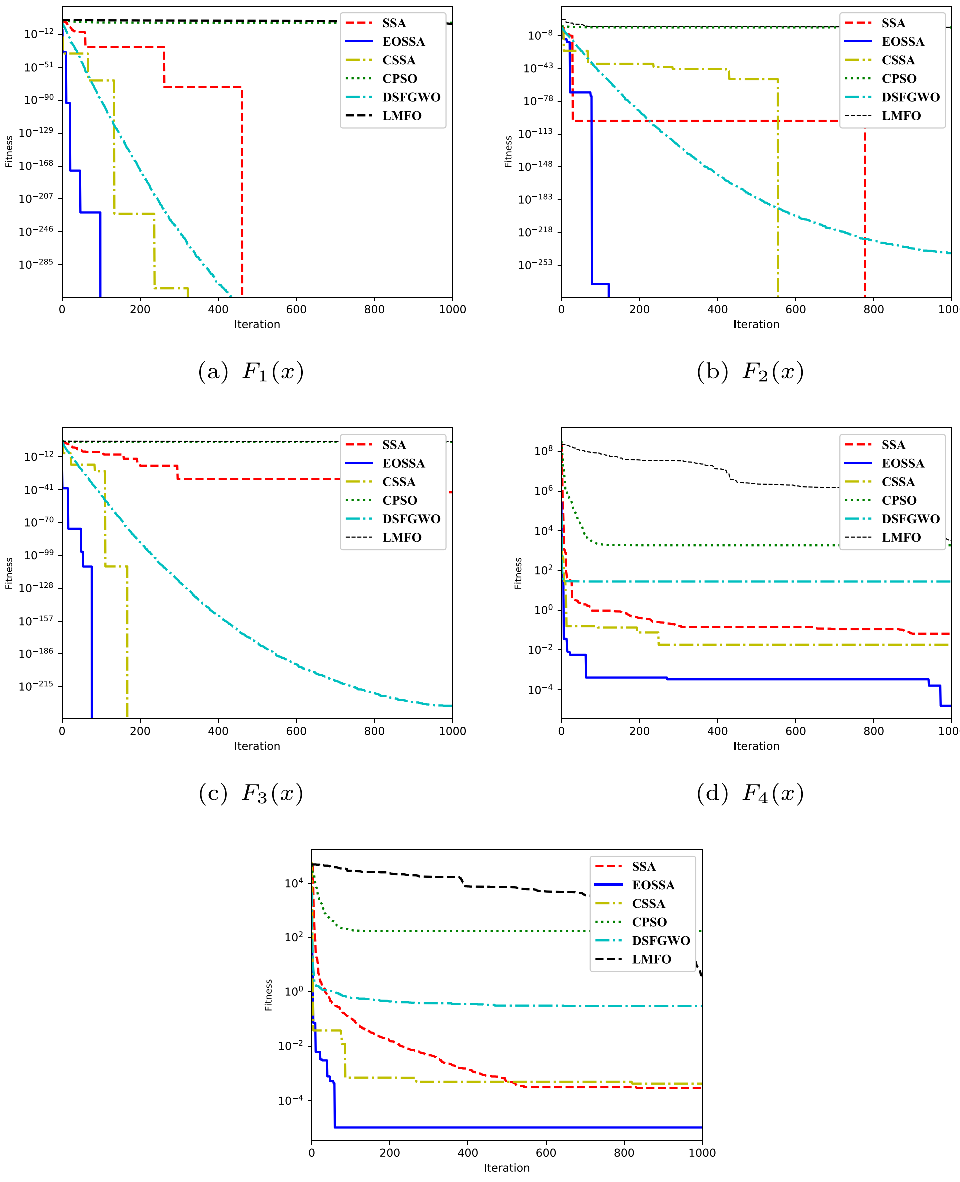
The main steps of the fault diagnosis approach can be summarized as follows. Firstly, CLLDE is used to map the original high-dimensional data to the low-dimensional feature space. Secondly, EOSSA is used to optimize the hyper-parameters of LightGBM to establish a classifier to obtain a better diagnosis effect. Finally, the trained EOSSA-LightGBM is used to identify the new test fault data. The flowchart of EOSSA-LightGBM is shown in Fig. 9.

The fault data used in this paper is Tennessee Eastman (TE) process (Yin et al 2012) dataset. Each sample of the simulation dataset contains 52 observed variables, among which the first 22 are non-categorical variables, the 23rd to 41st are categorical variables, and the last 11 are control variables. Considering the influence of the real environment, Gaussian noise is added to all observed variables.

No. 18 data set is selected to test the effectiveness of the algorithm. There are 980 training samples, including 480 fault samples and 500 normal samples. The training samples are randomly divided into training set and validation set according to the ratio of 7 : 3. There are 960 samples in the testing set, including 160 normal samples and 800 fault samples. The intrinsic dimension of CLLDE is set to 7 and the numbers of the nearest neighbor point are 13 after validating.

