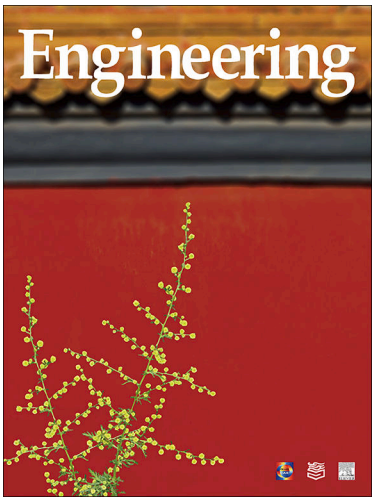
Journal Pre-proofs



Feature Article

PLSaoNET: A Generalized ANN Model Under PLS Statistical Constraints for Industrial Sensing

Lanxiang Sun, Tong Chen, Haibin Yu, Peng Zeng, Peng Zhang, Lifeng Qi, Yong Xin, Liming Zheng, Yang Zhou

PII: S2095-8099(26)00249-3
DOI: https://doi.org/10.1016/j.eng.2026.01.032
Reference: ENG 2328

To appear in: *Engineering*

Received Date: 4 February 2024
Revised Date: 9 March 2025
Accepted Date: 9 January 2026

Please cite this article as: L. Sun, T. Chen, H. Yu, P. Zeng, P. Zhang, L. Qi, Y. Xin, L. Zheng, Y. Zhou, PLSaoNET: A Generalized ANN Model Under PLS Statistical Constraints for Industrial Sensing, *Engineering* (2026), doi: https://doi.org/10.1016/j.eng.2026.01.032

Research

AI for Process Manufacturing—Feature Article

PLSaoNET: A Generalized ANN Model Under PLS Statistical Constraints for Industrial Sensing

Lanxiang Sun [a,b,c,*], Tong Chen [a,b,c], Haibin Yu [a,b,c,*], Peng Zeng [a,b,c], Peng Zhang [a,c], Lifeng Qi [a,c], Yong Xin [a,c], Liming Zheng [a,c], Yang Zhou [a,b,c]

[a] *State Key Laboratory of Robotics, Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016, China*

[b] *University of Chinese Academy of Sciences, Beijing 100049, China*

[c] *Liaoning Liaohe Laboratory, Shenyang 110169, China*

* Corresponding authors.

E-mail addresses: sunlanxiang@sia.cn (L. Sun), yhb@sia.cn (H. Yu).

Abstract: The application of artificial neural network (ANN) models to achieve higher accuracy in industrial sensing has become a popular research topic in recent years. However, neural network models are purely data-driven multivariate "black-box" models, and the features extracted from the hidden layer have no actual physical meaning, making the performance of ANN-based sensing models unstable and difficult to practically apply at process industry sites. To address these challenges, this paper proposes a generalized ANN model called the partial least squares (PLS)-assisted optimization network (PLSaoNET). PLSaoNET employs the PLS model to assist in determining the initialization weights of the network and the number of hidden-layer neurons. The subsequent training serves as a reoptimization process guided by the PLS regression result, enabling the network to incorporate statistical constraints and thereby reducing its reliance on data. In addition, to address the problem of uneven distributions of sample labels at industrial sites, this paper designs a stratified sampling method for network retraining. The efficiency and superiority of the proposed method are verified via two industrial sensing applications: the monitoring of iron grade in iron ore concentrate slurry samples based on laser-induced breakdown spectroscopy (LIBS) data, and the assessment of the quality of diesel fuels based on near-infrared (NIR) spectroscopy data. In comparison with a PLS regression model and a Xavier initialization-based backpropagation neural network (BPNN) model, PLSaoNET exhibits the best modeling accuracy and generalization performance. This work designs a complete theoretical framework to guide the determination of hyperparameters and specify the solution paths of the network, thereby satisfying the triple requirements of accuracy, robustness, and ease of use in industrial processes. The proposed model holds great potential for improving the accuracy and reliability of industrial sensing in production processes.

Keywords: Industrial sensing; Artificial neural network; PLS statistical constraints; PLSaoNET

1. Introduction

The process industry—a fundamental sector of national economies—is facing the challenge of upgrading to smart manufacturing [1]. As a real-time detection technology, industrial sensing can measure various parameters in a production process rapidly and accurately, providing massive amounts of real-time information for optimizing and regulating the production process [2–5]. Therefore, industrial sensing is a key technology in upgrading to smart manufacturing in the process industry and has attracted considerable attention from industry and academia [6–9]. Industrial sensing methods can

be roughly classified into two categories: mechanistic-based approaches and data-driven approaches [10–12]. Mechanistic-based approaches rely on accurate mathematical representations to describe the physical or chemical behavior of a system [13,14]. In contrast, data-driven approaches do not necessitate an in-depth understanding of the underlying mechanisms and are capable of modeling complex multivariate nonlinear relationships. Due to this advantage, data-driven approaches have become the mainstream method for complex industrial sensing [15].

In particular, due to their powerful learning and nonlinear modeling capabilities, as well as the continuous development of deep learning technologies, neural-network-based industrial sensing methods have been widely studied [16]. Such studies primarily concentrate on applying mature network technologies, including convolutional neural networks (CNNs), long short-term memory networks (LSTMs), and deep belief networks (DBNs), to industrial sensing systems within specific process industries. The objective is to utilize the powerful data analysis capabilities of neural networks to automatically capture complex nonlinear relationships in industrial data, thereby constructing higher-precision models [17]. For instance, CNNs have demonstrated significant capability for processing high-dimensional data. Long et al. [18] proposed a sensing framework utilizing a CNN model to facilitate the rapid and precise identification of mixed plastic wastes (MPWs), and Guo et al. [19] designed a multitask CNN model for bearing fault diagnosis and positioning tasks. LSTMs can capture temporal dependencies in data. For example, Soni et al. [20] presented an LSTM model for system failure prediction. DBNs, with their layer-wise feature learning, have also been used to model complex sensor data. Yuan et al. [21–23] conducted a series of industrial sensor modeling studies based on DBNs. By leveraging these advanced networks, researchers aim to reduce dependence on manual feature extraction and domain expertise, instead utilizing extensive industrial data to develop innovative solutions for sensing tasks under various process conditions.

However, industrial sensing in complex industrial environments presents many challenges. Firstly, industry's requirement for real-time sensing requires sensors to perform online *in-situ* measurements at the production sites of process industries, but the unprocessed measurement data are often accompanied by severe noise interference, resulting in poor data quality [24]. Secondly, the original industrial data of complex processes are usually high-dimensional, and not all features are conducive to the sensing task. Moreover, it is difficult to establish causal relationships for feature selection [25], and these features often have weak mechanistic and complex nonlinear relationships with the monitoring target [26]. Finally, simultaneous sampling and assaying at industrial sites is labor intensive, making the collection of labeled samples difficult [27]. These issues constrain the precision of mechanistic-based approaches and the reliability of data-driven approaches, and present a key challenge in the application of artificial neural networks (ANNs) in industrial sensing: namely, the problem of interpretability [28].

In recent years, numerous studies have focused on improving the interpretability of ANNs [29]. Such improvements can be categorized into three types:

(1) **Pre-modeling interpretability.** Here, the aim is to deeply understand the dataset by means of data analysis and preprocessing to extract crucial features [30]. However, due to the complexity of industrial processes and the black-box nature of neural networks, it is challenging to pre-estimate the sensitivity of a network on input features. Therefore, the results of feature engineering conducted independently of network training may not be optimal.

(2) **Post-modeling interpretability.** In this approach, techniques such as visualization [31] and feature relevance analysis [32] are primarily used to interpret the trained model. However, industrial applications impose stringent requirements on reliability, such that high-performance models should ideally be inherently interpretable, rather than results just being interpreted retrospectively. This is why linear methods, such as partial least squares (PLS), remain favored in industrial sensing [33].

(3) **Interpretable models.** Some models are intended to be directly interpretable, with physical information neural networks (PINNs) being a typical example [34]. However, PINNs require separate constraints to be set for different partial differential equations, which leads to a lack of generalizability. Moreover, PINNs add constraints to the objective function by fusing physical information, which can only indirectly constrain the network weight optimization and the representations in the hidden layers. As a result, PINNs do not fully address the issue of the hidden layers lacking clear physical meaning.

In addition to the interpretability problem, ANN-based industrial sensing suffers from the optimization complexity problem. In industrial applications, the quality and quantity of data are usually limited; thus, it is difficult to rely on big data to obtain stable model performance based on random initialization, as is often done in image and text processing.

Therefore, it is imperative to both ensure model performance and guarantee the interpretability, reliability, and ease of use of ANNs in practical industrial process applications. Although a great deal of work has focused on reducing optimization complexity [35,36] and augmenting the interpretability of neural networks, there is a scarcity of research specifically addressing the unique challenges and fundamental requirements encountered in the application of ANNs in industrial sensing processes. To tackle these challenges, we have designed a generalized ANN model with a comprehensive theoretical framework from the mechanistic level, named the PLS-assisted optimization network (PLSaoNET). The PLSaoNET approach is inspired by the PLS algorithm. As a statistical learning method, PLS has clear statistical interpretability, ease of model training, and a good ability to address high-dimensional small-sample problems. By solving the covariance matrix via singular value decomposition (SVD), several latent variables with orthogonal structures that best represent the information in the input data are gradually extracted, and these variables have the highest correlation with the outputs. The PLS model has a clear solution path, does not require much data or computational resources, and can be easily determined with only one hyperparameter—the number of latent variables [37,38].

In the proposed PLSaoNET model, PLS is utilized to assist in determining the initial weights and the number of hidden-layer neurons of the network, thereby transforming the neural network from a purely data-driven multivariate black-box model to a model comparable with a traditional statistical learning method (i.e., PLS) in terms of interpretability and reliability. Another key factor limiting the practical application of ANN-based models at industrial sites is the problem of uneven distributions of sample labels. Therefore, we also design a stratified sampling method for the network retraining process.

The principal contributions of this paper are summarized as follows: Firstly, a generalized PLSaoNET method is proposed for industrial sensing technology. This network is subject to PLS statistical constraints, inheriting the advantages of a PLS model while preserving the relearning and nonlinear modeling capabilities of a neural network. In this way, the interpretability and controllability of the model are enhanced, while high performance is maintained. Secondly, a complete theoretical framework for PLSaoNET modeling is proposed to guide the determination of the number of neurons and to elucidate the paths for determining the weighting parameters. Thirdly, the accuracy and robustness of the proposed method are demonstrated via numerous experiments using samples obtained from industrial sites. The experimental results show that the proposed method retains the excellent fitting ability of an ANN model while being more suitable for online industrial data characteristics and application requirements. Therefore, this method has broad application prospects for sensing tasks in complex industrial environments.

The rest of the paper is organized as follows. Section 2 briefly introduces the principles of the PLS and backpropagation neural network (BPNN) methods, outlines the motivation of this paper, and provides the complete theoretical framework and effective solution steps of the PLSaoNET model. Section 3 describes the study area of the validation data used in this paper: a case involving the laser-induced breakdown spectroscopy (LIBS)-based iron ore slurry grade monitoring technique and its data acquisition process (case 1), and a case involving the near-infrared (NIR) spectroscopy-based diesel fuel quality assessment technique and its data characterization (case 2). Section 4 presents a comprehensive test using LIBS data and NIR data as examples to validate the effectiveness of the PLSaoNET method. Section 5 provides conclusions and presents a future research outlook.