Table 2 Results of benchmark functions

Functions		SSA	EOSSA	CSSA	CPSO	DSFGWO	LMFO
F_1							
	Ave	4.022E – 47	0	8.001E – 223	54.066	0	3.309
	Best	0	0	0	11.773	0	0.145
	Std	1.784E – 46	0	0	34.647	0	6.468
F_2							
	Ave	2.517E – 29	0	9.404E – 140	11.927	1.108E – 235	22.722
	Best	0	0	0	4.653	8.638E – 241	0.085
	Std	6.982E – 29	0	4.099E – 139	5.324	0	25.826
F_3							
	Ave	8.226E – 22	0	4.654E – 128	12.916	7.119E – 230	48.662
	Best	0	0	0	7.524	1.915E – 234	25.917
	Std	3.679E – 21	0	1.975E – 127	2.590	0	11.438
F_4							
	Ave	0.029	2.383E – 04	0.010	1202.341	27.120	1271.089
	Best	7.462E – 09	3.053E – 08	1.398E – 04	224.222	26.178	219.703
	Std	0.025	5.155E – 04	0.009	693.940	0.533	1255.157
F_5							
	Ave	2.071E – 04	2.092E – 05	4.184E – 04	48.772	0.259	3.332
	Best	3.382E – 07	1.209E – 07	6.611E – 07	12.967	0.046	0.641
	Std	1.651E – 04	2.879E – 05	4.000E – 04	41.854	0.236	4.034
F_6							
	Ave	– 8407.882	– 10515.376	– 7805.348	– 8033.901	– 6939.431	– 7736.184
	Best	– 12131.162	– 12569.486	– 12349.014	– 10071.662	– 9085.051	– 9279.032
	Std	1388.270	2327.085	4301.964	4020.411	1053.820	711.895
F_7							
	Ave	0	0	0	63.160	0	159.565
	Best	0	0	0	39.971	0	106.136
	Std	0	0	0	12.943	0	40.705
F_8							
	Ave	4.441E – 16	4.441E – 16	4.441E – 16	274.244	1.700E – 12	311.964
	Best	4.441E – 16	4.441E – 16	4.441E – 16	158.920	4.441E – 16	55.547
	Std	0	0	0	66.046	3.697E-12	537.414
F_9							
	Ave	0	0	0	1.440	0	12.337
	Best	0	0	0	0.831	0	0.084
	Std	0	0	0	0.348	0	29.392
F_{10}							
	Ave	– 1.0316	– 1.0316	– 1.0316	– 1.0316	– 1.0316	– 1.0316
	Best	– 1.0316	– 1.0316	– 1.0316	– 1.0316	– 1.0316	– 1.0316
	Std	0	0	0	0	0	0
F_{11}							
	Ave	2.539E – 04	2.245E – 05	0.006	2.667	0.004	2
	Best	1.329E – 06	3.652E – 08	8.748E – 05	1.506	5.040E – 07	2
	Std	4.535E – 04	3.135E – 05	0.007	0.606	0.005	2



(a) $F_1(x)$

(b) $F_2(x)$

(c) $F_3(x)$

(d) $F_4(x)$

(e) $F_5(x)$

Fig. 5 The convergence curves on the unimodal benchmark functions

F_1 score is used to analyze and evaluate the performance of the fault diagnosis approach. F_1 score can be calculated by precision rate and recall rate, the mathematical expressions of the three above indicators can be written as

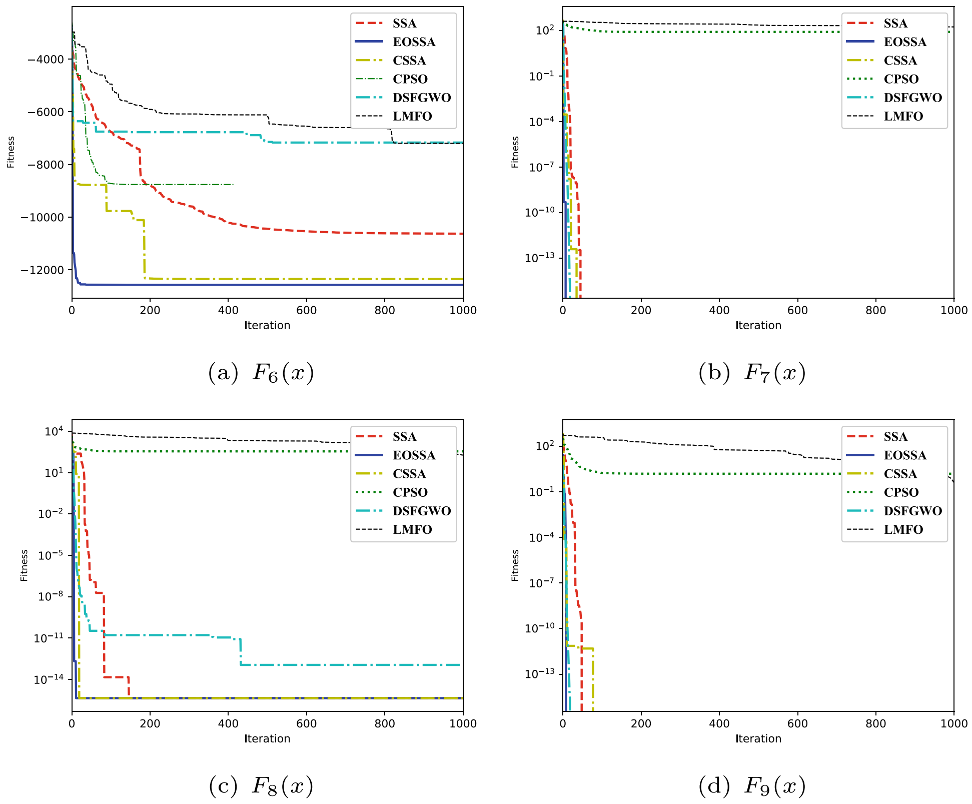
$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \tag{32}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \tag{33}$$

$$F_1 \text{ score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}, \tag{34}$$

where TP is the number that the predicted label of the sample is the same as the real label, FN is the number of the sample that the actual label is the misjudgment, and FP represents the number of the false positive sample.