2. Theory and method

This section briefly reviews the principles of the PLS method and the BPNN model and introduces the derivation of PLSaoNET. Assume that $\boldsymbol{X}=[x_1, x_2,\ldots, x_n,\ldots, x_N]^{\mathrm{T}} \in \mathbf{R}^{N\times D}$ is the input of the regression model with N samples and D dimensional features, where x_n denotes the nth sample. $\boldsymbol{Y}=[y_1, y_2,\ldots, y_n,\ldots, y_N]^{\mathrm{T}} \in \mathbf{R}^{N\times 1}$ is the output of the regression model.

2.1. Partial least squares

PLS remains a popular method for practical applications in industrial sensing due to its simplicity, effectiveness, and ease of implementation. It is based on a linear relationship between the input and output:

$$\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{W} + \varepsilon \quad (1)$$

where $\boldsymbol{W}$ is the weight matrix that maps the input $\boldsymbol{X}$ to the output $\boldsymbol{Y}$, and ε is error.

PLS progressively extracts a set of latent variables $\boldsymbol{T}=[t_1, t_2,\ldots, t_h,\ldots, t_H]$ by means of linear mapping: $\boldsymbol{T}=\boldsymbol{X}\boldsymbol{W}^{(1)}$, where H is the number of latent variables, and $\boldsymbol{W}^{(1)}$ is the weight matrix that maps the input $\boldsymbol{X}$ to the latent variable matrix $\boldsymbol{T}$. The transformation process of $\boldsymbol{T}$ to output $\boldsymbol{Y}$ is given by $\boldsymbol{Y}=\boldsymbol{T}\boldsymbol{W}^{(2)}$, where $\boldsymbol{W}^{(2)}$ is the weight matrix that maps $\boldsymbol{T}$ to $\boldsymbol{Y}$. During the process of transformation, $\boldsymbol{T}$ should carry as much information as possible about the variation in $\boldsymbol{X}$; that is, $\max \operatorname{var}(\boldsymbol{T}) = \max \operatorname{var}\left(\boldsymbol{X}\boldsymbol{W}^{(1)}\right)$. Meanwhile, $\boldsymbol{T}$ should maintain maximum correlation with output $\boldsymbol{Y}$; that is, $\max \rho(\boldsymbol{T},\boldsymbol{Y}) = \max \rho\left(\boldsymbol{X}\boldsymbol{W}^{(1)},\boldsymbol{Y}\right)$. Therefore, the optimization objective of the PLS algorithm is as follows:

$$\max \operatorname{var}\left(\boldsymbol{X}\boldsymbol{W}^{(1)}\right) * \max \rho\left(\boldsymbol{X}\boldsymbol{W}^{(1)},\boldsymbol{Y}\right) = \max \operatorname{cov}\left(\boldsymbol{X}\boldsymbol{W}^{(1)},\boldsymbol{Y}\right) \quad (2)$$

The strength of PLS as a regression tool lies in its ability to decompose a multivariate regression problem into H uncorrelated univariate regression analyses. The specific calculation process is shown in Fig. 1. In the solution process, the latent variable matrix $\boldsymbol{T}$ and the weight matrix $\boldsymbol{W}^{(1)}$ are computed by the SVD of the $\boldsymbol{X}^{\mathrm{T}}\boldsymbol{Y}$ matrix, and the weight matrix $\boldsymbol{W}^{(2)}$ is obtained by the ordinary least squares (OLS) fit between $\boldsymbol{Y}$ and $\boldsymbol{T}$. The representation of $\boldsymbol{T}$ for $\boldsymbol{X}$ is obtained by $\boldsymbol{X}=\boldsymbol{T}\boldsymbol{P}^{\mathrm{T}}$, where $\boldsymbol{P}$ is the weight matrix that reconstructs $\boldsymbol{X}$ from the extracted latent variables $\boldsymbol{T}$. This weight matrix $\boldsymbol{P}$ is obtained by the OLS fit between $\boldsymbol{X}$ and $\boldsymbol{T}$. By iteratively regressing the residuals of the representation of $\boldsymbol{X}$ obtained from previous univariate regressions and combining the Schmidt orthogonality to ensure that the new latent variable does not overlap with the information of the previous latent variables, the most valuable information in $\boldsymbol{X}$ is extracted until the number of latent variables reaches the settled value.

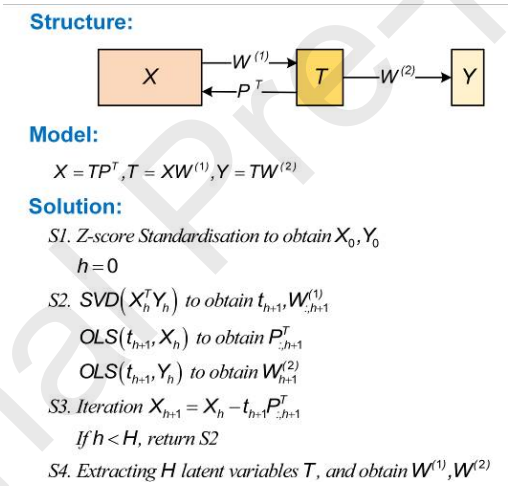


Fig. 1. Schematic diagram and solution process of the PLS model.

2.2. The backpropagation neural network

The BPNN is the most fundamental feed-forward neural network. It is named after its learning algorithm, the error backpropagation algorithm [39]. The BPNN is composed of an input layer, multiple hidden layers, and an output layer. The hidden layer is responsible for performing feature extraction and nonlinear transformation, and the output layer exports the predicted result. If the input $\boldsymbol{X}$ of the hth neuron of a hidden layer is D-dimensional, its output $\boldsymbol{O}_h$ can be expressed by Eq. (3):

$$\boldsymbol{O}_h = F_{\mathrm{act}}\left(\sum_{d=1}^{D} \boldsymbol{X}_{nd}\boldsymbol{W}_{dh} + b_h\right) \quad (3)$$

where $\boldsymbol{X}_{nd}$ is the d-dimensional feature of the nth sample, $\boldsymbol{W}_{dh}$ denotes the weight value of the d-dimensional input of this neuron, and b_h is the bias value. F_{act} denotes the activation function. The hyperbolic tangent (tanh) function is mostly chosen for regression applications and was also selected for this paper.

The weight and bias values are randomly initialized. During the training process, both the input data $\boldsymbol{X}$ and the corresponding output data $\boldsymbol{Y}$ are simultaneously fed into the network. Subsequently, the network computes the predicted output through the forward propagation process and compares it with the actual output $\boldsymbol{Y}$. The network then assesses the overall error between the predicted and actual outputs. For the regression problem, a linear activation function is always employed for the output layer. For a BPNN model with a single hidden layer, the loss function for optimal training of the neural network is shown in Eq. (4). The weight and bias values are updated based on the loss function $\boldsymbol{L}(w)$ using iteration algorithms such as the stochastic gradient descent (SGD) algorithm and the Adam algorithm. This process begins at the output layer and propagates backward through the layers of the network.

$$\boldsymbol{L}(w) = \frac{1}{2}(\boldsymbol{Y} - \boldsymbol{P})^2 = \frac{1}{2}\left\{\boldsymbol{Y} - \left[F_{\text{act}}\left(\boldsymbol{X}\boldsymbol{W}^{(1)} + b^{(1)}\right)\boldsymbol{W}^{(2)} + b^{(2)}\right]\right\}^2 \quad (4)$$

Multiple iterations are performed before $\boldsymbol{L}(w)$ is less than the expected value or the maximum threshold of iterations is reached. This process produces a set of finalized weight and bias values constituting a trained network.

2.3. The PLS-assisted optimization network

As shown in Fig. 1, a PLS model can be considered a feed-forward BPNN network with one hidden layer, which differs from a general BPNN in the following three ways.

(1) The training strategy of a BPNN model relies on the backpropagation of the output error, and the objective function is equivalent to maximizing the correlation between the model output and the reference label; in comparison, PLS takes into account both the expressiveness of the latent variables to the input and the correlation to the output. Therefore, in PLS, the objective function is the product of the variance and the correlation, also known as the covariance. The optimization objective of PLS directly constrains the weight solution of the hidden layer, whereas the constraint of BPNN on the hidden layer is indirect, so the hidden layer has no actual physical meaning.

(2) PLS directly employs the SVD of $\boldsymbol{X}^{\mathrm{T}}\boldsymbol{Y}$ to obtain the model weights and the latent variables (hidden-layer variables), bringing the advantage of a clear mathematical representation and optimization path. The disadvantage is that each latent variable is only the current optimal value and lacks a reoptimization process. In contrast, BPNN is based on mini-batch training with optimization algorithms such as SGD, which allows the simultaneous adjustment of the hidden-layer neurons.

(3) The PLS model can fit only linear relationships and has limitations in accuracy when applied to sensor modeling in modern industrial processes, which are characterized by intricate nonlinear relationships between sensor inputs and outputs. In contrast, the BPNN can introduce nonlinearity through the activation function and achieve greater accuracy in fitting the actual input–output relations through the combination of neurons.

Therefore, we propose the PLSaoNET model, which employs PLS to determine the initialization weights of the network and the number of hidden-layer neurons. A schematic diagram of the PLSaoNET model is shown in Fig. 2.

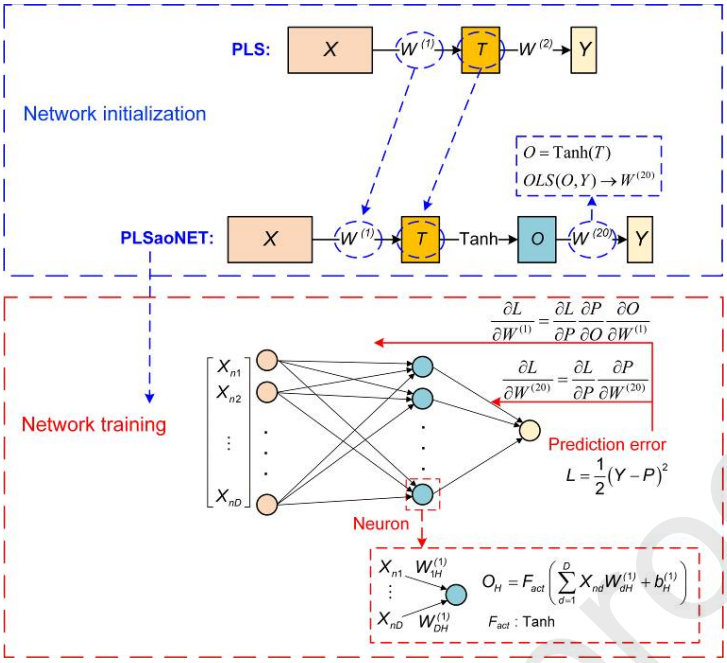


Fig. 2. Schematic diagram of the PLSaoNET model.