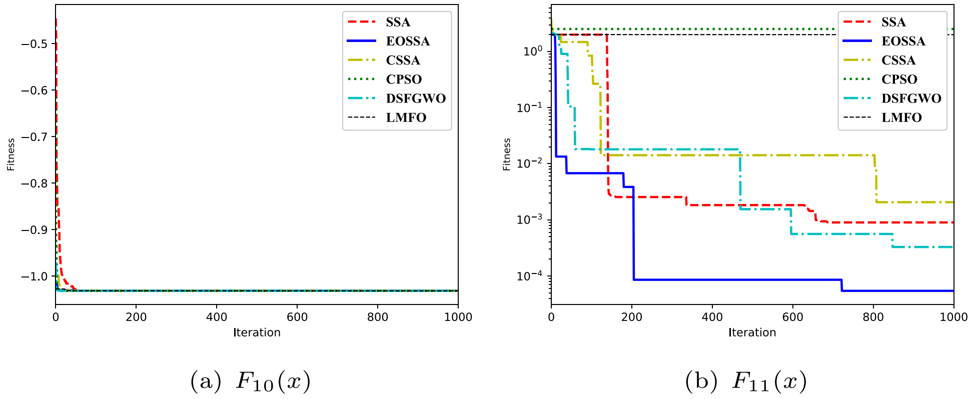
The process of fault diagnosis is divided into feature extraction and fault recognition. Different commonly used feature extraction methods and classification algorithms are combined and then compared with the method using CLLDE as feature extraction and EOSSA-LightGBM as fault diagnosis. Feature extraction methods include PCA,



(a) $F_6(x)$ (b) $F_7(x)$

(c) $F_8(x)$ (d) $F_9(x)$

Fig. 6 The convergence curves on the multimodal benchmark functions



(a) $F_{10}(x)$ (b) $F_{11}(x)$

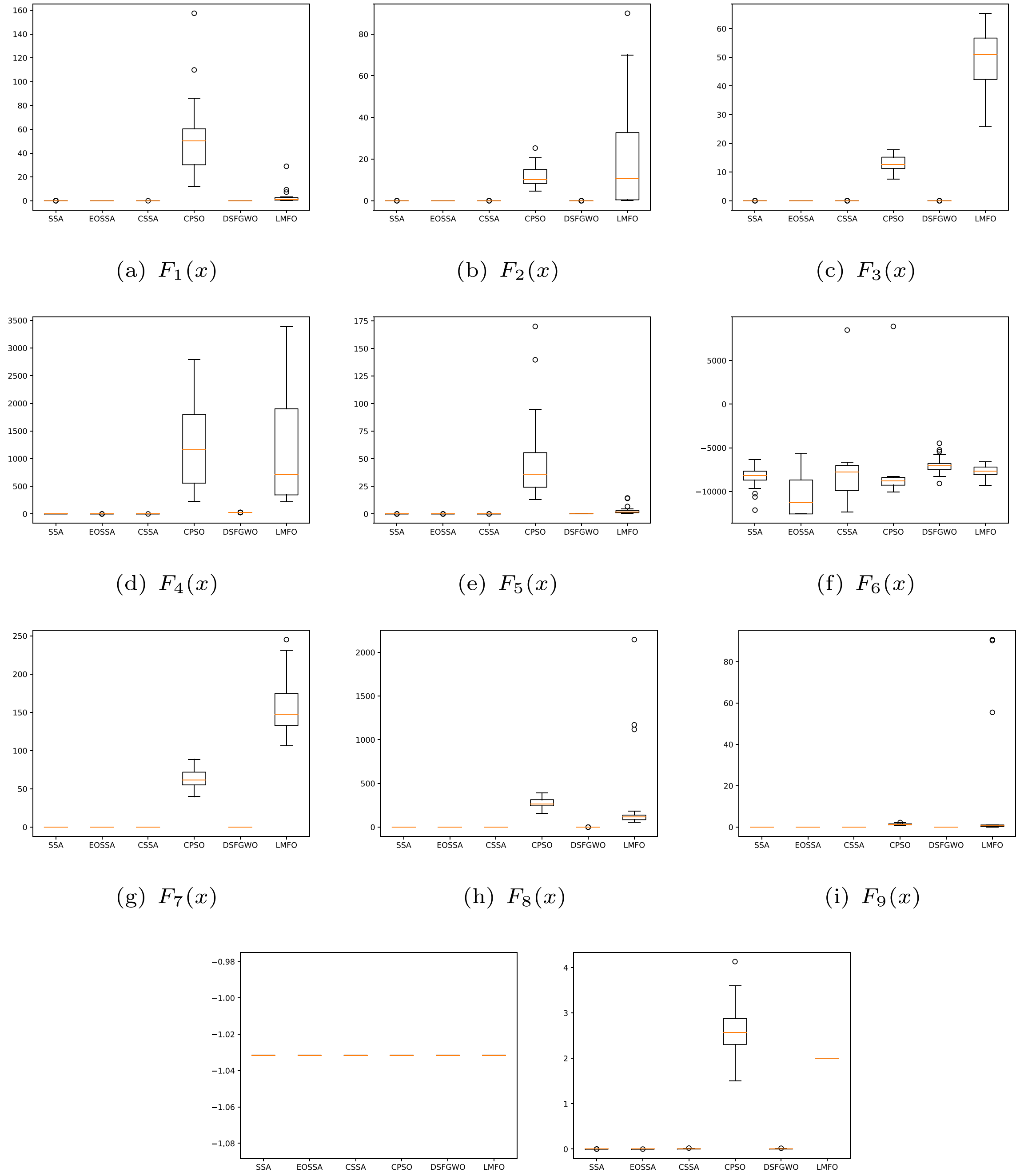
Fig. 7 The convergence curves on the fixed-dimension benchmark functions

KPCA, ISOMAP, LLE, and LLDE. Classification algorithms include random forest (RF) (Cerrada et al 2016) and support vector machine (SVM) (Mavroforakis and Theodoridis 2006; Pan et al 2008; Su et al 2015).

In order to obtain the best super parameters of LightGBM, EOSSA is used to optimize its parameters. The major hyper-parameters learning rate η and decision tree depth φ of LightGBM are selected for optimization. The calculation formula of F_1 score is selected as the fitness function of EOSSA, which is formulated as

$$f = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{35}$$

When the learning rate is 0.1058 and the depth of the decision tree is 7, the F_1 score is the best, and it is effectiveness is verified in the validation set. Therefore, the two super parameters η and φ of LightGBM are set to 0.1058 and 7, respectively. The experimental results on the test set are listed in Tables 3 and 4. In Table 3, the horizontal comparison is the performance of classifiers using different feature extraction algorithms, and the vertical comparison is the performance of classifiers using the same feature extraction algorithm. Table 4 lists the results of various optimization algorithms in fault diagnosis. In order to show the results more intuitively, its histogram is shown in Fig. 10. From Table 4 and Fig. 10, we can conclude the proposed approach



(a) $F_1(x)$ (b) $F_2(x)$ (c) $F_3(x)$

(d) $F_4(x)$ (e) $F_5(x)$ (f) $F_6(x)$

(g) $F_7(x)$ (h) $F_8(x)$ (i) $F_9(x)$

(j) $F_{10}(x)$ (k) $F_{11}(x)$

Fig. 8 Box plot comparison of six algorithms in this experiment

is better than others. The experiment proves the effectiveness of the approach we proposed.

Also, to compare the performance of various algorithms more intuitively, the receiver operating characteristic (ROC) curves and the precision-recall (PR) curves are shown in Figs. 11, 12, 13, 14, 15, and 16. Figures 11, 12, 13, 14, 15, and 16 shows that LightGBM is optimized by SSA, EOSSA, CSSA, CPSO, DSFGWO, and LMFO, respectively. They show the performance of different fault diagnosis approaches when using different feature extraction methods. From area under curve (AUC) values, we can conclude that the proposed approach is superior to other algorithms in fault identification in the contrast experiment.

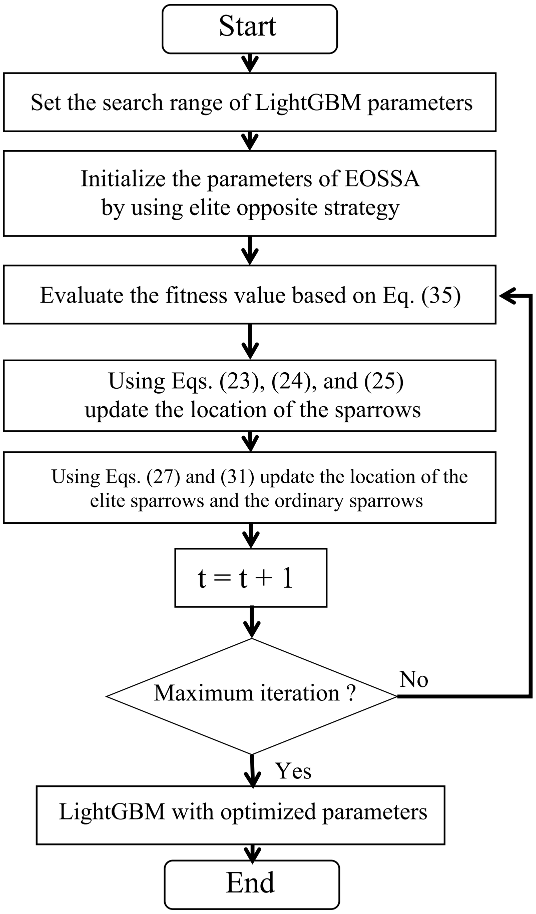
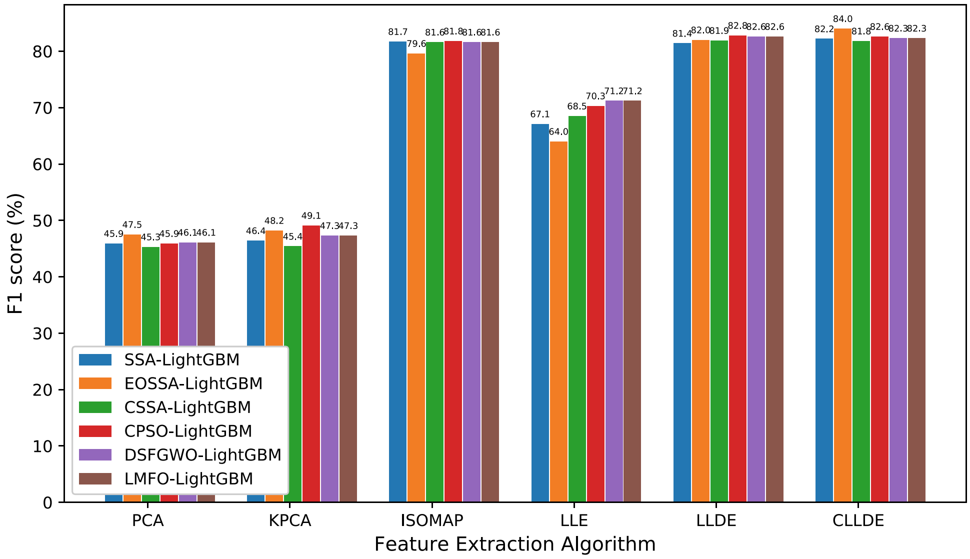