2.3.1. Network initialization

For the initialization process of the PLSaoNET model, we first establish the optimal PLS model. The number of latent variables H of the PLS model is optimized by the validation (or cross-validation) set, and the PLS model is rebuilt under the optimal H. The weight matrix $\boldsymbol{W}^{(1)}$ is obtained, which represents the transformation from input $\boldsymbol{X}$ to latent variables $\boldsymbol{T}$. The optimal number of latent variables H determined by the PLS model directly corresponds to the number of neurons in the hidden layer.

For the parameter initialization of the hidden layer, $\boldsymbol{W}^{(1)}$ is directly utilized as the initial weight, and the hidden-layer biases are initialized to $\mathbf{0}$. In the parameter initialization of the output layer, since this network adds a nonlinear transformation, the relationship between the hidden-layer variables and the target output $\boldsymbol{Y}$ is no longer $\boldsymbol{Y}=\boldsymbol{T}\boldsymbol{W}^{(2)}$. To solve this problem, this work first obtains the $\boldsymbol{O}$ matrix by the tanh transformation of the $\boldsymbol{T}$ matrix. The OLS fit is then performed on $\boldsymbol{Y}$ and $\boldsymbol{O}$ to obtain the weights $\boldsymbol{W}^{(20)}$ and bias $b^{(2)}$, which are utilized for the weight initialization and bias initialization of the output layer, respectively.

After initialization, the retraining of the weight and bias parameters can be performed by the backpropagation algorithm. The reoptimized process is based on the PLS result, which ensures the validity of the starting point of the training while simultaneously making sure that a better result than the PLS one can be obtained.

2.3.2. Network training

When industrial sensing technology is utilized in the process industry, the modeling data is obtained directly from the industrial site. Aside from the issues discussed above, such as small sample sizes, high dimensionality, and high noise in measurable variables, sample labels from industrial processes often present the problem of uneven distributions and lack of gradients. Consequently, the network may not be able to extract the statistical regularity accurately and is prone to overfitting, leading to poor performance on new data. Moreover, neural network training is based on mini-batch gradient descent. During the mini-batch training process, an uneven distribution of sample labels leads to large fluctuations in the loss function, thus affecting its convergence.

Therefore, we also design a network training method based on stratified sampling of labels and stochastic gradient descent (SSL–SGD). The details of the SSL–SGD method are described in Algorithm 1. For a training set with N samples,

we first sort the labels of the samples in ascending order. Based on the preset number of strata M (where $M = 5$ in this paper), we need to calculate $M-1$ quantiles. These quantiles are determined by selecting the label values located at the 20%, 40%, 60%, and 80% positions after sorting. The quantiles divide the labels into five equal-width intervals, with each interval corresponding to a stratum. Subsequently, we assign each sample to the corresponding stratum based on the magnitude relationship between its label value and the quantiles. For each iteration, if Mbs is the mini-batch size used for training, then the mini-batch training process will be repeated N/Mbs times. Mbs/M samples are taken separately from each stratum to compose the current mini-batch for SGD training.

Algorithm 1. PLSaoNET.

Input: Training dataset $\boldsymbol{D}_T = \{\boldsymbol{X}_T, \boldsymbol{Y}_T\}$

Validation dataset $\boldsymbol{D}_V = \{\boldsymbol{X}_V, \boldsymbol{Y}_V\}$

Parameters: Number of latent variables H

Mini-batch size Mbs

Learning rate Lr

Number of iterations epoch = {1, 2, …, Epo}

Initialization:

Scale $\boldsymbol{X}_T$ and $\boldsymbol{Y}_T$ to have zero mean and unit variance

Call the PLS algorithm with the number of latent variables H

Obtain $\boldsymbol{T}$, $\boldsymbol{W}^{(1)}$

Obtain $\boldsymbol{O} = \tanh(\boldsymbol{T})$

Obtain $\boldsymbol{W}^{(20)}$, $b^{(2)} = \mathrm{OLS}(\boldsymbol{O}, \boldsymbol{Y}_T)$

Initialize the network weight parameters as $\boldsymbol{W}^{(1)}$ and $\boldsymbol{W}^{(20)}$

Initialize the network bias parameters as $\boldsymbol{0}^{(1)}$ and $b^{(2)}$

Reoptimized: SSL–SGD

Stratify the training dataset based on labels:

Sort the $\boldsymbol{Y}_T$

Calculate the quantiles: $y_{q1}, y_{q2}, \ldots, y_{q(M-1)}$

for $\{x_i, y_i\}$ in $\{\boldsymbol{X}_T, \boldsymbol{Y}_T\}$ **do**

if $y_i \le y_{q1}$ **then** $\{x_i, y_i\} \in \boldsymbol{D}_{T1}$

else if $y_i > y_{q1}$ **and** $y_i \le y_{q2}$ **then** $\{x_i, y_i\} \in \boldsymbol{D}_{T2}$

…

if $y_i > y_{q(M-1)}$ **then** $\{x_i, y_i\} \in \boldsymbol{D}_{TM}$

end if

end for

$\boldsymbol{D}_T = \{\boldsymbol{D}_{T1}, \boldsymbol{D}_{T2}, \ldots, \boldsymbol{D}_{TM}\}$

Training process:

for epoch in {1, 2, …, Epo} **do**

Randomly shuffle the data in $\boldsymbol{D}_{T1}, \boldsymbol{D}_{T2}, \ldots, \boldsymbol{D}_{TM}$

for _ in *N*/Mbs **do**

for *m* in *M* **do**

Sample subset d_{tm} from $\boldsymbol{D}_{Tm}$

end for

Mini-batch $\boldsymbol{B}_{epoch} = d_{t1} \cup d_{t2} \cup \cdots \cup d_{tM}$

Update $\boldsymbol{W}^{(1)}$, $\boldsymbol{W}^{(20)}$, $b^{(1)}$, and $b^{(2)}$ by $\boldsymbol{B}_{epoch}$ based on the backpropagation

end for

Obtain $\boldsymbol{P}_T$ and $\boldsymbol{P}_V$

Calculate $e_T = \text{error}(\boldsymbol{P}_T, \boldsymbol{Y}_T)$

Calculate $e_V = \text{error}(\boldsymbol{P}_V, \boldsymbol{Y}_V)$

if $\text{times}(e_T \geq \min(e_T)) \geq 5$ and $\text{times}(e_V \geq \min(e_V)) \geq 5$

end for

Output: optimum PLSaoNET model

To minimize the risk of overfitting, this work also incorporates an early stop strategy. The maximum threshold of iterations is set to a substantial value (500 in this work), and training termination is triggered when the training loss and validation loss do not decrease for several consecutive times (5 times in this paper). The optimal neural network will be preserved before the termination of the training.

2.3.3. Procedures of the PLSaoNET method

The PLSaoNET method is described by Algorithm 1, where the number of latent variables (*H*), mini-batch size (Mbs), and learning rate (Lr) are hyperparameters that need to be further optimized by means of validation-set performance.

3. Study area and data acquisition

The efficiency and superiority of the proposed method are evaluated through two industrial sensing applications: a LIBS-based iron ore slurry grade monitoring technique and an NIR spectroscopy-based diesel fuel quality detection technique. This section introduces the data acquisition process and data characterization.

3.1. LIBS spectra of iron ore slurry samples

The first case is the LIBS data. LIBS is a direct measurement technique that relies on physical processes and spectral analysis [40]. It is based on the principle of focusing high-energy pulsed laser radiation on the surface of a sample to produce plasma and then analyzing the emission spectra of the plasma to obtain information about the composition and content of the sample [41]. LIBS has many advantages, such as fast analysis speed, reduced sample preparation, and onsite operability, and is expected to be used for detecting material composition and product quality for real-time production control at process industry sites.

However, the accuracy of LIBS analysis has been a major factor limiting its widespread use. This is mainly due to the nonlinear phenomena caused by the matrix effect and self-absorption effect [42–44]. Current understanding of the mechanisms underlying the matrix effect is limited, and, although the self-absorption effect is better understood, there are no reliable methods for detecting and controlling either of these effects. Therefore, the use of advanced data-driven approaches to compensate for matrix effects has been a hot research topic [45–47]. Among the modeling methods developed for LIBS data analysis, an important category is based on ANNs [48–51].

Fig. 3 shows the schematic diagram of the iron ore flotation process (Fig. 3(a)), the LIBSlurry analyzer developed by our team (Fig. 3(b)), and the online experimental setup inside the analyzer (Fig. 3(c)). The analyzer has been installed in the iron ore flotation plant for more than two years and has been previously described in the literature [52]. As shown in Fig. 3(b), a sampling valve is installed on the iron ore concentrate slurry pipeline to divert the slurry into a sampling vessel. Inside the analyzer, as shown in Fig. 3(c), a laser beam focuses on the surface of the slurry flow in this vessel, generating a plasma. The spectrometers subsequently acquire the plasma emission spectra. Finally, a pretrained quantitative analysis model is utilized to convert the spectral features to the iron (Fe) grade, thus realizing the real-time monitoring of the iron ore concentrate slurry grade.

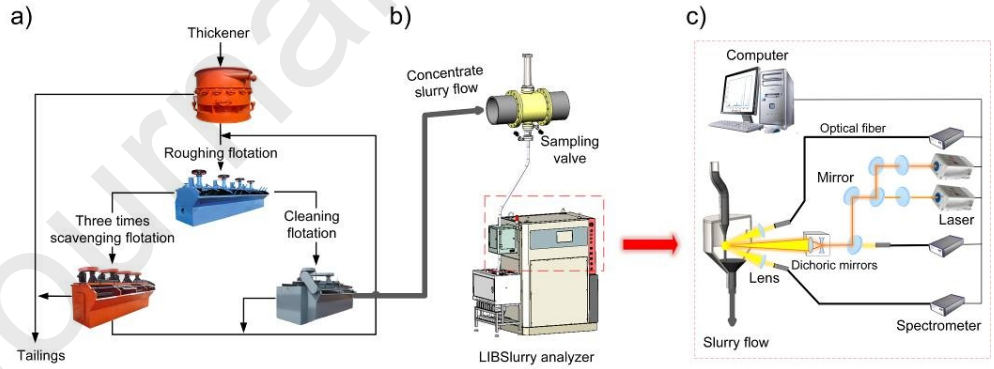


Fig. 3. LIBS-based Fe grade sensing in iron ore concentrate slurry during flotation. (a) Schematic diagram of the iron ore flotation process; (b) diagram of the LIBSlurry analyzer installed at the flotation site; (c) diagram of the optical device inside the analyzer.

3.1.1. Sample preparation

As flotation is a continuous production process, it was necessary to conduct the performance validation of the data-driven model over an extended period and with a sufficient number of samples. To achieve this, data collection was conducted in the field for nearly two months, from June to July 2022, and a total of 380 labeled samples were obtained. During the online measurement process, the analyzer operates continuously, facilitating the LIBS-based measurement at a

frequency of three times per hour. Each measurement takes approximately 2 min, as the analyzer averages over 1000 laser pulses to enhance signal stability and reduce noise.

In contrast to the online measurement, the offline sampling and assay process requires a minimum of 2 h between sampling. During the sampling process, a 150 mL slurry sample was taken from the production line and subsequently sent to the onsite laboratory. The laboratory utilized the titanium trichloride reduction potassium dichromate titration method (GB/T 6730.65–2009) to provide a reference value of Fe grade (with an error margin of 0.5%).

3.1.2. Data preprocessing

The yellow line in Fig. 4 shows a typical LIBS spectrum of the slurry sample. The spectrum covers a wavelength range of 227–784 nm and contains a total of 6116 dimensional features. According to the National Institute of Standards and Technology (NIST) database, the primary spectral features are the emission lines of the main elements in the slurry samples, including iron (Fe), silicon (Si), calcium (Ca), sodium (Na), hydrogen (H), oxygen (O), and nitrogen (N) [53]. Of these, the important elements associated with the iron grade in the slurry sample are Fe, Si, Ca, and Na. Fe is the analytical element that is theoretically most relevant to the grade of the iron ore concentrate slurry and is also the matrix element in the slurry samples. Regarding Si, the associated minerals in iron ore are mainly SiO_2, the presence of which affects the flotation process and thus the grade of the iron ore concentrate. Ca and Na are additives used in the flotation process and are related to the final iron grade; the remaining Ca and Na in the slurry also affect the LIBS spectra.

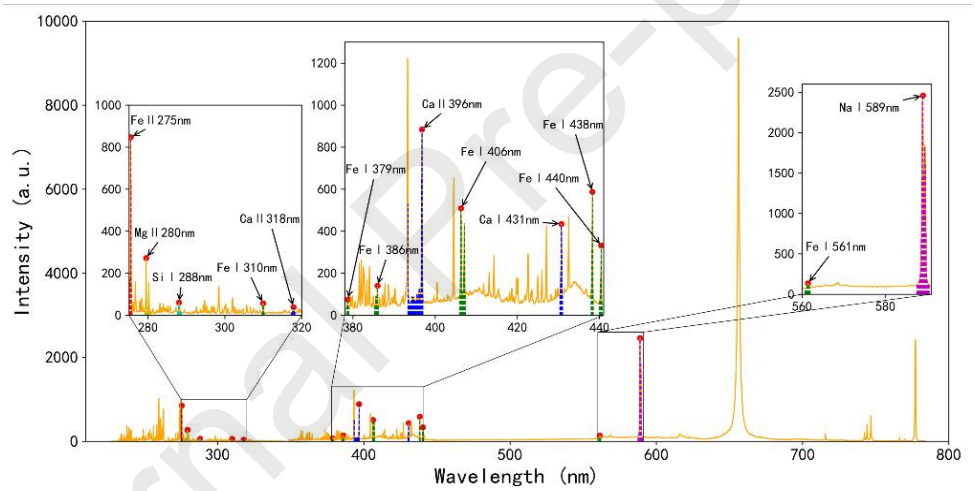


Fig. 4. A LIBS spectrum and the position of the selected feature.

Since the LIBS spectra have more than six thousand dimensions and the labeled dataset has only 380, too much dimensionality of this input will lead to the overfitting of subsequent data-driven models. Therefore, we first selected 285 dimensional characteristic spectral lines as input variables for subsequent modeling. The selected wavelength features are shown in Fig. 4, and the details are shown in Table 1.

Table 1

Information on the selected feature.

No.	Wavelength position (nm)	Index position	Length	Element

1	275.06–275.81	731–743	13	Fe II
2	279.13–280.54	797–820	24	Mg II
3	287.71–288.61	938–953	16	Si I
4	309.67–310.24	1314–1323	10	Fe I
5	317.31–318.20	1449–1465	17	Ca II
6	378.47–378.97	2563–2572	10	Fe I
7	385.30–386.22	2688–2705	18	Fe I
8	393.40–396.99	2840–2909	70	Ca II
9	405.94–407.32	3085–3113	29	Fe I
10	430.35–430.96	3604–3618	15	Ca I
11	437.98–438.44	3781–3792	12	Fe I
12	440.03–440.69	3830–3846	17	Fe I
13	560.75–561.92	4095–4104	10	Fe I
14	587.54–590.41	4306–4329	24	Na I
Total	—	—	285	—

3.2. NIR spectra of diesel fuel samples

The second experiment focused on diesel fuel quality evaluation based on NIR spectral data. The dataset[†], collected from a US Army-sponsored project at the Southwest Research Institute (SWRI) [54,55], comprises 395 diesel fuel samples, originating from the same oil field. The NIR spectral data for each sample span the wavelength range from 900 to 1700 nm, encompassing 401 wavelength features. These data were utilized to predict the diesel fuel viscosity (measured in centistokes (cSt)), a crucial parameter for monitoring diesel product quality.

[†] https://eigenvector.com/data/SWRI/index.html

From a process perspective, diesel viscosity reflects the combined influence of all the components in the diesel fuel, implying that there may not be a distinct spectral region directly correlated with viscosity. From a modeling standpoint, this dataset also presents the challenge of an insufficient sample size relative to the spectral dimensions, potentially leading to challenges such as overfitting or reduced model accuracy.

4. Results and discussion

To fully verify the effectiveness of our proposed method, we used the same dataset to construct two comparative models: a PLS regression model and a BPNN model with random initialization. In the first part of this section, we showcase the high accuracy and reliability of the proposed PLSaoNET model by using the iron ore slurry grade monitoring task based on LIBS data as an example. Then, in the second part, we further validate our work through the diesel fuel quality assessment task using NIR data. Finally, we demonstrate how the proposed method improves model performance, specifically by utilizing a post-modeling interpretability method based on feature relevance analysis.

The root mean square error (RMSE) and determination coefficient (R^2) were employed as the evaluation metrics. These are defined as follows:

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{n=1}^{N}\left(\hat{y}^n - y^n\right)^2} \quad (5)$$

$$R^2 = 1 - \frac{\sum_{n=1}^{N}\left(\hat{y}^n - y^n\right)^2}{\sum_{n=1}^{N}\left(\bar{y} - y^n\right)^2} \quad (6)$$

where y^n is the reference value of the nth sample, $\hat{y}^n$ is the predicted value of the nth sample, and $\bar{y}$ is the average reference value of all samples. In the subsequent evaluation of the model performance, we mainly examine four specific metrics: the RMSE of the training set (RMSEC), RMSE of the validation set (RMSEV), RMSE of the test set (RMSEP), and R^2 of the test set (R2P).

4.1. Case 1: Iron ore slurry grade monitoring

Based on the sample collection order, the first 300 iron ore concentrate slurry samples were randomly divided into a training set, a validation set, and a test set, at a ratio of 3:1:1. Thus, 180 samples were selected as the training set to determine the weight parameters, 60 samples were selected as the validation set to optimize the hyperparameters, and the other 60 samples were used as the test set to evaluate the model's generalization ability. The process of modeling and evaluation was repeated 100 times to fully investigate the precision and robustness of the different models. In addition, since flotation is a continuous production process, to match actual production, the last 80 samples were employed to provide a true picture of the ability of the different models to predict future samples.

4.1.1. PLS regression model

One of the dataset divisions is taken as an example to observe the detailed performance of the different models. For the PLS model, there is only one hyperparameter that needs to be determined—the number of latent variables (H). In this work, H was set to vary from 1 to 20 in step 1. The RMSEs of the training set and validation set are shown in Fig. 5(a). The optimal H is 13, and the performance of the corresponding optimal PLS model is shown in Fig. 5(b). The two black dashed lines in Fig. 5(b) are used to compare the deviation of the prediction values from the reference values; it can be seen that the deviation of the PLS model predictions from the reference values is generally large. The optimal RMSEs of the validation set and test set are 1.21% and 1.03%, respectively. It is difficult for the PLS model to meet the precision requirement of process control, as the prediction error needs to be less than 1%.

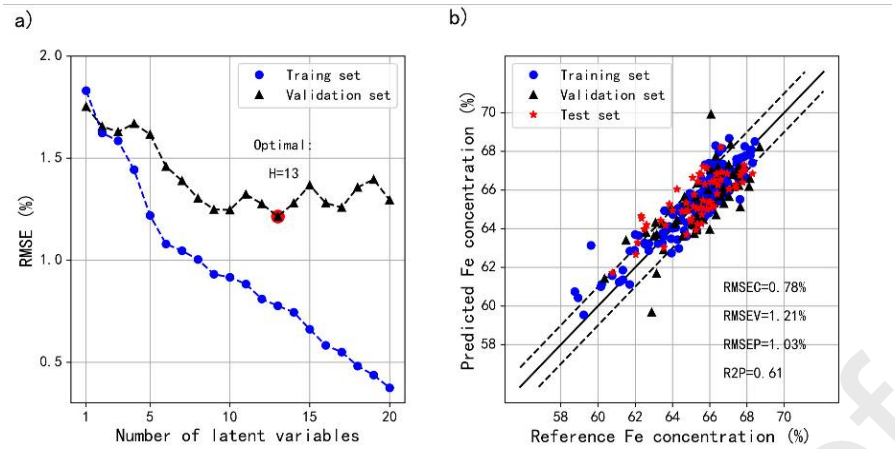


Fig. 5. PLS model performance. (a) Hyperparameter optimization process; (b) optimal PLS model result.

4.1.2. BPNN with random initialization

This section outlines the performance of a BPNN model based on random Xavier initialization under the same dataset division mode as that used in Section 4.1.1. There are many hyperparameters in the neural network; these generally include the learning rate, the number of layers, the number of neurons in each layer, the activation function of the neurons, the number of iterations, and the mini-batch size. To fully validate the performance of the proposed method, the BPNN was directly set as a single hidden-layer network, and the number of neurons in the hidden layer was the number of latent variables in the corresponding PLS model. Since it is a regression problem, the neurons employ the tanh activation function. Ultimately, only two parameters need to be determined: the mini-batch size (Mbs) and the learning rate (Lr).

The optional range of Mbs was determined first. Given the sample size and the fact that neural networks often set Mbs as a power of 2, we selected Mbs values of 4, 8, 16, 32, 64, and 128. Furthermore, we determined an optional range of Lr to ensure training efficiency and convergence. Since a too-small Lr may lead to inefficient training and delayed convergence, while an overly large Lr may cause oscillations and also affect convergence, we experimentally determined a range of Lr from 0.0001 to 0.0010, with a step size of 0.0001. Figs. 6(a) and (b) show the RMSEs of the model for the training set and validation set, respectively. The best hyperparameter combinations are Mbs = 64 and Lr = 0.0007; the performance of the corresponding optimal BPNN model is shown in Fig. 7.