Start
Set the search range of LightGBM parameters
Initialize the parameters of EOSSA
by using elite opposite strategy
Evaluate the fitness value based on Eq. (35)
Using Eqs. (23), (24), and (25)
update the location of the sparrows
Using Eqs. (27) and (31) update the location of the
elite sparrows and the ordinary sparrows
t = t + 1
Maximum iteration ?
No
Yes
LightGBM with optimized parameters
End

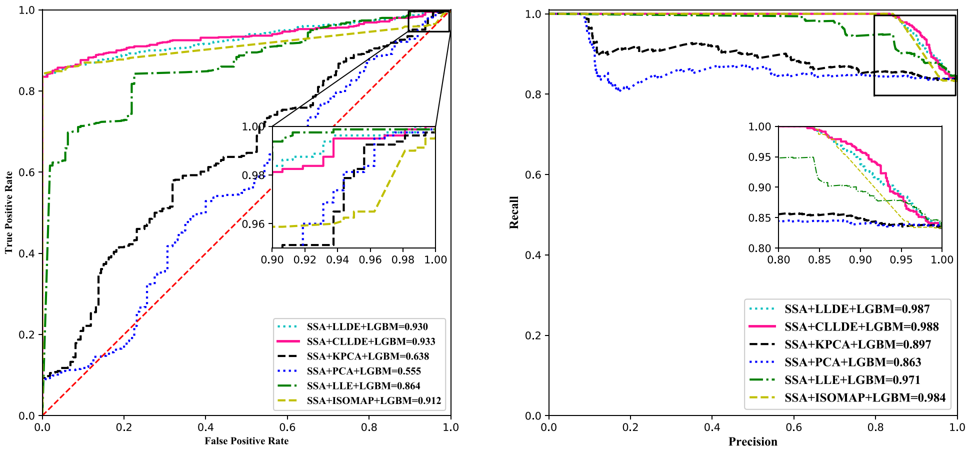
Fig. 9 The flowchart of EOSSA-LightGBM

5 Conclusion

In this paper, a new fault diagnosis approach based on EOSSA optimized LightGBM is proposed. Aimed at the deformation problem of data distribution, cam weighted distance is introduced into LLDE to extract the feature. The experiments show that CLLDE is effective. The influence of the randomly selected *k* value on CLLDE is smaller than that of LLDE in comparison. In the fault diagnosis experiment, CLLDE is used as a feature extraction method to improve the performance of various classifiers, which shows that it can effectively extract data features. EOSSA is proposed by introducing the elite opposite strategy and orifice imaging opposite learning strategy into SSA. The experimental results show that the strategies used in EOSSA are worked, which can accelerate the convergence speed of SSA as well as making the solution more effective. Overall, EOSSA shows superior performance on unimodal, multimodal, and fixed-dimension problems, but its statistical performance on the No. 6 benchmark function is not as good as other algorithms in comparison. EOSSA has the best solution of all the algorithms in the No. 6 benchmark function. However, its standard deviation is greater than other algorithms, which indicates that its performance in the No. 6 benchmark function is poor because it cannot guarantee that the solution of each searching mission is in an appropriate range. Compared with other optimization algorithms, EOSSA still shows advantages in the fault diagnosis problem, which indicates that EOSSA is still feasible and effective in this problem. In future work, we can study the application of deep learning (DL) (Liu et al 2017) and reinforcement learning (RL) in fault diagnosis (Song et al 2021). Also, we will study

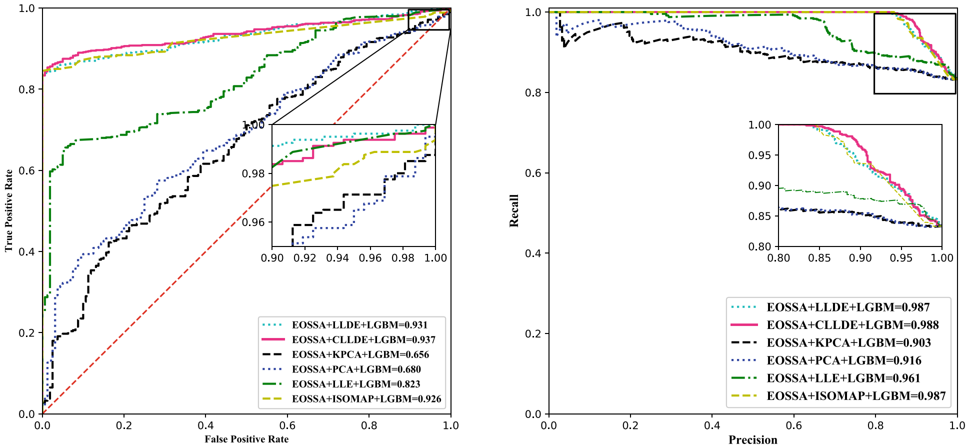
Fig. 10 The histogram of Table 4





(a) ROC curve (b) PR curve

Fig. 11 The ROC curve and PR curve of LightGBM optimized by SSA



(a) ROC curve (b) PR curve

Fig. 12 The ROC curve and PR curve of LightGBM optimized by EOSSA

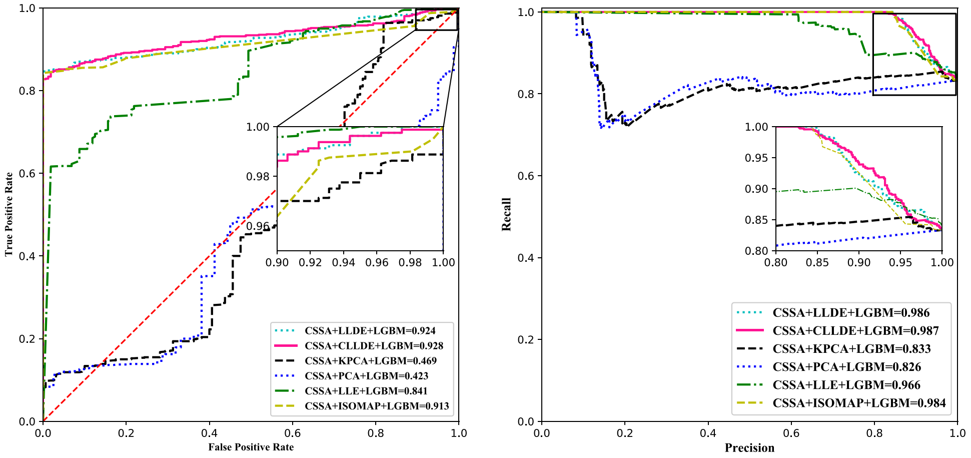
Table 3 F_1 score (%) comparison of different feature extraction methods and classifiers

Methods	PCA	KPCA	ISOMAP	LLE	LLDE	CLLDE
RF	45.5	45.4	66.7	75.7	77.6	81.2
SVM	45.0	45.1	78.6	71.4	78.0	79.2
LightGBM	43.5	43.3	77.9	63.2	64.7	81.4

Table 4 F_1 score (%) comparison of different feature extraction methods and optimization algorithm

Methods	PCA	KPCA	ISOMAP	LLE	LLDE	CLLDE
SSA-LightGBM	45.9	46.4	81.7	67.1	81.4	82.2
EOSSA-LightGBM	47.5	48.2	79.6	64.0	82.0	84.0
CSSA-LightGBM	45.3	45.4	81.6	68.5	81.9	81.8
CPSO-LightGBM	45.9	49.1	81.8	70.3	82.8	82.6
DSFGWO-LightGBM	46.1	47.3	81.6	71.2	82.6	82.3
LMFO-LightGBM	43.9	45.4	81.2	64.1	82.8	82.8

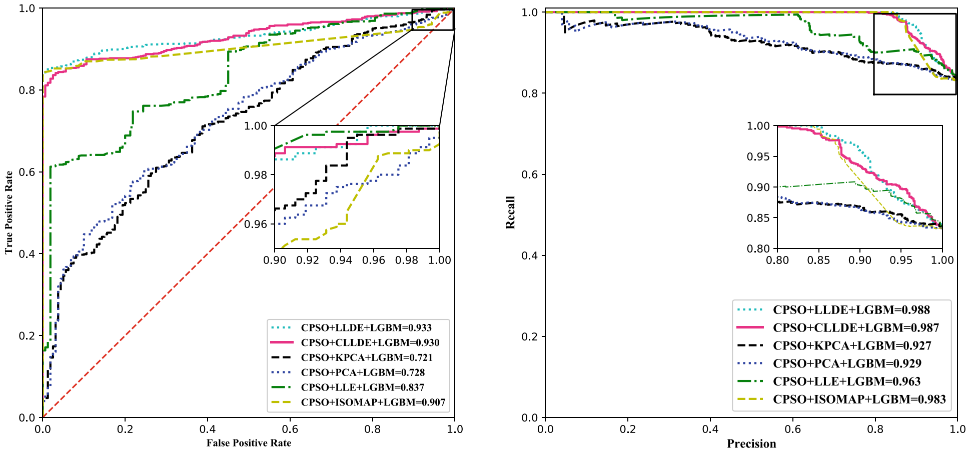
Fig. 13 The ROC curve and PR curve of LightGBM optimized by CSSA



(a) ROC curve

(b) PR curve

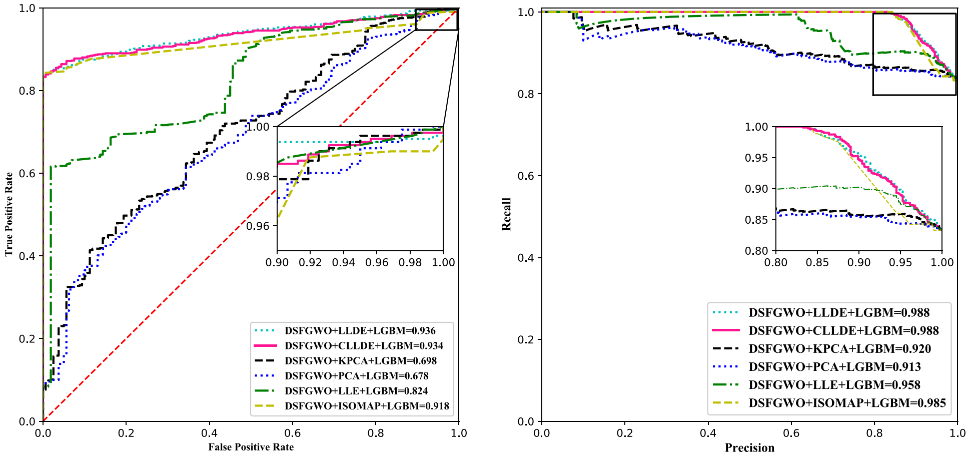
Fig. 14 The ROC curve and PR curve of LightGBM optimized by CPSO



(a) ROC curve

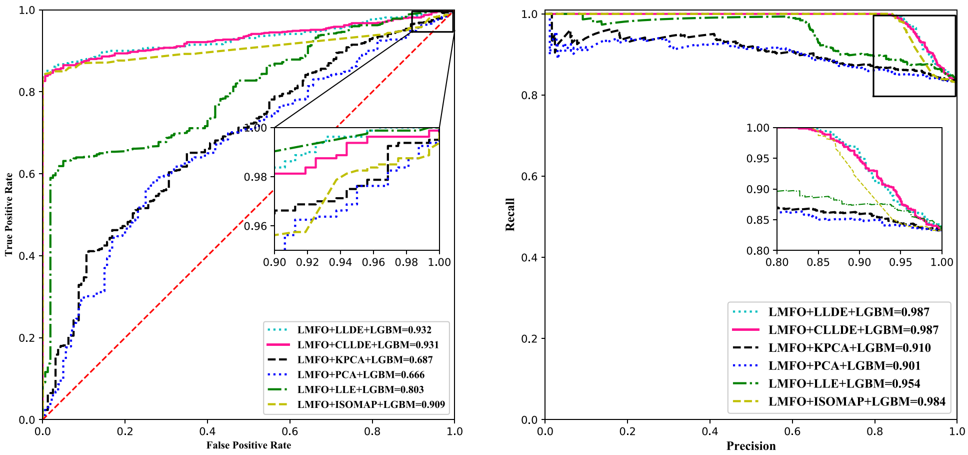
(b) PR curve

Fig. 15 The ROC curve and PR curve of LightGBM optimized by DSFGWO



(a) ROC curve

(b) PR curve



(a) ROC curve

(b) PR curve

Fig. 16 The ROC curve and PR curve of LightGBM optimized by LMFO

the performance of DL and RL combined with EOSSA in engineering applications.

Author Contributions All authors discussed the results and contributed to the final manuscript.

Funding This work was supported in part by the National Natural Science Foundation of China under Grants 61873059 and 61922024, and the Program of Shanghai Academic/Technology Research Leader of China under Grant 20XD1420100.

Availability of data and material Iris dataset https://scikit-learn.org/stable/. Tennessee Eastman process dataset https://github.com/caoyue123202/tennessee-eastman-profBraatz.

Declarations

Conflicts of 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.

Code availability The codes of the algorithms could be provided upon request.

References

Cerrada M, Zurita G, Cabrera D et al (2016) Fault diagnosis in spur gears based on genetic algorithm and random forest. Mech Syst Signal Proc 70–71:87–103. https://doi.org/10.1016/j.ymssp.2015.08.030

Cox T, Cox M (1994) Multidimensional scaling. Chapman and Hill, London

Duchene J, Leclercq S (1988) An optimal transformation for discriminant and principal component analysis. IEEE Trans Pattern Anal Mach Intell 10(6):978–983. https://doi.org/10.1109/34.9121

Duda R, Hart P, Stork D (2001) Pattern classification. John Wiley and Sons, New York

Etemad K, Chellapa R (1997) Discriminant analysis for recognition of human face images. J Opt Soc Am A-Opt Image Sci Vis 14(8):1724–1733. https://doi.org/10.1364/JOSAA.14.001724

Fan M, Liu Y, Zhang X et al (2019) Fault prediction for distribution network based on CNN and LightGBM algorithm. IEEE Int Conf Electron Meas Instruments, ICEMI, Changsha, pp 1020–1026

Ke G, Meng Q, Finley T et al (2017) LightGBM: a highly efficient gradient boosting decision tree. Adv Neural Inf Proces Syst, Long Beach, pp 3149–3157

Kennedy J, Eberhart R (1995) Particle swarm optimization. IEEE Int Conf Neural Networks Conf Proc 4:1942–1948

Law M, Jain A (2006) Incremental nonlinear dimensionality reduction by manifold learning. IEEE Trans Pattern Anal Mach Intell 28(3):377–391. https://doi.org/10.1109/TPAMI.2006.56

Lei Y, Hu R, Tang L et al (2010) Orthogonal linear local spline discriminant embedding for face recognition. In: Proceeding of International Joint Conference of Neural Networks (IJCNN), Barcelona, Spain, pp 1–8

Li H, Jiang T, Zhang K (2006) Efficient and robust feature extraction by maximum margin criterion. IEEE Trans Neural Networks 17(1):157–165. https://doi.org/10.1109/TNN.2005.860852

Li B, Zheng C, Huang D (2008) Locally linear discriminant embedding: an efficient method for face recognition. Pattern Recogn 41(12):3813–3821. https://doi.org/10.1016/j.patcog.2008.05.027

Liu W, Wang Z, Liu X et al (2017) A survey of deep neural network architectures and their applications. Neurocomputing 234:11–26. https://doi.org/10.1016/j.neucom.2016.12.038

Liu W, Wang Z, Liu X et al (2019) A novel particle swarm optimization approach for patient clustering from emergency departments. IEEE Trans Evol Comput 23(4):632–644. https://doi.org/10.1109/TEVC.2018.2878536

Mavroforakis M, Theodoridis S (2006) A geometric approach to support vector machine (SVM) classification. IEEE Trans Neural Networks 17(3):671–682. https://doi.org/10.1109/TNN.2006.873281

Mirjalili S (2015) Moth-flame optimization algorithm: a novel nature-inspired heuristic paradigm. Knowl Based Syst 89:228–249. https://doi.org/10.1016/j.knosys.2015.07.006

Mirjalili S, Mirjalili S, Lewis A (2014) Grey wolf optimizer. Adv Eng Software 69:46–61. https://doi.org/10.1016/j.advengsoft.2013.12.007

Pan Y, Ge S, Mamun A et al (2008) Detection of seizures in EEG signal using weighted locally linear embedding and SVM classifier. IEEE Int Conf Cybern Intell Syst (CIS) 1–2:358–363

Pan Y, Ge S, Mamun A (2009) Weighted locally linear embedding for dimension reduction. Pattern Recogn 42:798–811. https://doi.org/10.1016/j.patcog.2008.08.024

Pathana S, Siddalingaswamya P, Ali T (2021) Weighted locally linear embedding for dimension reduction. Appl Soft Comput. https://doi.org/10.1016/j.asoc.2021.107238

Pluhacek M, Senkerik R, Viktorin A et al (2018) Chaos driven PSO with attractive search space border points. IEEE Congr Evol Comput 2018:1–6

Roweis S, Saul L (2000) Nonlinear dimensionality reduction by locally linear embedding. Science 290(5500):2323–2326. https://doi.org/10.1126/science.290.5500.2323

Schölkopf B, Smola A, Müller K (1997) Kernel principal component analysis. Int Conf Artif Neural Networks 1327:583–588

Sihwail R, Omar K, Ariffin K et al (2020) Improved harris hawks optimization using elite opposition-based learning and novel search mechanism for feature selection. IEEE Access 8:121127–121145. https://doi.org/10.1109/ACCESS.2020.3006473

Song X, Cong Y, Song Y et al (2021) A bearing fault diagnosis model based on CNN with wide convolution kernels. J Ambient Intell Humaniz Comput. https://doi.org/10.1007/s12652-021-03177-x

Su Z, Tang B, Liu Z et al (2015) Multi-fault diagnosis for rotating machinery based on orthogonal supervised linear local tangent space alignment and least square support vector machine. Neurocomputing 157:208–222. https://doi.org/10.1016/j.neucom.2015.01.016

Suja K (2021) Mitigation of power quality issues in smart grid using levy fight based moth fame optimization algorithm. J Ambient Intell Humaniz Comput 12:9209–9228. https://doi.org/10.1007/s12652-020-02626-3

Tenenbaum J, Silva D, Langford J (2000) A global geometric framework for nonlinear dimensionality reduction. Science 290(5500):2319–2323. https://doi.org/10.1126/science.290.5500.2319

Wang H, Wu Z, Rahnamayan S et al (2011) Enhancing particle swarm optimization using generalized opposition-based learning. Inf Sci 181(20):4699–4714. https://doi.org/10.1016/j.ins.2011.03.016

Wang Z, Cheng F, You W et al (2021) Grey wolf optimization based on disturbance and somersault foraging. Appl Res Comput 38(5):1434–1437. https://doi.org/10.19734/j.issn.1001-3695.2020.04.0102

Wold S, Esbensen K, Geladi P (1987) Principal component analysis. Chemometrics Intell Lab Syst 2(1–3):37–52. https://doi.org/10.1016/0169-7439(87)80084-9

Xing Z, Yi C, Lin J et al (2021) Multi-component fault diagnosis of wheelset-bearing using shift-invariant impulsive dictionary matching pursuit and sparrow search algorithm. J Int Meas Conf. https://doi.org/10.1016/j.measurement.2021.109375

Xue J, Shen B (2020) A novel swarm intelligence optimization approach: sparrow search algorithm. Syst Sci Control Eng 8(1):22–34. https://doi.org/10.1080/21642583.2019.1708830

Yin S, Ding S, Haghani A et al (2012) A comparison study of basic data-driven fault diagnosis and process monitoring methods on the benchmark tennessee eastman process. J Process Control 22(9):1567–1581. https://doi.org/10.1016/j.jprocont.2012.06.009

You Z, Lu C (2018) A heuristic fault diagnosis approach for electro-hydraulic control system based on hybrid particle swarm optimization and levenberg-marquardt algorithm. J Ambient Intell Humaniz Comput. https://doi.org/10.1007/s12652-018-0962-5

Zeng N, Zhang H, Liu W et al (2017) A switching delayed PSO optimized extreme learning machine for short-term load forecasting. Neurocomputing 240:175–182. https://doi.org/10.1016/j.neucom.2017.01.090

Zeng N, Qiu H, Wang Z et al (2018) A new switching-delayed-PSO-based optimized SVM algorithm for diagnosis of Alzheimer's disease. Neurocomputing 320:195–202. https://doi.org/10.1016/j.neucom.2018.09.001

Zhang C, Ding S (2021) A stochastic configuration network based on chaotic sparrow search algorithm. Knowl Based Syst. https://doi.org/10.1016/j.knosys.2021.106924

Zhang Y, Ye D, Liu Y (2018) Robust locally linear embedding algorithm for machinery fault diagnosis. Neurocomputing 273:323–332. https://doi.org/10.1016/j.neucom.2017.07.048

Zhang D, Xu H, Wang Y et al (2021) A whale optimization algorithm based on embedding circle mapping and orifice imaging opposite learning with dimension by dimension. Control Decis 36(5):1173–1180. https://doi.org/10.13195/j.kzyjc.2019.1362

Zheng W, Zhao L, Zou C (2004) An efficient algorithm to solve the small sample size problem for lda. Pattern Recogn 37:1077–1079. https://doi.org/10.1016/j.patcog.2003.02.001

Zhou C, Chen Y (2006) Improving nearest neighbor classification with cam weighted distance. Pattern Recogn 39:1–11. https://doi.org/10.1016/j.patcog.2005.09.004

Zhu Y, Yousefi N (2021) Optimal parameter identification of pemfc stacks using adaptive sparrow search algorithm. Int J Hydrogen Energy 46(14):9541–9552. https://doi.org/10.1016/j.ijhydene.2020.12.107