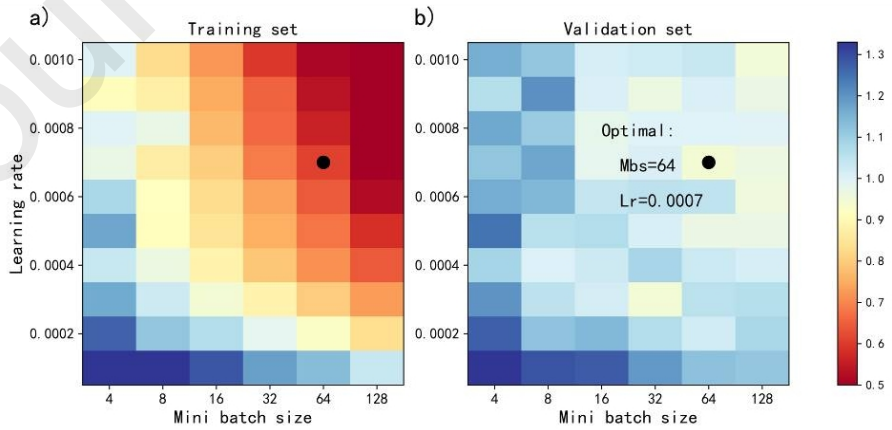


Fig. 6. The hyperparameter optimization process of the BPNN model. (a) Training set; (b) validation set.

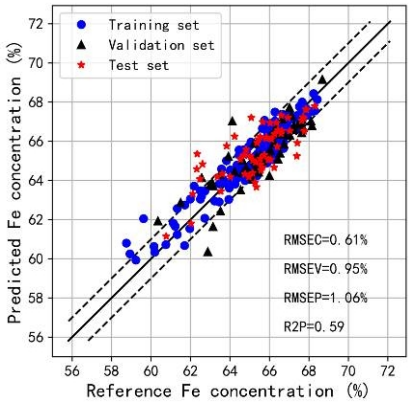


Fig. 7. The optimal result of the BPNN model.

The fitting performance of the training set and the validation set based on the BPNN model is significantly improved, with the RMSEC and the RMSEV reduced to 0.61% and 0.95%, respectively. In addition, it can be seen that the regression relationship fitted by the BPNN model is completely different from that of the PLS model, as the outlier (reference value of 66.07%) in the validation set based on the PLS model turns out to be in the normal range. The datapoints in Fig. 7 are significantly more compactly distributed between the error bands than those in Fig. 5(b), which suggests that it is logical to introduce nonlinearity to the model based on the tanh activation function. However, the RMSEP and the R2P are 1.06% and 0.59, respectively, which are worse than the results of the PLS model, and the network may have overfitted.

4.1.3. The PLSaoNET model

With the same hyperparameter optimization process as that used for the BPNN model, the PLSaoNET model proposed in this paper was built. The PLSaoNET model employs PLS to assist in completing the initialization to provide a definite starting point for subsequent iterative training. Moreover, an optimization method for iterative training based on stratified sampling is proposed to address the uneven sample distribution problem. The hyperparameter optimization process based on the training set and validation set is shown in Fig. 8. The optimal hyperparameter combination is Mbs = 32 and Lr = 0.0009; the corresponding model results are shown in Fig. 9, with RMSEC = 0.33%, RMSEV = 0.82%, RMSEP = 0.91%, and R2P = 0.70. The combined accuracies of this method with different sets are significantly better than those of the other two models. The R2P of the PLSaoNET model has increased to 0.70, indicating that the model fits a more correct regression relationship. The most important parameter, the RMSEP, has decreased to 0.91%, which indicates that the results of the proposed method are representative of the general situation in practical applications. The training process takes 1.63 s to achieve convergence, and the prediction of test data takes 0.0019 s. Thus, the model-based online prediction time is much shorter than the spectral measurement time, meeting production requirements.

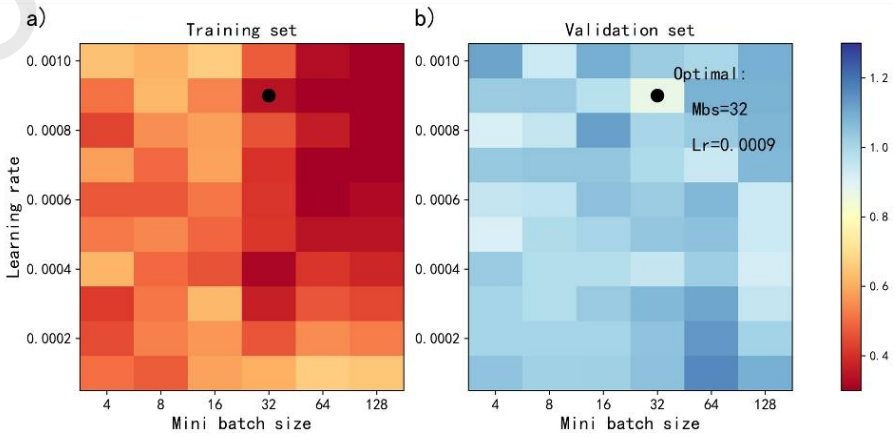


Fig. 8. The hyperparameter optimization process of the PLSaoNET model. (a) Training set; (b) validation set.

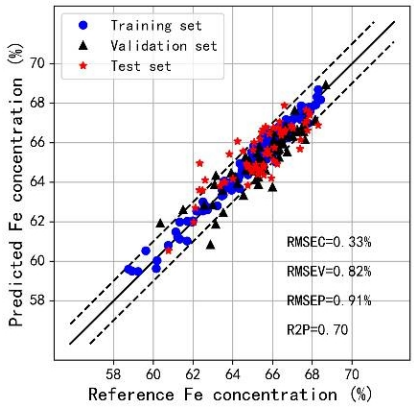


Fig. 9. The optimal result of the PLSaoNET model.

4.1.4. Comparison of the following precisions of different models

The industrial sensor results of the iron ore concentrate grade prediction task under different models are compared in Fig. 10. An examination and comparison of Figs. 10(a) and (c) reveal that the trend prediction of PLSaoNET is generally consistent with that of the PLS model, but it is smoother, which is consistent with the actual production situation. This result indicates that PLSaoNET has high reliability in predicting the iron ore concentrate grade. By observing and comparing Figs. 10(b) and (c), it can be found that the BPNN results have more anomalous predictions compared with the reference values. This indicates that the prediction results of the BPNN model have large uncertainty, which will seriously affect the control of the subsequent production process and may not be suitable for practical industrial site applications. Fig. 10(d) shows the probability density curves of the industrial sensor results under the different models. It can be seen that the probability density curve based on the PLSaoNET method is thinner and higher than those of the other methods, which indicates that the PLSaoNET method has higher accuracy and a smaller error range in predicting the iron ore concentrate grade.

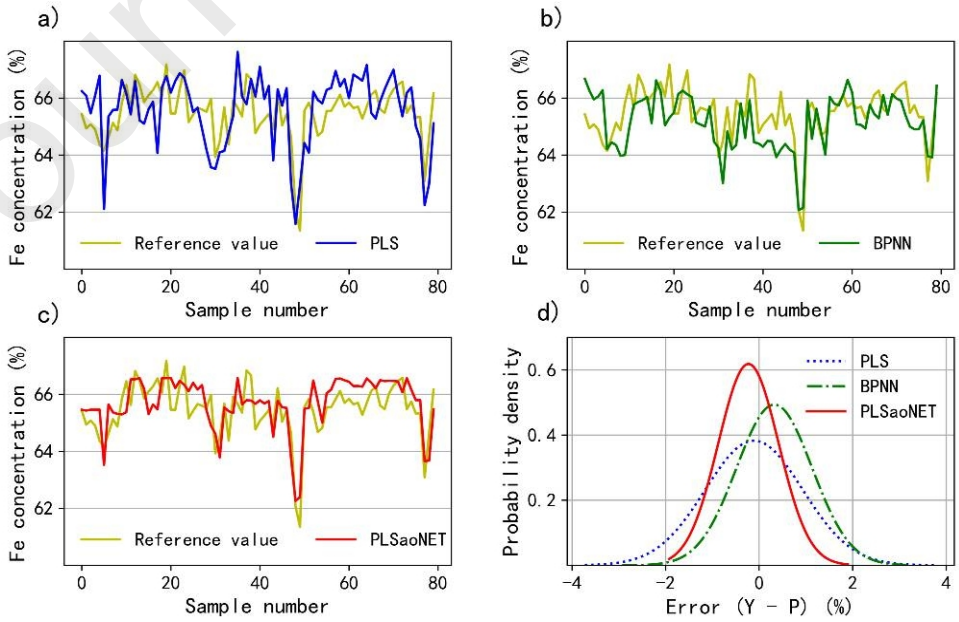


Fig. 10. Comparison of the industrial sensor results of the iron ore concentrate grade prediction task under different models. (a) PLS; (b) BPNN; (c) PLSaoNET; (d) probability density curves of error.

4.1.5. Comparison of the precision and robustness of 100 experiments

To comprehensively compare the performances of the three different models, the dataset random division process was repeated 100 times. Following the process described above, the PLS, BPNN, and PLSaoNET models were built. The precision and robustness of the different models are shown in Fig. 11, and the details are shown in Table 2. In Figs. 11(a)–(d), violin plots of the RMSEC, RMSEV, RMSEP, and R2P are respectively provided. For each model, the median value is represented by the white dot, while the horizontal thick black bar represents the interquartile range (IQR). The thin black line represents the nonoutlier range, which lies within 1.5 times the IQR away from the thick black bar. Additionally, the kernel density plot flanking the line shows the estimated distribution of the data [56]. In Figs. 11(e)–(h), the rankings of the different models calculated for all the division modes are shown, where the dots indicate the mean values and the black lines indicate the corresponding standard deviations.

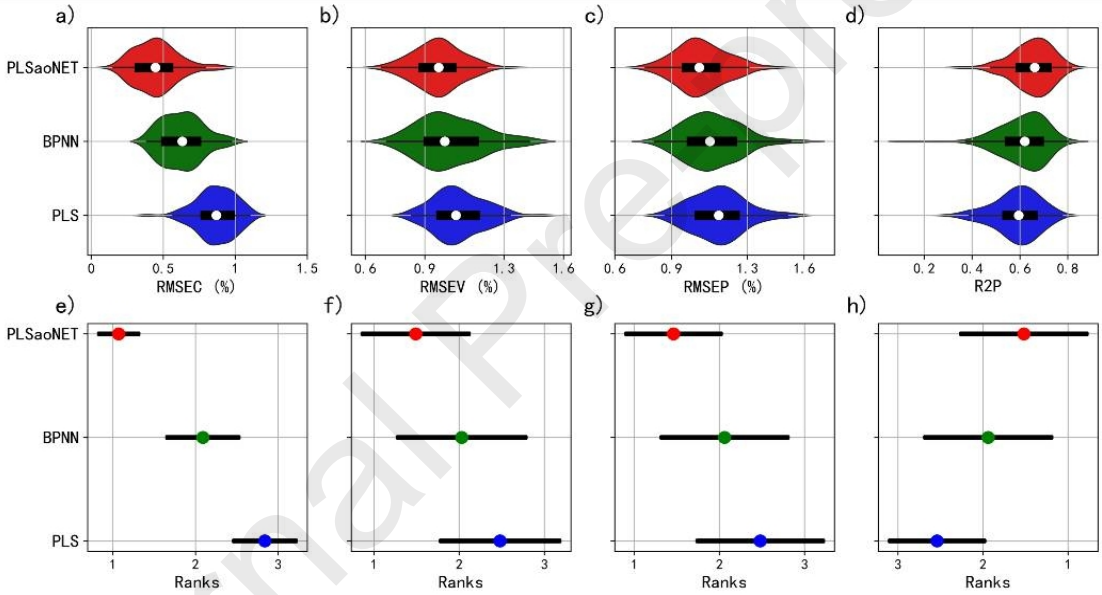


Fig. 11. (a–d) Violin plots of the performance of (a) RMSEC, (b) RMSEV, (c) RMSEP, and (d) R2P for different models under 100 experiments. (e–h) Corresponding mean values and standard deviations of the rankings for different models for different evaluation metrics.

In terms of precision, the results of the PLS model are the worst. Therefore, it can be concluded that the linear PLS model has difficulty accurately fitting the industrial sensor model between the LIBS spectra and the iron grade in the iron ore slurry grade prediction task. The BPNN model has moderate precision and the worst robustness; as shown in Table 2, it has the largest standard deviation values for the RMSEV, RMSEP, and R2P results among the three models. To reflect variability, the results in Table 2 are presented as mean ± standard deviation, highlighting the greater fluctuations in the BPNN model's predictions. This is because—as BPNN is a purely data-driven multivariate black-box model—the feature transformation process easily overfits the noise.

Table 2

Details of the evaluation metrics for different models under 100 experiments.

As in the results discussed earlier, PLSaoNET achieves the best optimal result in comparison with the other two models. As shown in Table 2, the proposed PLSaoNET method yields the best modeling precision, with an average RMSEC (mRMSEC) of 0.45%, which is 48.3% lower than that of the PLS regression model (0.87%) and 28.6% lower than that of the BPNN based on Xavier initialization (0.63%). It also exhibits the best generalization performance, with an average RMSEP (mRMSEP) of 1.06%, which is 7.8% lower than that of the PLS (1.15%) and 5.4% lower than that of the BPNN (1.12%). Moreover, it has the best robustness, with a standard deviation of the RMSEP (sRMSEP) of 0.132%, which is 5.7% lower than that of the PLS (0.140%) and 10.8% lower than that of the BPNN (0.148%). The reason for these excellent results is that the results of the PLSaoNET model are obtained by further training based on the PLS model; therefore, the PLSaoNET model combines the dual advantages of the PLS model, as it is interpretable and controllable, and the ANN model, as it introduces nonlinearity to make full use of the data.

4.2. Case 2: Diesel fuel quality assessment

Similarly, the dataset in the diesel fuel quality assessment case was randomly divided into three sets in a ratio of 3:1:1. That is, 236 samples were selected to build the model, 79 samples were employed to optimize the hyperparameters of the model, and the remaining 79 samples were used as an independent test set for evaluating the model's generalization ability. Since the dataset size and feature dimensions were similar to those of the LIBS-based iron ore slurry grade monitoring task, the selection of the mini-batch size (Mbs) was kept unchanged. The optimization range for the learning rate (Lr) was experimentally determined to be from 0.00002 to 0.00020, incremented in steps of 0.00002. The optimal model performances of the PLS, BPNN, and PLSaoNET models are shown in Fig. 12.

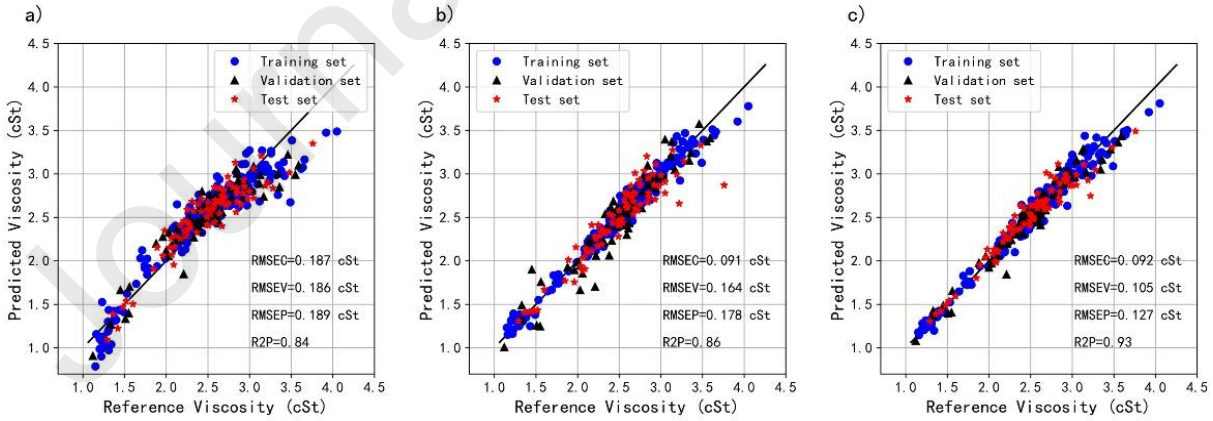


Fig. 12. The optimal results for the (a) PLS model, (b) BPNN model, and (c) PLSaoNET model.

A comparison of the results shown in Fig. 12 reveals that there is clearly a nonlinear relationship between the NIR spectra of diesel fuel and its viscosity. Therefore, the PLS prediction results exhibit deviation, resulting in the lowest accuracy among the three models. The BPNN model fits the nonlinear relationship, but it may overfit, resulting in the

appearance of certain outlier prediction points. In comparison, it can be seen that the PLSaoNET model achieves the best optimal modeling and testing results in comparison with the other two models.

4.3. Model interpretability

In this section, we take the iron ore slurry grade monitoring task as an example and introduce in detail how the proposed PLSaoNET can maintain a high performance while having the advantage of high reliability, compared with ordinary black-box networks. Firstly, the details of the regression process of the PLS model were presented in Section 4.1.1, including the relationship between the extracted latent variable LV*j* and the 285-dimensional input ***X***, as well as the relationship between LV*j* and the output ***Y*** (taking *j* = 1, 3, 5, 7, 9, 11, and 13 as examples). For the PLS model, LV*j* is the *j*th column of the latent variable matrix ***T***. Fig. 13(a) shows a schematic of the 285 dimensional spectral features. In Figs. 13(b)–(h), the weights for transforming the 285-dimensional input into the corresponding latent variable LV*j* are shown. Fig. 13(i) represents the contribution of the latent variable LV*j* to the output ***Y***, which is obtained by calculating the absolute value of the product of the corresponding LV*j* and its weight to ***Y***. The contributions are processed by means of max–min normalization. The Pearson correlation coefficients between the latent variable LV*j* and the output ***Y***: $r(\mathrm{LV}_j, \boldsymbol{Y})$ are calculated below.

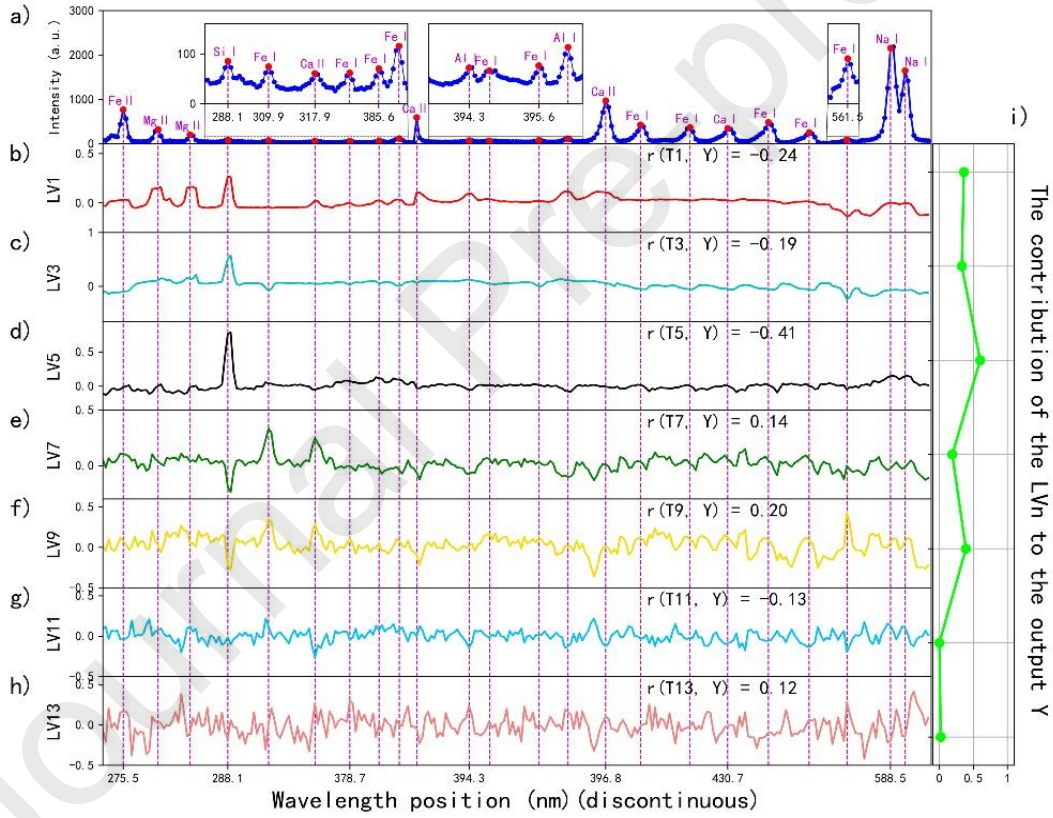


Fig. 13. Details of the regression process of the PLS model. (a) Schematic of the 285 dimensional spectral features; (b–h) weights for transforming the 285-dimensional input into the corresponding latent variable LV*j* with *j* = 1, 3, 5, 7, 9, 11, and 13, respectively; (i) the contribution of the latent variable LV*j* to the output ***Y***.

First, it can be seen that, as the number of latent variables increases, the noisy data begin to influence or even dominate the feature extraction process. The correlations between the latent variables LV1, LV3, and LV5 and the feature lines of elements such as Si, Mg, Al, Ca, and Na are clearer. All give very high attention to Si (Si I 288.1 nm), which coincides with the actual production, because the iron ore flotation process is the process of selecting iron and removing silicon. Due to the sample iron grade distribution of 60%–68%, the Fe feature lines are mostly saturated, and the regression model is mainly

built by other elements, such as Si lines. However, less attention is given to Si in LV7, and more attention is given to the weak Fe atomic lines (Fe I 309.9 nm); at the same time, the weight starts to oscillate at the positions of the nonfeature lines. Little attention is given to meaningful feature spectral line information after LV11, and most of what is extracted are background and noise. Furthermore, the maximum absolute value of the Pearson correlation coefficient between LV*j* and ***Y*** is 0.41 (LV5), which implies that the latent variables are less linearly correlated with ***Y***. The rule for establishing the PLS model involves two constraints: the largest linear correlation with ***Y*** and the most representative of the information of ***X***. However, the former constraint, which focuses on maximizing the correlation with ***Y***, has almost loses its significance. In addition, it can be observed that the latent variables LV11 and LV13 have the lowest contributions to the output, demonstrating the noise immunity of the PLS model as a linear model with an explicit solution path.

The details of the regression process of the BPNN model are also shown in Fig. 14. For the BPNN model, LV*j* is the *j*th column of the hidden-layer output matrix ***O***. Fig. 14(a) is the same as Fig. 13(a), and the value range of the vertical coordinate of Figs. 14(b)–(h) is the same as that of Figs. 13(b)–(h). It can be seen that the 5th neuron assigns some attention to the Si feature line, and the 7th neuron assigns some attention to the Ca feature line. However, unlike in the PLS model, where the dominant latent variables explicitly focus on the emission spectral line information of the feature elements, the feature extraction process in the middle layer of the BPNN model has difficulty distinguishing between background noise and emission spectral signals. It is a purely data-driven process of feature combination, which is consistent with the nature of the BPNN as a black-box model. Moreover, the Pearson correlation coefficients of the middle layer feature LV*j* with ***Y*** are –0.56, –0.31, 0.41, 0.05, 0.36, 0.07, and –0.68. The maximum absolute value is 0.68 (LV13), which is higher than that of the PLS model, indicating that—through the combination of neurons and the nonlinear transformation by the activation function—the neural network extracts features with a higher correlation with the output, so it achieves better modeling results than PLS. However, this approach does not achieve a better test set performance. Further observation of Fig. 14(i) reveals that LV1 and LV13, which have a higher correlation with ***Y***, contribute less to the output ***Y*** than LV2, LV9, and LV11, which have a lower correlation with ***Y***. This again illustrates the purely data-driven fitting process of the neural network model, which leads to incorrect attention being given to noisy information and, as a result, to the existence of some points where the prediction bias is larger than that of the PLS model.

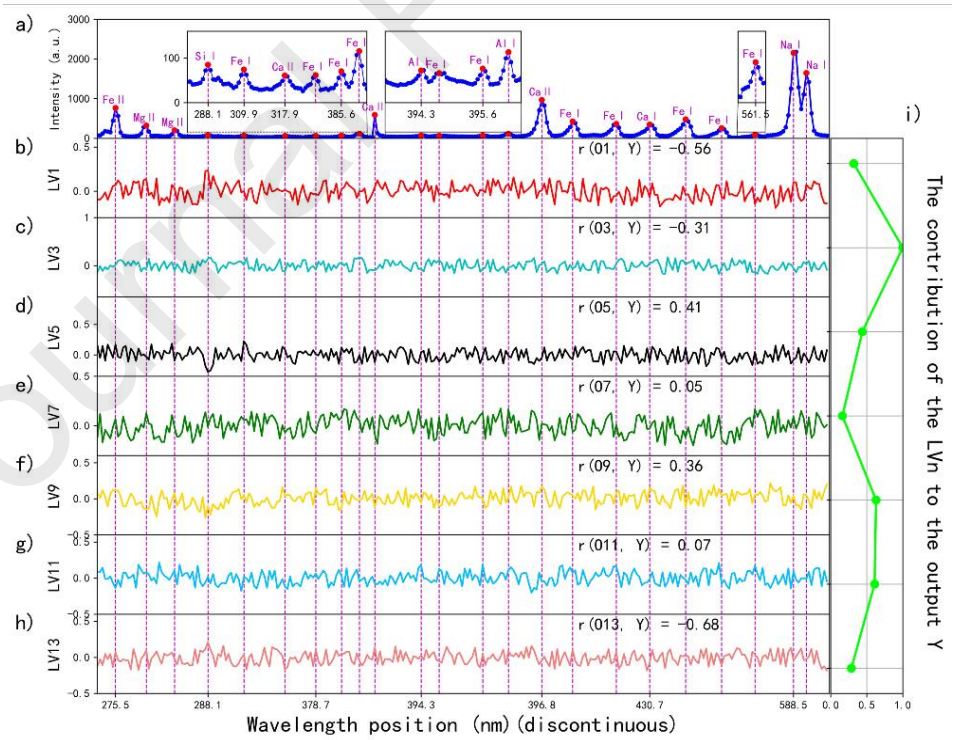


Fig. 14. Details of the regression process of the BPNN model.

Similarly, Fig. 15 provides the details of the regression process of the PLSaoNET model. Consistent with the BPNN model, LV*j* is the *j*th column of the hidden-layer output matrix ***O***. It can be seen that, compared with Fig. 13, LV1 undergoes significant change, increasing the attention given to the feature line of Ca (Ca II 396.8 nm). In addition, LV3, LV5, and LV9 retain their very high attention to Si (Si I 288.1 nm), and LV7 retains its very high attention to Fe (Fe I 309.9 nm) and increases the attention to the feature line of Al (Al I 394.3 nm). The transform weights between the inputs and latent variables also began to oscillate at non-signal positions, and it is clear that the model has learned a new feature combination based on PLS. The Pearson correlation coefficients of the middle layer feature LV*j* with ***Y*** are –0.19, –0.11, –0.42, 0.17, 0.29, –0.21, and 0.21. The maximum absolute value is 0.42 (LV5), which is higher than that of the PLS model but lower than that of the BPNN model. However, as shown in Fig. 15(i), the dominant role of LV5 in the transformation process from LV*j* to ***Y*** is retained, which effectively optimizes the solution process, improves the interpretability of the model, and reduces the risk of overfitting; therefore, the PLSaoNET model yields the optimal training, validation, and testing performances compared with the other two models.

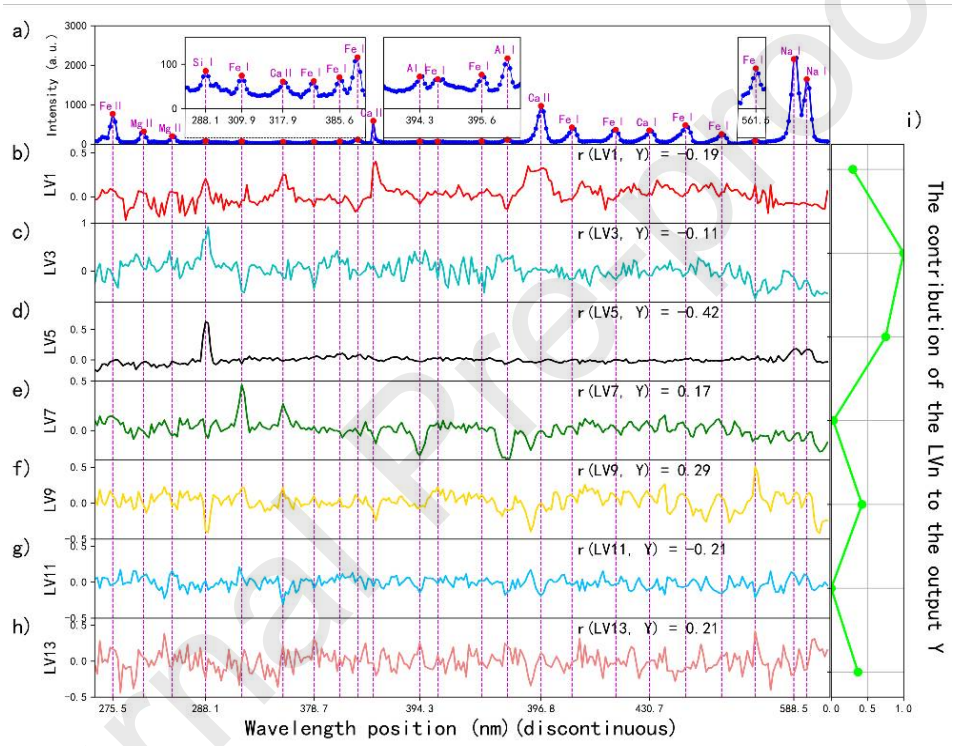


Fig. 15. Details of the regression process of the PLSaoNET model.

5. Conclusions

In summary, this study proposed a novel PLS-assisted optimization network named PLSaoNET. By employing PLS to determine the number of hidden-layer neurons and the initialization weights of the network, the network is subject to PLS statistical constraints, transforming it from a purely data-driven multivariate black-box model to a new methodology with stability and reliability comparable to those of traditional linear methods such as PLS. Moreover, these findings enhance our understanding of how industrial sensor models utilize measured variables to obtain key performance indicators such as material composition, equipment status, and product quality under complex industrial processes. In addition, to address the problem of uneven sample distribution, which severely affects the network optimization process in industrial online applications, this study proposed the use of stratified sampling to optimize the training process in order to reduce the unevenness effect.

The PLSaoNET model is a generalized method that effectively reduces the difficulty of applying ANNs for industrial online sensors. Starting from the mechanism of a neural network, this method offers a comprehensive theoretical framework for guiding the determination of neural network hyperparameters and specifying solution paths, while meeting the triple requirements of accuracy, robustness, and ease of use in industrial applications. The superiority of the proposed method was fully validated in two applications—the online sensing of iron ore concentrate slurry during a mineral flotation process and a quality assessment task of diesel fuel based on NIR data. The proposed model achieved higher accuracy than PLS while maintaining a more stable generalization performance than BPNN with random initialization. The proposed model has been successfully applied to the LIBSlurry analyzer at a mineral processing industrial site.

Remarkably, the current work has only implemented the concept of PLS-assisted optimization in the most basic neural network (BPNN with a single hidden layer). Through theoretical analysis and experimental validation, we have preliminarily demonstrated the potential of the proposed PLSaoNET framework for industrial sensing applications with high input dimensions, limited labeled samples, and measurements under harsh conditions.

The proposed PLSaoNET framework can be further developed to achieve higher sensing accuracy or generalized to more industrial sensing scenarios. Firstly, the powerful fitting capability of a neural network mainly depends on the depth and width of the network (i.e., the number of layers and neurons in each layer). With the continuous development of the Industrial Internet and sensing technology, industrial data volume is experiencing exponential growth, and the data quality is constantly improving. Hence, there is a need for the development of more complex network model structures. In the future, we will explore the method of stacking multiple PLS models to increase the depth of the network.

Secondly, the PLSaoNET method proposed in this paper is a fully connected neural network. In typical industrial sensing applications, visual-based defect detection for product classification requires the processing of image inputs, while bearing remaining useful life (RUL) prediction necessitates the handling of time-series inputs. The current fully connected PLSaoNET model encounters challenges in handling these industrial sensing applications with unstructured information, such as spatial and temporal locations in image and time-series data. However, mature neural networks such as CNNs and LSTMs have developed spatial or temporal information processing units, including convolutions and gating mechanisms. Integrating these processing units into the PLSaoNET model will be another research focus of our future work.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (62173321) and in part by the Research Program of the Liaoning Liaohe Laboratory (LL23ZZ-05-02).

References

[1] Qian F, Zhong W, Du W. Fundamental theories and key technologies for smart and optimal manufacturing in the process industry. Engineering 2017;3(2):154–60.

[2] Chen P, Yang S, McCann JA. Distributed real-time anomaly detection in networked industrial sensing systems. IEEE Trans Ind Electron 2015;62(6):3832–42.

[3] Roy U, Li Y, Zhu B. Building a rigorous foundation for performance assurance assessment techniques for “smart” manufacturing systems. In: 2014 IEEE International Conference on Big Data; 2014 Oct 27–30; Washington, DC, USA. New York City: IEEE; 2014. p. 1015–23.

[4] Ren L, Meng Z, Wang X, Zhang L, Yang L. A data-driven approach of product quality prediction for complex production systems. IEEE Trans Industr Inform 2021;17(9):6457–65.

[5] Tambare P, Meshram C, Lee CC, Ramteke RJ, Imoize AL. Performance measurement system and quality management in data-driven Industry 4.0: a review. Sensors 2022;22(1):224.

[6] Shang C, You F. Data analytics and machine learning for smart process manufacturing: recent advances and perspectives in the big data era. Engineering 2019;5(6):1010–6.

[7] Salpavaara T, Järveläinen M, Seppälä S, Yli-Hallila T, Verho J, Vilkko M, et al. Passive resonance sensor based method for monitoring particle suspensions. Sens Actuators B 2015;219:324–30.

[8] Liu C, Wang K, Wang Y, Yuan X. Learning deep multimanifold structure feature representation for quality prediction with an industrial application. IEEE Trans Industr Inform 2022;18(9):5849–58.

[9] Ji Z, Chen C, Zhu S, Ma Y, Guan X. Intelligent edge sensing and control co-design for industrial cyber–physical system. IEEE Trans Signal Inf Proc Netw 2023;9:175–89.

[10] Gao Z, Cecati C, Ding SX. A survey of fault diagnosis and fault-tolerant techniques—part II: fault diagnosis with knowledge-based and hybrid/active approaches. IEEE Trans Ind Electron 2015;62(6):3768–74.

[11] Shi J, Zhou S. Quality control and improvement for multistage systems: a survey. IIE Trans 2009;41(9):744–53.

[12] Schrangl P, Tkachenko P, Del Re L. Iterative model identification of nonlinear systems of unknown structure: systematic data-based modeling utilizing design of experiments. IEEE Contr Syst Mag 2020;40(3):26–48.

[13] Mao J, Chen D, Zhang L. Mechanical assembly quality prediction method based on state space model. Int J Adv Manuf Technol 2016;86(1–4):107–16.

[14] Zhou X, Zhang Y, Mao T, Zhou H. Monitoring and dynamic control of quality stability for injection molding process. J Mater Process Technol 2017;249:358–66.

[15] Ren L, Jia Z, Lai L, Zhou L, Zhang L, Li B. Data-driven industrial intelligence: current status and future directions. Comput Integr Manuf Syst 2022;28(7):1913–39.

[16] Qin R, Zhang Z, Hu Z, Du Z, Xiang X, Wen G, et al. On-line evaluation and monitoring technology for material surface integrity in laser shock peening—a review. J Mater Process Technol 2023;313:117851.

[17] Russell M, Wang P, Liu SP, Jawahir IS. Mixed-up experience replay for adaptive online condition monitoring. IEEE Trans Ind Electron 2024;71(2):1979–86.

[18] Long F, Jiang S, Adekunle AG, Zavala VM, Bar-Ziv E. Online characterization of mixed plastic waste using machine learning and mid-infrared spectroscopy. ACS Sustain Chem Eng 2022;10(48):16064–9.

[19] Guo S, Zhang B, Yang T, Lyu D, Gao W. Multitask convolutional neural network with information fusion for bearing fault diagnosis and localization. IEEE Trans Ind Electron 2020;67(9):8005–15.

[20] Soni M, Khan IR, Basir S, Chadha R, Alguno AC, Bhowmik T. Light-weighted deep learning model to detect fault in IoT-based industrial equipment. Comput Intell Neurosci 2022;2022(1):2455259.

[21] Yuan X, Gu Y, Wang Y. Supervised deep belief network for quality prediction in industrial processes. IEEE Trans Instrum Meas 2021;70:2503711.

[22] Yuan X, Gu Y, Wang Y, Chen Z, Sun B, Yang C. FeO content prediction for an industrial sintering process based on supervised deep belief network. IFAC-PapersOnLine 2020;53(2):11883–8.

[23] Yuan X, Rao J, Gu Y, Ye L, Wang K, Wang Y. Online adaptive modeling framework for deep belief network-based quality prediction in industrial processes. Ind Eng Chem Res 2021;60(42):15208–18.

[24] Puli VK, Huang B. Variational Bayesian approach to nonstationary and oscillatory slow feature analysis with applications in soft sensing and process monitoring. IEEE Trans Control Syst Technol 2023;31(4):1708–19.

[25] Sun Y, Qin W, Hu J, Xu H, Sun PZH. A causal model-inspired automatic feature-selection method for developing data-driven soft sensors in complex industrial processes. Engineering 2023;22:82–93.

[26] Ding J, Yang C, Chai T. Recent progress on data-based optimization for mineral processing plants. Engineering 2017;3(2):183–7.

[27] Finkeldey F, Volke J, Zarges JC, Heim HP, Wiederkehr P. Learning quality characteristics for plastic injection molding processes using a combination of simulated and measured data. J Manuf Process 2020;60:134–43.

[28] Jiang Q, Yan X, Huang B. Review and perspectives of data-driven distributed monitoring for industrial plant-wide processes. Ind Eng Chem Res 2019;58(29):12899–912.

[29] Minh D, Wang H, Li Y, Nguyen TN. Explainable artificial intelligence: a comprehensive review. Artif Intell Rev 2022;55(5):3503–68.

[30] Wang X, Hu T, Tang L. A multiobjective evolutionary nonlinear ensemble learning with evolutionary feature selection for silicon prediction in blast furnace. IEEE Trans Neural Netw Learn Syst 2022;33(5):2080–93.

[31] Selvaraju RR, Cogswell M, Das A, Vedantam R, Parikh D, Batra D. Grad-CAM: visual explanations from deep networks via gradient-based localization. Int J Comput Vis 2020;128(2):336–59.

[32] Ye M, Li L, Yoo DY, Li H, Zhou C, Shao X. Prediction of shear strength in UHPC beams using machine learning-based models and SHAP interpretation. Constr Build Mater 2023;408:133752.

[33] Qader G, Junaid M, Abbas Q, Mubarik MS. Industry 4.0 enables supply chain resilience and supply chain performance. Technol Forecast Soc Chang 2022;185:122026.

[34] Shen L, Chen Z, Wang X, He J. Soft sensor modeling for 3D transient temperature field of large-scale aluminum alloy workpieces based on multi-loss consistency optimization PINN. Sensors 2023;23(14):6371.

[35] Lei Y, Yang H. A Gaussian process ensemble modeling method based on boosting algorithm. In: Pan Q, Zhao Q, editors. Proceedings of the 32nd Chinese Control Conference; 2013 Jul 26–28; Xi'an, China. New York City: IEEE; 2013. p. 1704–7.

[36] Pan S, Yang Q. A survey on transfer learning. IEEE Trans Knowl Data Eng 2010;22(10):1345–59.

[37] Dirks M, Turner D, Poole D. Spectral sensor fusion for prediction of Li and Zr in rocks: neural network and PLS methods. Chemom Intell Lab Syst 2023;240:104915.

[38] Khajehzadeh N, Haavisto O, Koresaar L. On-stream and quantitative mineral identification of tailing slurries using LIBS technique. Miner Eng 2016;98:101–9.

[39] Lecun Y, Bottou L, Bengio Y, Haffner P. Gradient-based learning applied to document recognition. Proc IEEE 1998;86(11):2278–324.

[40] Guo L, Zhang D, Sun L, Yao S, Zhang L, Wang Z, et al. Development in the application of laser-induced breakdown spectroscopy in recent years: a review. Front Phys 2021;16:22500.

[41] Araújo AS, Castro JP, Sperança MA, Andrade DF, de Mello ML, Pereira-Filho ER. Multiway calibration strategies in laser-induced breakdown spectroscopy: a proposal. Anal Chem 2021;93(16):6291–300.

[42] Frydenvang J, Kinch KM, Husted S, Madsen MB. An optimized calibration procedure for determining elemental ratios using laser-induced breakdown spectroscopy. Anal Chem 2013;85(3):1492–500.

[43] Li X, Wang Z, Fu Y, Li Z, Liu J, Ni W. Application of a spectrum standardization method for carbon analysis in coal using laser-induced breakdown spectroscopy (LIBS). Appl Spectrosc 2014;68(9):955–62.

[44] Tian Y, Hou S, Wang L, Duan X, Xue B, Lu Y, et al. CaOH molecular emissions in underwater laser-induced breakdown spectroscopy: spatial–temporal characteristics and analytical performances. Anal Chem 2019;91(21):13970–7.

[45] Gu W, Hou Z, Song W, Ji J, Yu X, Liu J, et al. Extended total number density compensation for uranium determination by laser-induced breakdown spectroscopy. Anal Chim Acta 2024;1288:342167.

[46] Laville S, Sabsabi M, Doucet FR. Multi-elemental analysis of solidified mineral melt samples by laser-induced breakdown spectroscopy coupled with a linear multivariate calibration. Spectrochim Acta Part B 2007;62(12):1557–66.

[47] Tognoni E, Cristoforetti G, Legnaioli S, Palleschi V. Calibration-free laser-induced breakdown spectroscopy: state of the art. Spectrochim Acta Part B 2010;65(1):1–14.

[48] Mukhono PM, Angeyo KH, Dehayem-Kamadjeu A, Kaduki KA. Laser induced breakdown spectroscopy and characterization of environmental matrices utilizing multivariate chemometrics. Spectrochim Acta Part B 2013;87:81–5.

[49] Xue X, Sun H, Yang M, Liu X, Hu H, Deng Y, et al. Advances in the application of artificial intelligence-based spectral data interpretation: a perspective. Anal Chem 2023;95(37):13733–45.

[50] Allegrini F, Olivieri AC. Linear or non-linear multivariate calibration models? That is the question. Anal Chim Acta 2022;1226:340248.

[51] Lee Y, Foster RI, Kim H, Choi S. Machine learning-assisted laser-induced breakdown spectroscopy for monitoring molten salt compositions of small modular reactor fuel under varying laser focus positions. Anal Chim Acta 2023;1241:340804.

[52] Chen T, Sun L, Yu H, Zeng P, Qi L. Online Fe grade monitoring of iron ore slurry by Morse wavelet transform and lightweight convolutional neural network based on LIBS. Spectrochim Acta Part B 2023;210:106821.

[53] Kramida A, Ralchenko Y, Reader J; NIST ASD Team. NIST Atomic Spectra Database (version 5.11) [Internet]. Gaithersburg: National Institute of Standards and Technology; 2023 Dec 5 [cited 2023 Dec 25]. Available from: https://physics.nist.gov/asd.

[54] Corona F, Zhu Z, de Souza AH, Mulas M, Muru E, Sassu L, et al. Supervised distance preserving projections: applications in the quantitative analysis of diesel fuels and light cycle oils from NIR spectra. J Process Control 2015;30:10–21.

[55] Wang S, Liu S, Yuan Y, Zhang J, Wang Z, Che X. A novel CC-tSNE-SVR model for rapid determination of diesel fuel quality by near infrared spectroscopy. Infrared Phys Technol 2020;106:103276.

[56] Hintze JL, Nelson RD. Violin plots: a box plot–density trace synergism. Am Stat 1998;52(2):181–4.

Declaration of interests

☒ 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.

☐ The author is an Editorial Board Member/Editor-in-Chief/Associate Editor/Guest Editor for [Engineering] and was not involved in the editorial review or the decision to publish this article.

☐ The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: