Modified Label Propagation on Manifold With Applications to Fault Classification

YING XIE[ID]

College of Information Science and Engineering, Northeastern University, Shenyang 110819, China
School of Intelligent Engineering, Shenyang City University, Shenyang 110112, China

e-mail: xieyingemail@163.com

ABSTRACT In process monitoring, fault classification performance heavily relies on the labels of training data. However, the labeled data are inadequate and difficult to obtain because they require experienced human annotators. In this paper, a modified label propagation (MLP) method is proposed to propagate labels from labeled data to unlabeled data. The proposed label propagation algorithm has the following advantages: (1) It constructs a global and local consistency framework with the aid of a data graph, manifold learning, and data labels. This framework follows the assumption that data on the manifold will have similar structures, and nearby data will have similar labels. (2) Considering the inner relationship between the unlabeled data and historical data, a new definition for the initial label matrix is offered, which is significant for label propagation. (3) The new method propagates labels in a low-dimensional manifold space, which is different from most existing label propagation methods that propagate them in the original space. The results reveal that under the global and local consistency framework, soft labels of unlabeled data are given more effective predictions. With additional soft labels of unlabeled data, the MLP-based fault classification method is introduced. The simulation results obtained using a toy example demonstrate the label propagation performance of the MLP, and those obtained for the penicillin fermentation process verify the effectiveness of the MLP-based fault classification method.

INDEX TERMS Modified label propagation, manifold learning, fault classification, semi-supervised learning, fisher discriminant analysis.

I. INTRODUCTION

For process monitoring operations in control engineering, fault classification plays a very important role in locating the fault and helping operators take correct remedial measures [1]–[5]. However, data collected from an industrial process are usually difficult to classify because of the high-dimensional data characteristics and complex data relationships involved. Based on these facts, some classification methods have been proposed, such as Fisher discriminant analysis (FDA) [6]–[8], support vector machine (SVM) [9], [10], and the k nearest neighbor (kNN) classification [11], [12]. These methods are supervised learning methods, requiring that the classes of all training data are known; that is, all training data are labeled. However, in industrial production pro-

cesses, labeled data are usually inadequate, and the acquisition of labeled data by employing skilled experts is expensive.

Hence, in recent years, to overcome the disadvantages of supervised learning methods, semi-supervised learning (SSL) methods have drawn research interest [13]–[16]. These methods can acquire knowledge via both labeled and unlabeled data for classification and are different from supervised learning methods that are heavily reliant on labeled data. Several SSL methods have been introduced in process monitoring applications. For example, Feng *et al*. proposed semi-supervised principal component analysis for process monitoring [17]. Yan *et al*. constructed a semi-supervised mixture-discriminant monitoring scheme for an injection molding process [18]. Zhong *et al*. proposed a semi-supervised FDA model for fault classification in industrial processes [19]. SSL methods can improve fault detection and fault classification

The associate editor coordinating the review of this manuscript and approving it for publication was Moussa Boukhnifer[ID].

performance even when labeled data are insufficient because the structures and features of unlabeled data are effectively explored.

Label propagation is a type of SSL method that has attracted considerable attention recently [20]–[25]. Label propagation is a commonly used method for propagating labels of labeled data to unlabeled data using their similarities and initial states. According to whether the model can handle the outside data directly, existing label propagation methods can be divided into transductive and inductive ones. Transductive learning methods can predict the labels of inside unlabeled data but cannot predict the labels of outside unlabeled data, such as linear neighborhood propagation [26], special label propagation [27], projective label propagation [28], sparse neighborhood propagation [29], adaptive neighborhood propagation [30], and positive and negative label propagation [31]. Owing to its effectiveness and efficiency, label propagation is applied in many fields. Several related methods have been researched as well. For example, Wang *et al*. proposed a label propagation method for synthetic data, and digit and text classification tasks [26]. Zhang *et al*. proposed a novel mechanism to obtain more supervised information using propagated soft labels through special label propagation [32]. Zhang *et al*. proposed a nonnegative sparse neighborhood propagation method for semi-supervised image classification [33]. Lin *et al*. proposed dynamic graph-fusion label propagation for semi-supervised multi-modality classification [34]. Zhang *et al*. proposed a projective label propagation framework involving label embedding, which can obtain the deep labels of all new data [28]. Zoidi *et al*. proposed a positive and negative label propagation method, which extends regular label propagation to negative label propagation [31].

However, the aforementioned existing label propagation methods have certain potential shortcomings that may degrade the classification results. First, data graph construction is an independent procedure before label propagation; thus, the similarity weights resulting from precalculations on the data graph may not be jointly optimal for subsequent label propagation. Therefore, traditional label propagation methods may suffer from inaccurately predicted results. Second, in traditional label propagation methods, the initial label vector is usually simply defined as a zero vector for the unlabeled data, without considering the inner relationship between the unlabeled data and historical data. However, the relationship between unlabeled data and labeled data and their implicit characteristics are valuable and should be explored for defining the initial label matrix. Third, most traditional label propagation methods propagate labels based on the original high-dimensional space. However, real-world process data usually contain various noises; undesirable, unnecessary, and irrelevant features; and even corruptions. Therefore, it is more likely that the predicted labels are inaccurate in practical scenarios. These features may lead to inaccurate label prediction results.

To address these drawbacks, a modified label propagation (MLP) method is proposed in this paper. The major contributions of the study findings to the field at large are summarized as follows:

1) A global and local consistency framework is constructed based on a data graph, manifold learning, and label propagation. This framework follows the global consistency assumption that data on the manifold will have similar structures and the local consistency assumption that nearby data will have similar labels. By solving for minimization of the objective function, optimal parameters such as the similarity weights and projection matrix were obtained and used for further label propagation.

2) In the initialization phase of label propagation, considering the inner relationship between the unlabeled data and historical data, a new definition of the initial label matrix is proposed based on the similarity and weight of each class, which is significant in label propagation processes.

3) By minimizing the feature variations with respect to neighboring structures, a low-dimensional manifold can be obtained. Moreover, the proposed method propagates label in a low-dimensional manifold. Therefore, the label propagation result would be more accurate because a low-dimensional manifold can remove noise and unfavorable features and preserve the significant features hidden in the data.

The remainder of this paper is organized as follows. A new MLP method is proposed in Section II. A fault classification method based on the MLP is introduced in Section III. A toy example and the penicillin fermentation process (PFP) are discussed in Section IV to demonstrate the effectiveness of the proposed approaches. Finally, the conclusions are summarized in the Section V.

II. MODIFIED LABEL PROPAGATION METHOD

Label propagation is a method for propagating labels of labeled data to unlabeled data, according to the relationship between the two data classes [20]–[25]. However, traditional label propagation methods propagate labels after performing an independent data graph construction process, in which the similarity weights may not be optimal for subsequent label propagation. Moreover, most traditional label propagation methods propagate labels in the original high-dimensional space, which usually contains undesirable and unnecessary features. Considering this circumstance, an MLP method is proposed, which is described as follows.

A given dataset is mapped onto a graph, and each data in the dataset correspond to a node in that graph [27], [35]. Dataset $X = [X_L, X_U] \in \mathbf{R}^{s \times n}$, where $X_L = [x_1, x_2, \ldots, x_l] \in \mathbf{R}^{s \times l}$ is the labeled dataset, $X_U = [x_{l+1}, x_{l+2}, \ldots, x_{l+u}] \in \mathbf{R}^{s \times u}$ is the unlabeled dataset, s is the original dimensionality of each data, and $l + u = n$ is the number of samples. Assume that $C = \{1, 2, \ldots, c\}$ is the class label set, and sample x_i in X_L has a unique label vector η_i ($i = 1, 2, \ldots, l$), where $\eta_{i,j} = 1$, if x_i is labeled as $j \in \{1, 2, \ldots, c\}$; otherwise, $\eta_{i,j} = 0$.

The objective function of the MLP model can be formulated as follows

$$
MinJ\,(\boldsymbol{P}, \boldsymbol{S}, \boldsymbol{F})
$$

$$
= \sum_{i=1}^{n}\sum_{j=1}^{n}\left\|\boldsymbol{P}^{T}\boldsymbol{x}_{i} - \boldsymbol{P}^{T}\boldsymbol{x}_{j}\right\|^{2}\boldsymbol{S}_{ij} + \sum_{i=1}^{n}\sum_{j=1}^{n}\left\|\boldsymbol{S}_{ij}\right\|^{2}
$$

$$
+ \alpha\sum_{i=1}^{n}\sum_{j=1}^{n}\left\|\boldsymbol{F}_{i} - \boldsymbol{F}_{j}\right\|^{2}\boldsymbol{S}_{ij}
$$

$$
+ \beta\left(\gamma_{1}\sum_{i=1}^{n}\left\|\boldsymbol{F}_{i} - \boldsymbol{D}_{i}\right\|^{2} + \gamma_{2}\sum_{i=1}^{n}\left\|\boldsymbol{F}_{i} - \boldsymbol{Q}_{i}\right\|^{2}\right),
$$

$$
s.t.\,\boldsymbol{PP}^{T} = \boldsymbol{I} \tag{1}
$$

where $\boldsymbol{P} \in \mathbf{R}^{s \times d}$ is a projection matrix in the low-dimensional space; $\boldsymbol{S}$ is the similarity weight matrix on the graph; $\boldsymbol{F} = [\boldsymbol{f}_1, \boldsymbol{f}_2, \dots, \boldsymbol{f}_n] \in \mathbf{R}^{n \times c}$ is the soft label matrix; $\boldsymbol{D} \in \mathbf{R}^{n \times c}$ and $\boldsymbol{Q} \in \mathbf{R}^{n \times c}$ are the initial label matrices, which will be explained later in this section; α, β, γ_1, and γ_2 are the regulation parameters; $\boldsymbol{I}$ is the identity matrix. $\sum_{i=1}^{n}\sum_{j=1}^{n}\left\|\boldsymbol{P}^{T}\boldsymbol{x}_{i} - \boldsymbol{P}^{T}\boldsymbol{x}_{j}\right\|^{2}\boldsymbol{S}_{ij}$ is the manifold feature smoothness term. It expresses the total variation in the manifold features with respect to the neighboring structures. It follows the global consistency assumption that data on the manifold will have similar structures. $\sum_{i=1}^{n}\sum_{j=1}^{n}\left\|\boldsymbol{F}_{i} - \boldsymbol{F}_{j}\right\|^{2}\boldsymbol{S}_{ij}$ is the label smoothness term. It expresses the total variation in the labels with respect to the neighboring structures. This term follows the local consistency assumption that nearby data will have similar labels. $\sum_{i=1}^{n}\left\|\boldsymbol{F}_{i} - \boldsymbol{D}_{i}\right\|^{2} + \sum_{i=1}^{n}\left\|\boldsymbol{F}_{i} - \boldsymbol{Q}_{i}\right\|^{2}$ is the fitted term; it expresses how well the predicted soft labels fit the initial labels. It is worth noting that $\sum_{i=1}^{n}\sum_{j=1}^{n}\left\|\boldsymbol{P}^{T}\boldsymbol{x}_{i} - \boldsymbol{P}^{T}\boldsymbol{x}_{j}\right\|^{2}\boldsymbol{S}_{ij}$, $\sum_{i=1}^{n}\sum_{j=1}^{n}\left\|\boldsymbol{S}_{ij}\right\|^{2}$, and $\sum_{i=1}^{n}\sum_{j=1}^{n}\left\|\boldsymbol{F}_{i} - \boldsymbol{F}_{j}\right\|^{2}\boldsymbol{S}_{ij}$ share the same similarity weight matrix. Therefore, this method can construct a global and local consistency framework to explicitly integrate the data graph, manifold learning, and label propagation.

Parameters $\boldsymbol{P}$, $\boldsymbol{S}$, and $\boldsymbol{F}$ in (1) are unknown. Because these parameters are coupled, there is no direct method to solve for them. To this end, we adopted an optimization strategy that updates one of the parameters while fixing the others and vice versa. The procedure for optimizing the objective function is described below in detail.

The initial value of $\boldsymbol{S}$ is $\boldsymbol{S}_{ij} = e^{\|x_i - x_j\|^2/2\delta^2}$, if $i \neq j$; otherwise, $\boldsymbol{S}_{ij} = 0$. δ is the width parameter of the Gaussian function. The initial value of $\boldsymbol{F}$ is $f_{i,j} = 1$, if $\boldsymbol{x}_i$ is labeled as $j \in \{1, 2, \dots, c\}$; otherwise, $f_{i,j} = 0$ [21], [22]. Then, projection matrix $\boldsymbol{P}$ will be calculated in the low-dimensional manifold [36]–[38], while the other parameters are fixed. The objective function about $\boldsymbol{P}$ is

$$
Min\,J\,(\boldsymbol{P}) = \sum_{i=1}^{n}\sum_{j=1}^{n}\left\|\boldsymbol{P}^{T}\boldsymbol{x}_{i} - \boldsymbol{P}^{T}\boldsymbol{x}_{j}\right\|^{2}\boldsymbol{S}_{ij}, \; s.t.\,\boldsymbol{PP}^{T} = \boldsymbol{I} \tag{2}
$$

which is equal to

$$
Min\,J\,(\boldsymbol{P}) = tr\left(\left(\boldsymbol{P}^{T}\boldsymbol{X}\right)\boldsymbol{L}\left(\boldsymbol{P}^{T}\boldsymbol{X}\right)^{T}\right), \; s.t.\,\boldsymbol{PP}^{T} = \boldsymbol{I} \tag{3}
$$

$\boldsymbol{P}_{t+1}$ at the $(t+1)$th iteration is obtained by choosing the smallest d eigenvectors, which correspond to the d smallest eigenvalues of $\boldsymbol{XL}_t\boldsymbol{X}^T$ at the t th iteration [39]. Here, $\boldsymbol{L}_t = (\boldsymbol{I} - \boldsymbol{S}_t)^T(\boldsymbol{I} - \boldsymbol{S}_t)$, and d denotes the dimensionality of the low-dimensional manifold space.

After the low-dimensional manifold projection matrix $\boldsymbol{P}$ is computed and the other parameters are fixed, the similarity weight matrix $\boldsymbol{S}$ will be updated. The objective function about $\boldsymbol{S}$ is expressed as follows:

$$
Min\,J\,(\boldsymbol{S}) = \sum_{i=1}^{n}\sum_{j=1}^{n}\left\|\boldsymbol{P}^{T}\boldsymbol{x}_{i} - \boldsymbol{P}^{T}\boldsymbol{x}_{j}\right\|^{2}\boldsymbol{S}_{ij} + \sum_{i=1}^{n}\sum_{j=1}^{n}\left\|\boldsymbol{S}_{ij}\right\|^{2}
$$

$$
+ \alpha\sum_{i=1}^{n}\sum_{j=1}^{n}\left\|\boldsymbol{F}_{i} - \boldsymbol{F}_{j}\right\|^{2}\boldsymbol{S}_{ij} \tag{4}
$$

$\boldsymbol{S}$ can be calculated from the derivative of $J\,(\boldsymbol{S})$ with respect to $\boldsymbol{S}$.

$$
\partial J\,(\boldsymbol{S})/\partial \boldsymbol{S} = \left(\boldsymbol{P}^{T}\boldsymbol{X}\right)^{T}\left(\boldsymbol{P}^{T}\boldsymbol{X}\right)\boldsymbol{S} + \alpha\boldsymbol{FF}^{T}\boldsymbol{S} + \boldsymbol{S}
$$

$$
- \left(\left(\boldsymbol{P}^{T}\boldsymbol{X}\right)^{T}\left(\boldsymbol{P}^{T}\boldsymbol{X}\right) + \alpha\boldsymbol{FF}^{T}\right) \tag{5}
$$

By setting $\partial J\,(\boldsymbol{S})/\partial \boldsymbol{S}$ to zero, $\boldsymbol{S}_{t+1}$ at the $(t+1)$th iteration can be updated as

$$
\boldsymbol{S}_{t+1} = \left(\boldsymbol{X}^{T}\boldsymbol{P}_{t+1}\boldsymbol{P}_{t+1}^{T}\boldsymbol{X} + \alpha\boldsymbol{F}_{t}\boldsymbol{F}_{t}^{T} + \boldsymbol{I}\right)^{-1}
$$

$$
\times \left(\boldsymbol{X}^{T}\boldsymbol{P}_{t+1}\boldsymbol{P}_{t+1}^{T}\boldsymbol{X} + \alpha\boldsymbol{F}_{t}\boldsymbol{F}_{t}^{T}\right) \tag{6}
$$

In this method, the similarity weights can be obtained in a global and local consistency framework, which is optimal for the next iteration.

Next, the predicted soft labels matrix, $\boldsymbol{F}$, can be updated by the following formulation

$$
MinJ\,(\boldsymbol{F})
$$

$$
= \alpha\sum_{i=1}^{n}\sum_{j=1}^{n}\left\|\boldsymbol{F}_{i} - \boldsymbol{F}_{j}\right\|^{2}\boldsymbol{S}_{ij} + \beta\left(\gamma_{1}\sum_{i=1}^{n}\left\|\boldsymbol{F}_{i} - \boldsymbol{D}_{i}\right\|^{2}\right.
$$

$$
\left. + \gamma_{2}\sum_{i=1}^{n}\left\|\boldsymbol{F}_{i} - \boldsymbol{Q}_{i}\right\|^{2}\right) \tag{7}
$$

The first term of (7) is a label smoothness term. It indicates that similar samples have similar labels. The second term is a fitted term, which measures the difference between the predicted soft labels and initial labels.

Because initial labels are significant to the label propagation process, a detailed definition of initial labels is given in this section. Let $\boldsymbol{D} = [\boldsymbol{d}_1, \boldsymbol{d}_2, \dots, \boldsymbol{d}_{l+u}] \in \mathbf{R}^{(l+u) \times c}$ denote the initial labels of all data based on similarity. For labeled data, $d_{i,j} = 1$, if $\boldsymbol{x}_i$ is labeled as $j \in \{1, 2, \dots, c\}$; otherwise, $d_{i,j} = 0$. For unlabeled data, the initial values of $d_{i,j}$ are
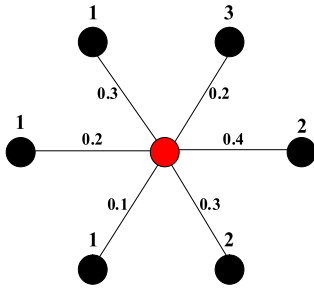
FIGURE 1. Illustration for initialization of $q_{i,j}$.

calculated using $e^{\|x_i - \mu_j\|^2 / 2\sigma^2} / \sum_j e^{\|x_i - \mu_j\|^2 / 2\sigma^2}$, where μ_j is the mean of the j-th class of labeled data, and σ is a known parameter. $Q = [q_1, q_2, \ldots, q_{l+u}] \in \mathbf{R}^{(l+u) \times c}$ denotes the initial labels of all data based on the weight of each class. For labeled data, $q_{i,j} = 1$, if x_i is labeled as $j \in \{1, 2, \ldots, c\}$; otherwise, $q_{i,j} = 0$. For unlabeled data, suppose that $N_i(1)$, $N_i(2), \ldots N_i(k)$ are the k nearest neighbors of x_i and their labels are $lN_i(1), lN_i(2), \ldots lN_i(k)$. Then, the weight of x_i from each class is calculated as follows

$$q_{i,j} = \sum_{x_m \in N_i(K), lN_i(K) \in j} e^{\|x_i - x_m\|^2 / 2\sigma^2}$$
$$/ \sum_j \sum_{x_m \in N_i(K), lN_i(K) \in j} e^{\|x_i - x_m\|^2 / 2\sigma^2},$$
$$m = 1, 2, \ldots, l, \ K \in \{1, 2, \ldots, k\} \quad (8)$$

Here, $[q_{i,1}, q_{i,2}, \cdots, q_{i,c}]$ is the initial label vector of x_i. Fig. 1 illustrates the calculation procedure of $q_{i,j}$ for the unlabeled data. The red point at the center of Fig. 1 indicates unlabeled data. The black points surrounding the red point indicate labeled data. The digits on top of the black points are the labels, and the digits on the edges between the red point and black points indicate weights. In Fig. 1, the red point has six neighbors from three different classes. According to (8), the weights of the first, second, and third classes of data can be $0.3 + 0.2 + 0.1 = 0.6, 0.3 + 0.4 = 0.7$, and 0.2, respectively. The total weight is $0.6 + 0.7 + 0.2 = 1.5$. Finally, the initial label vector of the red point should be $\left[\frac{0.6}{1.5}, \frac{0.7}{1.5}, \frac{0.2}{1.5}\right]$.

It follows from (7) that

$$MinJ(F) = \alpha tr\left(F^T L F\right) + \beta \left[\gamma_1 tr\left((F - D)^T (F - D)\right) \right.$$
$$\left. + \gamma_2 tr\left((F - Q)^T (F - Q)\right)\right] \quad (9)$$

By setting $\partial J(F) / \partial F$ to zero, the updated F_{t+1} at the $(t + 1)$th iteration is

$$F_{t+1} = (\alpha L_{t+1} + \beta (\gamma_1 + \gamma_2) I)^{-1} (\beta (\gamma_1 D + \gamma_2 Q)) \quad (10)$$

Parameters $0 < \alpha < 1$ and $0 < \beta < 1$ regulate the relative significance based on the label smoothness and fitted terms in (1), respectively, and α and β are restricted such that $\alpha + \beta = 1$. Moreover, parameters $0 < \gamma_1 < 1$ and

$0 < \gamma_2 < 1$ regulate the relative significance based on initial labels D and Q, respectively, and γ_1 and γ_2 are restricted such that $\gamma_1 + \gamma_2 = 1$. Parameter adjustment is mainly based on data structures and historical experience or knowledge. For example, for the centralized data in each class, γ_1 can be increased accordingly. When there is a sudden change in industrial production processes or the process data are highly clustered in a few operating points, γ_2 can be increased accordingly.

Parameters P, S, and F are updated based on the above iteration method until the following convergence condition is satisfied

$$\|F_{t+1} - F_t\|^2 \leq \varepsilon \quad (11)$$

where ε is a threshold. The convergence condition indicates there is no significant difference between the predicted soft labels for two sequences. Next, optimal values of P, S, and F can be obtained. Eventually, the soft label of sample x_i is determined according to $\arg \max_j \{f_{ij}\}$, i.e., the column index of the largest element in f_i.

III. FAULT CLASSIFICATION BASED ON MODIFIED LABEL PROPAGATION

Label propagation is a transductive learning procedure for predicting the labels of unlabeled data. In other words, label propagation can predict the soft labels of unlabeled data only in a given dataset, indicating insufficient generalization. In the following paragraphs, a fault classification approach is proposed based on the MLP and FDA models [6], [40].

The proposed MLP method is used to obtain the soft labels of unlabeled data. Now, the within-class scatter matrix S_w^U and between-class scatter matrix S_b^U of the unlabeled data, and the within-class scatter matrix S_w^L and between-class scatter matrix S_b^L of the labeled data can be obtained. Using them in FDA, a semi-supervised counterpart, called SFDA, is derived. S_b is the regularized between-class scatter matrix and S_w is the regularized within-class scatter matrix, and they are defined as follows:

$$S_b = \theta S_b^L + (1 - \theta) S_b^U$$
$$S_w = \theta S_w^L + (1 - \theta) S_w^U \quad (12)$$

where θ is a trade-off parameter adjusting the proportion based on the labeled data and unlabeled data. Then, the projection matrix is computed as $W = \arg \max_W \frac{tr(W^T S_b W)}{tr(W^T (S_w + \omega I) W)}$, where ω is the regularization parameter. The original data can be projected onto a lower-dimensional space via W. Then, a classifier is designed to classify data in this low-dimensional space. In this study, we used the probability density function as the classifier.

First, the mean μ_j and covariance ξ_j of the j-th class data are calculated in the low-dimensional space. The j-th class mean and covariance are calculated as follows

$$\mu_j = \theta \mu_j^l + (1 - \theta) \mu_j^u$$
$$\xi_j = \theta \xi_j^l + (1 - \theta) \xi_j^u \quad (13)$$

where $\boldsymbol{\mu}_j^l$ and $\boldsymbol{\mu}_j^u$ are the means of the j-th class of labeled data and unlabeled data, respectively; $\boldsymbol{\xi}_j^l$ and $\boldsymbol{\xi}_j^u$ are the covariances of the j-th class of labeled data and unlabeled data, respectively.

The linear discriminant analysis method assumes that each class of data obeys the Gaussian distribution. The conditional probability density function [34], [35] of the lower-dimensional projectio z can be represented by the mean $\boldsymbol{\mu}_j$ and covariance $\boldsymbol{\xi}_j$.

$$P(z\,|z \in j) = (2\pi)^{-\frac{r}{2}} \left(\det \boldsymbol{\xi}_j\right)^{-\frac{1}{2}}$$
$$\times \exp\left\{-\frac{1}{2}\left(z - \boldsymbol{\mu}_j\right)^{\mathrm{T}} \boldsymbol{\xi}_j^{-1}\left(z - \boldsymbol{\mu}_j\right)\right\} \quad (14)$$

where r represents the number of low-dimensional spaces. Suppose that the prior probability of each class is equal; then, according to Bayes' formula [43], the posterior probability $P(z \in j\,|z)$ can be calculated. For any new data, $\boldsymbol{x}_{new}$, the projection on a low-dimensional space is z. Then, z is brought into the conditional probability density functions of all classes of data. Thus, the conditional and posterior probabilities can be calculated. After that, the category of new data can be identified through the following classification criterion

$$C(\boldsymbol{x}_{new}) = \arg \max_{j}\{P(z \in j\,|z)\}, \quad j \in \{1, 2, \ldots, c\} \quad (15)$$

where $C(\boldsymbol{x}_{new})$ is the class of $\boldsymbol{x}_{new}$.

The process modeling and monitoring procedures based on MLP are summarized in the following subsections.

A. MODELING
(1) Obtain the labeled dataset $[\boldsymbol{X}_L, \boldsymbol{\eta}] = \left[\left(\boldsymbol{x}_1, \eta_1\right), \cdots, \left(\boldsymbol{x}_l, \eta_l\right)\right]$ and unlabeled dataset $\boldsymbol{X}_U = [\boldsymbol{x}_{l+1}, \boldsymbol{x}_{l+2}, \ldots, \boldsymbol{x}_{l+u}]$.

(2) Construct a neighborhood graph and initialize $\boldsymbol{S}, \boldsymbol{F}, \boldsymbol{D}$, and $\boldsymbol{Q}$.

(3) Update $\boldsymbol{P}$, $\boldsymbol{S}$, and $\boldsymbol{F}$ until $\boldsymbol{F}$ is convergent, and then obtain the soft label matrix $\boldsymbol{F}$.

(4) Given a set of training data, including labeled data and soft labeled data, use the SFDA method to obtain the projection matrix $\boldsymbol{W}$.

(5) Calculate $\boldsymbol{\mu}_j$ and $\boldsymbol{\xi}_j$ of the j-th class of data.

B. MONITORING
(1) Obtain new data, $\boldsymbol{x}_{new}$.

(2) Calculate the low-dimensional projection z using $z = \boldsymbol{W}^{\mathrm{T}} \boldsymbol{x}_{new}$.

(3) Calculate the conditional probability and posterior probability.

(4) The class of $\boldsymbol{x}_{new}$ can be identified.

The flowchart of the MLP-based fault classification approach is shown in Fig. 2.

IV. CASE STUDY
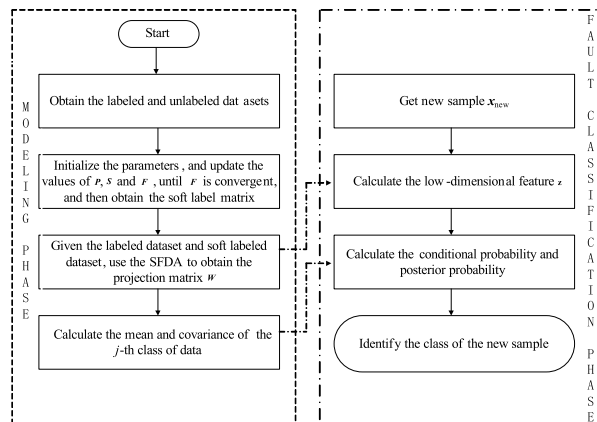In this section, a toy example is used to explain the label propagation performance of the MLP, and the PFP is adopted
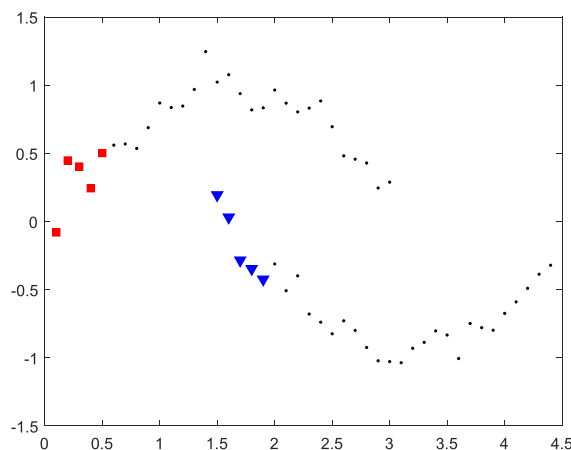

FIGURE 2. MLP-based fault classification procedure.

FIGURE 3. Testing dataset 2 with 10 labeled data.

to describe the performance of the MLP-based fault classification method.

A. TOY EXAMPLE
In the toy example [44], [45], we introduce the two-moon dataset. The two-moon datase contains two classes (called class 1 and class 2), each of which is located in a half-moon shape. We generate two testing datasets, namely testing dataset 1 and testing dataset 2. In testing dataset 1, each class consists of 60 samples; in testing dataset 2, each class consists of 30 samples. We consider the traditional label propagation(LP), linear neighborhood propagation (LNP), and positive and negative label propagation (PNLP) methods for comparison. Table 1 lists the accuracy levels of the LP, LNP, PNLP, and MLP methods used in this case study.

Let us consider testing dataset 2, as shown in Fig. 3. Five samples from each class are labeled and represented by red solid squares and blue solid triangles, respectively. The remaining 25 samples in each class are unlabeled data represented by black points. Parameter d is set to 1 in this simulation, k is 10, both α and β are 0.5, γ_1 and γ_2 are 0.4 and 0.6, respectively, and ε is $1e^{-6}$.
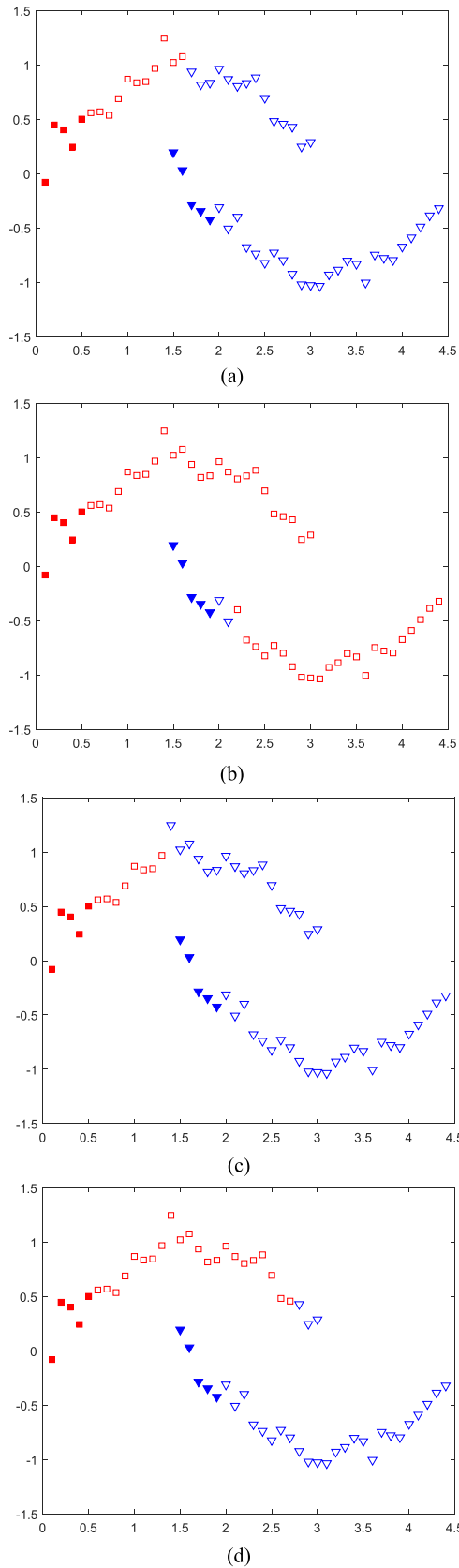
FIGURE 4. Label propagation results: (a) LP; (b) LNP; (c) PNLP; (d) MLP.
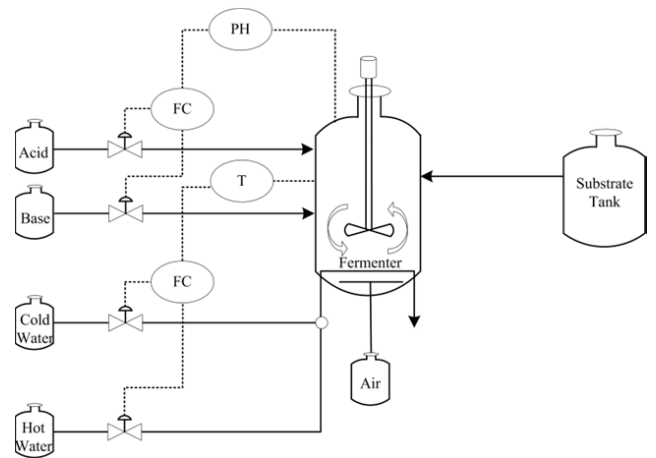

FIGURE 5. Penicillin fermentation process flow diagram.

The label propagation results obtained using LP, LNP, PNLP, and MLP are shown in Fig. 4. In this case study, the accuracy levels of the LP, LNP, and PNLP methods are 72%, 54%, and 66%, respectively. In contrast, the MLP method can yield the desired LP result, and its accuracy is 94%. From these results, we inferred that the MLP method can effectively leverage labeled and unlabeled data in the LP procedure, and the optimal global and local consistency framework in the MLP is beneficial for LP.

B. PENICILLIN FERMENTATION PROCESS
The PFP is a complex biochemical process [46]–[49]. The process flow diagram of the PFP is shown in Fig. 5. It consists of two major operational phases: bacterial growth phase and penicillin secretory phase. Because data generated under different initial conditions and operation modes have different categories, the PFP is a good candidate for evaluating the performance of the MLP-based fault classification method.

Data used for this evaluation were generated using Pensim V2.0. Training data and testing data from different classes were obtained by setting different initial conditions, set points, temperature controllers, and controller types for monitoring the pH. To achieve the best fault classification performance, 14 measurement variables were selected for monitoring, which are listed in Table 2. The PFP was run under four different modes—namely the normal mode, Fault 1 mode, Fault 2 mode, and Fault 3 mode—for generating different types of data (Table 3). The normal mode was run when default initial conditions, set points, and temperature controller settings were used. A PID controller was used to regulate the pH. Fault 1 was caused by increasing the aeration rate with a ramp fault. Fault 2 was caused by increasing the agitator power with a step fault. Fault 3 was caused by increasing the substrate feed rate with a ramp fault. The normal operation mode lasted for approximately 220 h. The

TABLE 1. Accuracy levels of LP, LNP, PNLP, and MLP (%).

Method	Testing dataset 1		Testing dataset 2	
	6 labeled samples	10 labeled samples	6 labeled samples	10 labeled samples
LP	64.0	65.5	70.4	72.0
LNP	53.5	53.6	53.7	54.0
PNLP	50.9	53.6	61.1	66.0
MLP	71.1	86.4	87.0	94.0

TABLE 2. Selected monitored variables for the PFP.

No.	Variables	No.	Variables
1	Substrate concentration	8	Substrate feed rate
2	Biomass concentration	9	Substrate feed temperature
3	Culture volume	10	Generated heat
4	CO_2	11	Base flow rate
5	Temperature	12	Acid flow rate
6	Aeration rate	13	pH
7	Agitator power	14	Penicillin concentration

TABLE 3. Description of selected four process classes for the PFP.

Class	Fault variable	Fault type
Normal	null	null
Fault 1	Aeration rate	Ramp
Fault 2	Agitator power	Step
Fault 3	Substrate feed rate	Ramp

other fault operation modes lasted for approximately 110 h each, and the sampling interval was 1 h.

In the modeling phase, the training dataset contained 300 samples (120 normal samples, and 60 Fault-1, Fault-2, and Fault-3 samples) from four different classes. We generated two testing datasets, namely testing dataset 1 and testing dataset 2. In testing dataset 1, the magnitudes of Fault 1, Fault 2, and Fault 3 are 0.7, 1.5%, and 0.7, respectively; in testing dataset 2, the magnitudes of Fault 1, Fault 2, and Fault 3 are 0.9, 2%, and 0.9, respectively.

When there are 25 labeled samples in training dataset (10 labeled samples concluded in normal class, 5 labeled samples concluded in Fault 1, Fault 2, and Fault 3 class, respectively), let us consider testing dataset 2. The testing dataset 2 consists of 100 normal samples, and 50 Fault-1, Fault-2, and Fault-3 samples, which appear in sequence. For comparison, FDA, LP-SFDA, LNP-SFDA, and PNLP-SFDA approaches were used for fault classification. Parameter r in the FDA, LP-SFDA, LNP-SFDA, PNLP-SFDA, and MLP-SFDA models is set to 4. Parameter k is set as 5; d is 3; α and β are set to

0.5; γ_1 and γ_2 are equal to 0.9 and 0.1, respectively; ε is $1e^{-6}$; and θ is 0.7 in the MLP-SFDA method.

After the FDA, LP-SFDA, LNP-SFDA, PNLP-SFDA, and MLP-SFDA models are established, the corresponding projection matrix W in the five models can be obtained. This helps calculate the corresponding low-dimensional projection z. Fig. 6 shows the first, second, and third directions of the projection results of the test data with the FDA, LP-SFDA, LNP-SFDA, PNLP-SFDA, and MLP-SFDA methods. The projections of the test data in low-dimensional subspace are separated using the five models. However, the projections obtained with FDA, as shown in Fig. 6(a), are closer to each other than those obtained with LP-SFDA, LNP-SFDA, PNLP-SFDA, and MLP-SFDA, as illustrated in Figs. 6(b), 6(c), 6(d), and 6(e), respectively. Fig. 6(a) demonstrates that when only a few labeled data are used, the FDA-based discriminant results are poor. This is because the FDA method relies on the information of labeled data. As observed in Figs. 6(b), 6(c), and 6(d), the projections obtained with LP-SFDA, LNP-SFDA, and PNLP-SFDA are farther away from one another than in Fig. 6(a). This reveals that the information from unlabeled data has been effectively used in the discrimination. In contrast, the projections obtained with MLP-SFDA are distinctly separate from each other. This indicates that using the soft labels of unlabeled data and because of their semi-supervised nature, the MLP-SFDA method can obtain a better discriminant subspace for more accurate predictions.

Next, the fault classification performances of FDA, LP-SFDA, LNP-SFDA, PNLP-SFDA, and MLP-SFDA are discussed. Fig. 7 shows the posterior probability values of testing samples in the five methods corresponding to FDA, LP-SFDA, LNP-SFDA, PNLP-SFDA, and MLP-SFDA. The final classification results produced by the five methods are shown in Fig. 8. The classification accuracy rates achieved with FDA, LP-SFDA, LNP-SFDA, PNLP-SFDA, and MLP-SFDA are 57.2%, 87.6%, 81.2%, 87.2%, and 97.6%, respectively. Specifically, according to Fig. 8(a), 23 samples that originally belong to Fault 2 are misclassified as belonging to Fault 3. For Fault 1, eight samples are misclassified as belonging to Fault 3. In the normal data, 76 samples are misclassified as belonging to Fault 3. Fig. 8(a) demonstrates that when there are only a few labeled data, the classification performance of FDA is poor because it depends on the
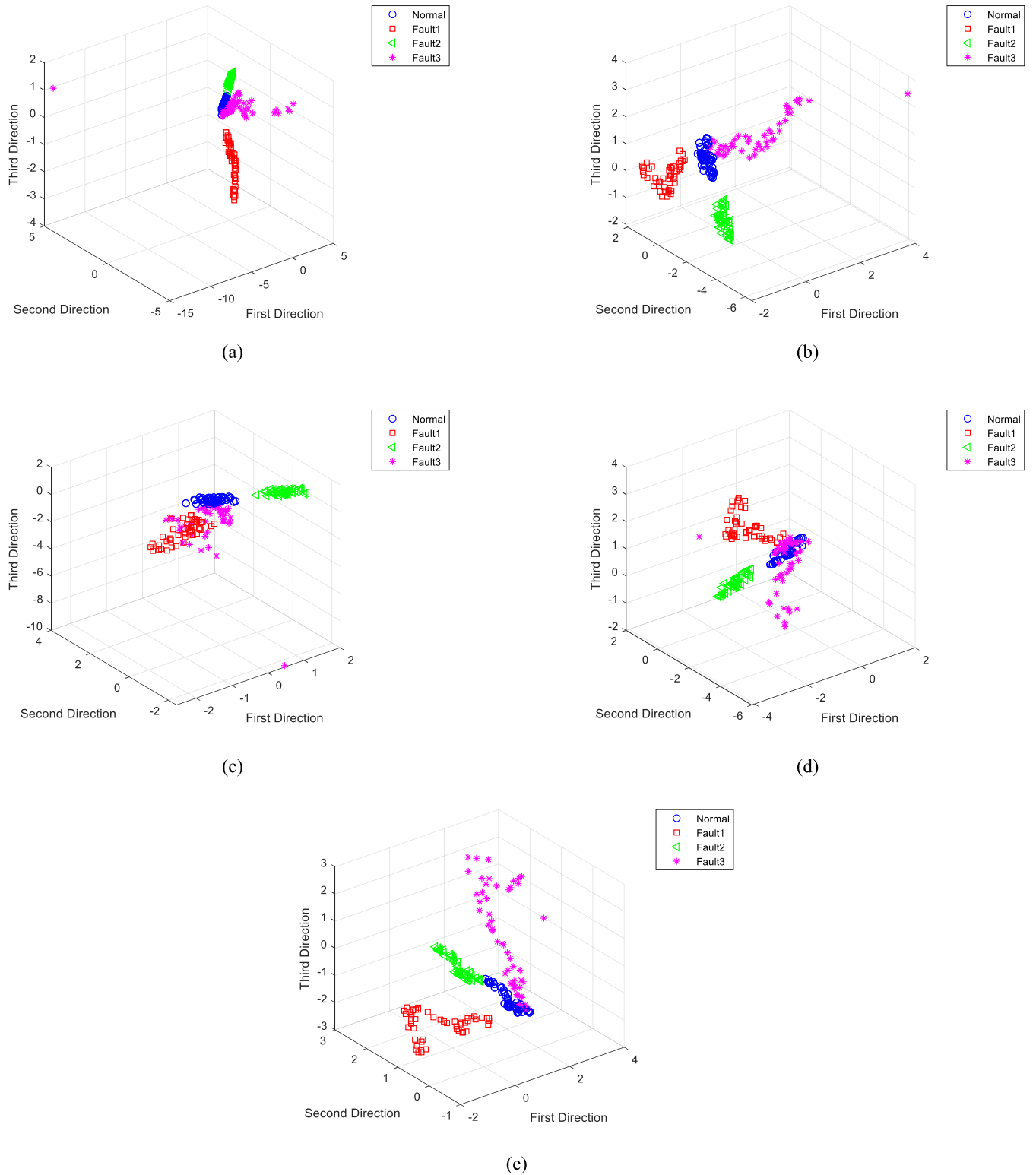
FIGURE 6. Results of dimension reduction: (a) FDA; (b) LP-SFDA; (c) LNP-SFDA; (d) PNLP-SFDA; (e) MLP-SFDA.

information of labeled data. Figs. 8(b), 8(c), and 8(d) indicate that the classification accuracy of the LP-SFDA, LNP-SFDA, and PNLP-SFDA methods is slightly higher than that of the FDA method, but there are still some misclassifications. Therefore, using a semi-supervised method improves classification performance. In contrast, Fig. 8(e) shows that the
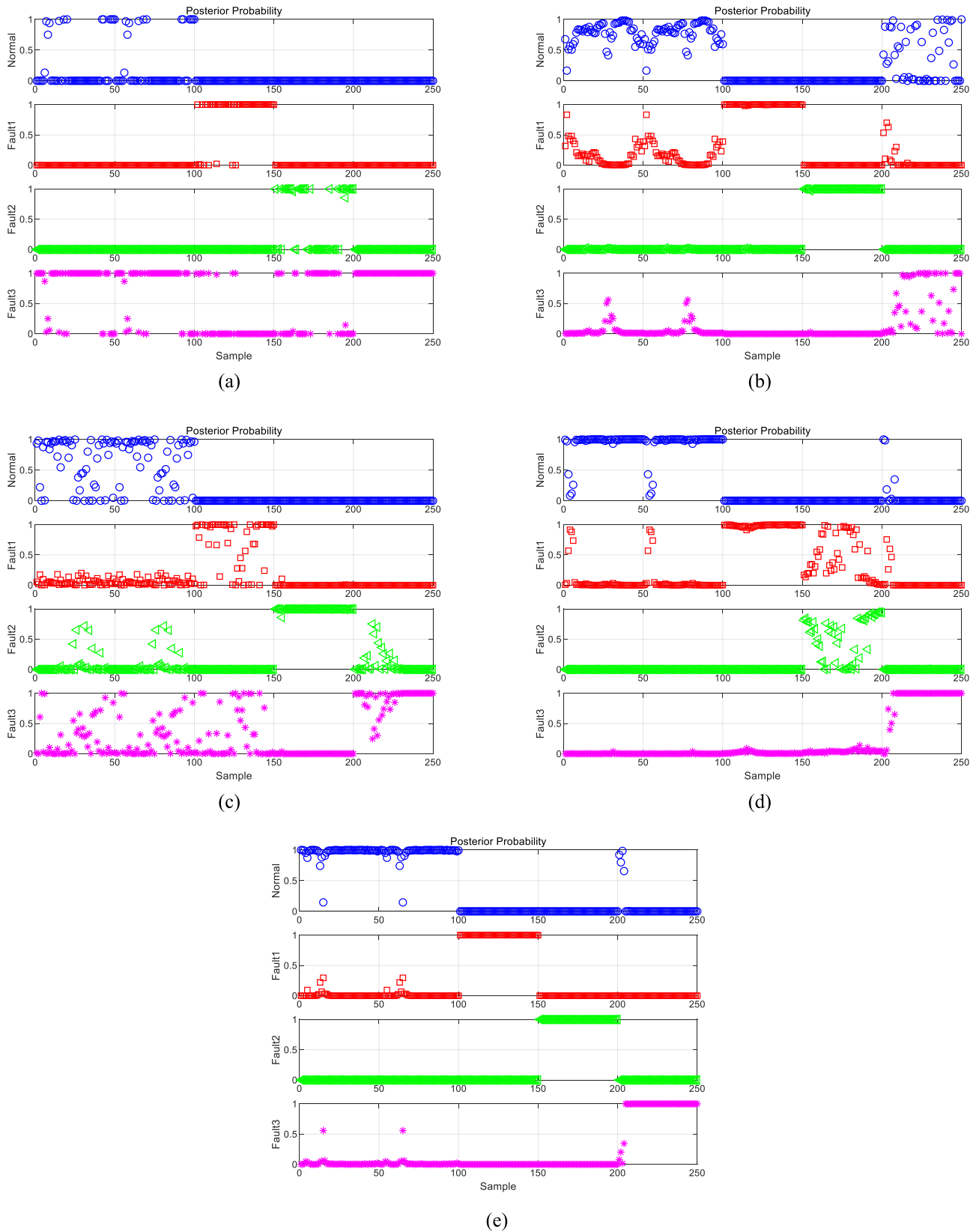
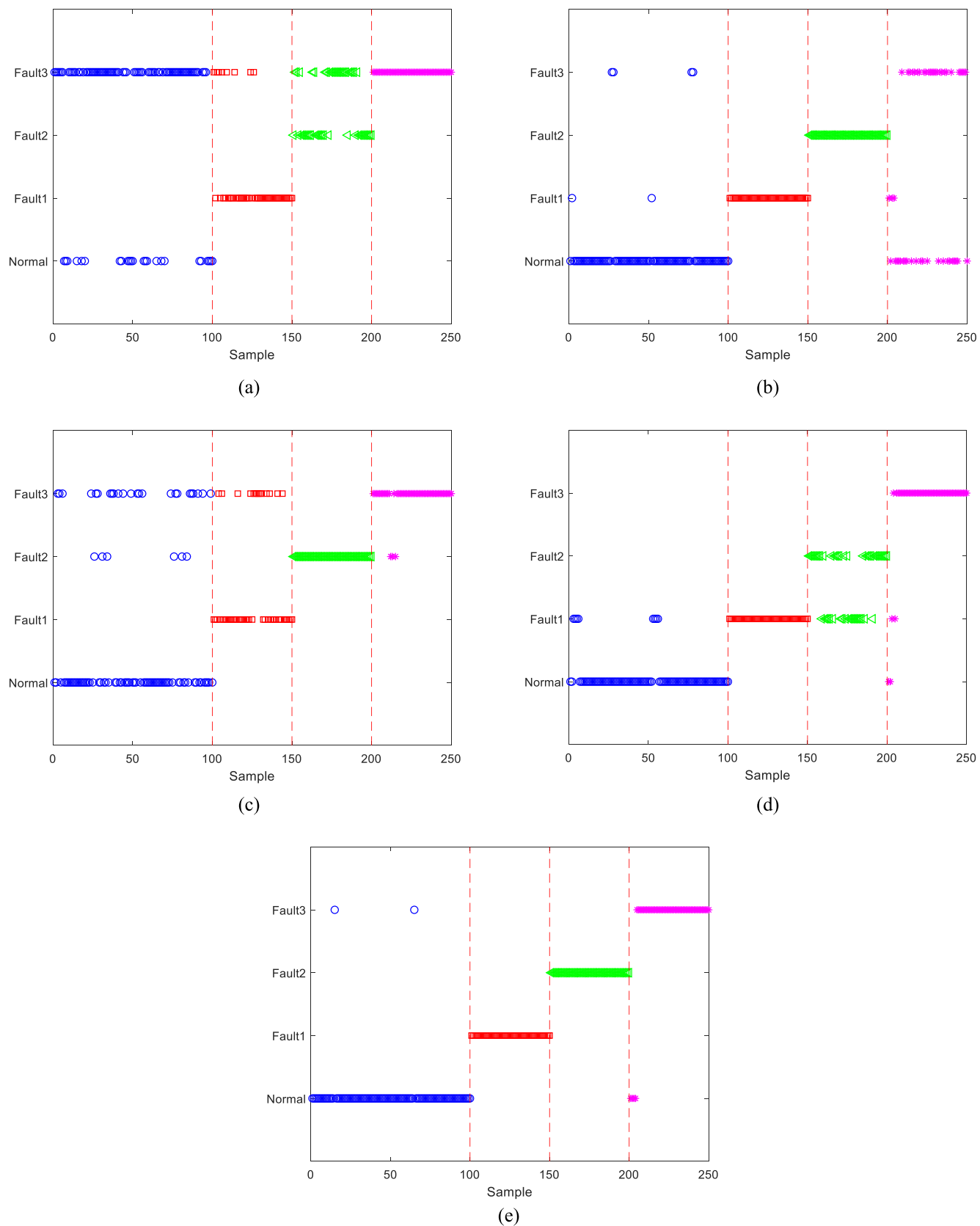**FIGURE 7.** Posterior probabilities of samples: (a) FDA; (b) LP-SFDA; (c) LNP-SFDA; (d) PNLP-SFDA; (e) MLP-SFDA.

FIGURE 8. Fault classification results: (a) FDA; (b) LP-SFDA; (c) LNP-SFDA; (d) PNLP-SFDA; (e) MLP-SFDA.

TABLE 4. Classification accuracy of FDA, LP-SFDA, LNP-SFDA, PNLP-SFDA, and MLP-SFDA (%).

Method	Testing dataset 1		Testing dataset 2	
	25 labeled samples	50 labeled samples	25 labeled samples	50 labeled samples
FDA	50.0	78.4	57.2	66.8
LP-SFDA	57.2	74.4	87.6	93.6
LNP-SFDA	66.0	79.6	81.2	92.4
PNLP-SFDA	58.0	78.8	87.2	96.8
MLP-SFDA	80.0	83.2	97.6	100.0

classification accuracy of the MLP-SFDA method is much higher than that of the FDA, LP-SFDA, LNP-SFDA, and PNLP-SFDA methods.

Table 4 lists the classification accuracy rates achieved using FDA, LP-SFDA, LNP-SFDA, PNLP-SFDA, and MLP-SFDA in this case study. The resuls indicate improvements in classification performance owing to the LP performed using MLP and the semi-supervised characteristic of unlabeled data. Therefore, based on the above results, we can conclude that the soft labels of unlabeled data can be predicted more accurately using the proposed method. With the additional soft labels of unlabeled data, the fault classification ability of MLP-SFDA is greater than that of FDA, LP-SFDA, LNP-SFDA, and PNLP-SFDA.

V. CONCLUSIONS

In this paper, an MLP method is proposed to accurately propagate labels from labeled data to unlabeled data. The toy example is utilized to evaluate the performance of MLP. Compared to the LP, LNP, and PNLP methods, the MLP method with a new global and local consistency framework, has been validated in predicting the soft labels of unlabeled data accurately. In addition, the MLP-based fault classification method is introduced with additional soft labels of unlabeled data. As a proof of concept, the PFP is utilized to verify the fault classification performance of the proposed method. The results proved that the MLP-SFDA method can achieve higher classification accuracy than the traditional FDA, LP-SFDA, LNP-SFDA, and PNLP-SFDA methods. Furthermore, the proposed approach can improve the fault classification accuracy effectively.

Although the proposed method yields encouraging results, optimization of the model parameters still needs to be investigated, and more subjects and complex industrial process data will be required to test the MLP and MLP-SFDA-based fault classification methods in future.

REFERENCES

[1] H. Zhao, J. Zheng, J. Xu, and W. Deng, "Fault diagnosis method based on principal component analysis and broad learning system," *IEEE Access*, vol. 7, pp. 99263–99272, Jul. 2019.
[2] Y. Xu, Y. Sun, J. Wan, X. Liu, and Z. Song, "Industrial big data for fault diagnosis: Taxonomy, review, and applications," *IEEE Access*, vol. 5, pp. 17368–17380, Jul. 2017.
[3] Z. Gao, C. Cecati, and S. X. Ding, "A survey of fault diagnosis and fault-tolerant techniques—Part I: Fault diagnosis with model-based and signal-based approaches," *IEEE Trans. Ind. Electron.*, vol. 62, no. 6, pp. 3757–3767, Jun. 2015.
[4] S. Yin, X. Li, H. Gao, and O. Kaynak, "Data-based techniques focused on modern industry: An overview," *IEEE Trans. Ind. Electron.*, vol. 62, no. 1, pp. 657–667, Jan. 2015.
[5] X. Z. Wang, S. Medasani, F. Marhoon, and H. Albazzaz, "Multidimensional visualization of principal component scores for process historical data analysis," *Ind. Eng. Chem. Res.*, vol. 43, no. 22, pp. 7036–7048, Oct. 2004.
[6] J. J. Hong, J. Zhang, and J. Morris, "Fault localization in batch processes through progressive principal component analysis modeling," *Ind. Eng. Chem. Res.*, vol. 50, no. 13, pp. 8153–8162, Jul. 2011.
[7] J. Yu, "Localized Fisher discriminant analysis based complex chemical process monitoring," *AIChE J.*, vol. 57, no. 7, pp. 1817–1828, Jul. 2011.
[8] L. H. Chiang, E. L. Russell, and R. D. Braatz, "Fault diagnosis in chemical processes using Fisher discriminant analysis, discriminant partial least squares, and principal component analysis," *Chemometric Intell. Lab. Syst.*, vol. 50, no. 2, pp. 243–252, Mar. 2000.
[9] S. U. Jan, Y.-D. Lee, J. Shin, and I. Koo, "Sensor fault classification based on support vector machine and statistical time-domain features," *IEEE Access*, vol. 5, pp. 8682–8690, May 2017.
[10] S.-F. Yuan and F.-L. Chu, "Support vector machines-based fault diagnosis for turbo-pump rotor," *Mech. Syst. Signal Process.*, vol. 20, no. 4, pp. 939–952, May 2006.
[11] G. Wang, J. Liu, and Y. Li, "Fault diagnosis using kNN reconstruction on MRI variables," *J. Chemometrics*, vol. 29, no. 7, pp. 399–410, Jul. 2015.
[12] S. Paudyal, M. S. A. Atique, and C. X. Yang, "Local maximum acceleration based rotating machinery fault classification using KNN," in *Proc. IEEE Int. Conf. Electro Inf. Technol. (EIT)*, May 2019, pp. 219–224.
[13] T. S. Abdelgayed, W. G. Morsi, and T. S. Sidhu, "Fault detection and classification based on co-training of semisupervised machine learning," *IEEE Trans. Ind. Electron.*, vol. 65, no. 2, pp. 1595–1605, Feb. 2018.
[14] I. Monroy, R. Benitez, G. Escudero, and M. Graells, "A semi-supervised approach to fault diagnosis for chemical processes," *Comput. Chem. Eng.*, vol. 34, no. 5, pp. 631–642, May 2010.
[15] S. Chen, F. Cerda, P. Rizzo, J. Bielak, J. H. Garrett, and J. Kovacevic, "Semi-supervised multiresolution classification using adaptive graph filtering with application to indirect bridge structural health monitoring," *IEEE Trans. Signal Process.*, vol. 62, no. 11, pp. 2879–2893, Jun. 2014.
[16] M. H. Rohban and H. R. Rabiee, "Supervised neighborhood graph construction for semi-supervised classification," *Pattern Recognit.*, vol. 45, no. 4, pp. 1363–1372, Apr. 2012.
[17] F. Jian, W. Jian, and H. Zhiyan, "Process monitoring for chemical process based on semi-supervised principal component analysis," in *Proc. 25th Chin. Control Decis. Conf.*, Guiyang, China, May 2013, pp. 4282–4286.
[18] Z. Yan, C.-C. Huang, and Y. Yao, "Semi-supervised mixture discriminant monitoring for chemical batch processes," *Chemometric Intell. Lab. Syst.*, vol. 134, pp. 10–22, May 2014.
[19] S. Zhong, Q. Wen, and Z. Ge, "Semi-supervised Fisher discriminant analysis model for fault classification in industrial processes," *Chemometric Intell. Lab. Syst.*, vol. 138, pp. 203–211, Nov. 2014.
[20] C. Gong, D. Tao, W. Liu, L. Liu, and J. Yang, "Label propagation via teaching-to-learn and learning-to-teach," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 6, pp. 1452–1465, Jun. 2017.

[21] D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. Schölkopf, "Learning with local and global consistency," in *Proc. Adv. Neural Inf. Proces. Syst.*, 2004, pp. 321–328.

[22] C. Zhang, S. Wang, D. Li, J. Yang, and H. Chen, "Prior class dissimilarity based linear neighborhood propagation," *Knowl.-Based Syst.*, vol. 83, pp. 58–65, Jul. 2015.

[23] Q. Ye, J. Yang, T. Yin, and Z. Zhang, "Can the virtual labels obtained by traditional LP approaches be well encoded in WLR?" *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 7, pp. 1591–1598, Jul. 2016.

[24] H. Zhang, Z. Zhang, M. Zhao, Q. Ye, M. Zhang, and M. Wang, "Robust triple-matrix-recovery-based auto-weighted label propagation for classification," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Jan. 22, 2020, doi: 10.1109/TNNLS.2019.2956015.

[25] Z. Zhang, M. Zhao, and T. W. S. Chow, "Graph based constrained semi-supervised learning framework via label propagation over adaptive neighborhood," *IEEE Trans. Knowl. Data Eng.*, vol. 27, no. 9, pp. 2362–2376, Sep. 2015.

[26] F. Wang and C. Zhang, "Label propagation through linear neighborhoods," *IEEE Trans. Knowl. Data Eng.*, vol. 20, no. 1, pp. 55–67, Jan. 2008.

[27] F. Nie, S. Xiang, Y. Liu, and C. Zhang, "A general graph-based semi-supervised learning with novel class discovery," *Neural Comput. Appl.*, vol. 19, no. 4, pp. 549–555, Jun. 2010.

[28] Z. Zhang, L. Wang, L. Jia, F. Li, L. Zhang, and M. Zhao, "Projective label propagation by label embedding: A deep label prediction framework for representation and classification," *Knowl.-Based Syst.*, vol. 119, pp. 94–112, Mar. 2017.

[29] Z. Zhang, L. Zhang, M. Zhao, W. Jiang, Y. Liang, and F. Li, "Semi-supervised image classification by nonnegative sparse neighborhood propagation," in *Proc. 5th ACM Int. Conf. Multimedia Retr. (ICMR)*, Shanghai, China, 2015, pp. 139–146.

[30] L. Jia, Z. Zhang, L. Wang, W. M. Jiang, and M. B. Zhao, "Adaptive neighborhood propagation by joint L2,1-norm regularized sparse coding for representation and classification," in *Proc. IEEE 16th Int. Conf. Data Mining (ICDM)*, Barcelona, Spain, Dec. 2016, pp. 201–210.

[31] O. Zoidi, A. Tefas, N. Nikolaidis, and I. Pitas, "Positive and negative label propagations," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 2, pp. 342–355, Feb. 2018.

[32] Z. Zhang, F. Li, L. Jia, J. Qin, L. Zhang, and S. Yan, "Robust adaptive embedded label propagation with weight learning for inductive classification," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 8, pp. 3388–3403, Aug. 2018.

[33] Z. Zhang, L. Jia, M. Zhao, Q. Ye, M. Zhang, and M. Wang, "Adaptive non-negative projective semi-supervised learning for inductive classification," *Neural Netw.*, vol. 108, pp. 128–145, Dec. 2018.

[34] G. Lin, K. Liao, B. Sun, Y. Chen, and F. Zhao, "Dynamic graph fusion label propagation for semi-supervised multi-modality classification," *Pattern Recognit.*, vol. 68, pp. 14–23, Aug. 2017.

[35] Y. Zhao, R. Ball, J. Mosesian, J.-F. de Palma, and B. Lehman, "Graph-based semi-supervised learning for fault detection and classification in solar photovoltaic arrays," *IEEE Trans. Power Electron.*, vol. 30, no. 5, pp. 2848–2858, May 2015.

[36] B. Yao, P. Zhen, L. Wu, and Y. Guan, "Rolling element bearing fault diagnosis using improved manifold learning," *IEEE Access*, vol. 5, pp. 6027–6035, Jun. 2017.

[37] F. Nie, D. Xu, I. W.-H. Tsang, and C. Zhang, "Flexible manifold embedding: A framework for semi-supervised and unsupervised dimension reduction," *IEEE Trans. Image Process.*, vol. 19, no. 7, pp. 1921–1932, Jul. 2010.

[38] X. Jin, F. Yuan, T. W. S. Chow, and M. Zhao, "Weighted local and global regressive mapping: A new manifold learning method for machine fault classification," *Eng. Appl. Artif. Intell.*, vol. 30, pp. 118–128, Apr. 2014.

[39] E. Kokiopoulou, J. Chen, and Y. Saad, "Trace optimization and eigenproblems in dimension reduction methods," *Numer. Linear Algebra Appl.*, vol. 18, no. 3, pp. 565–602, May 2011.

[40] X. Jin, M. Zhao, T. W. S. Chow, and M. Pecht, "Motor bearing fault diagnosis using trace ratio linear discriminant analysis," *IEEE Trans. Ind. Electron.*, vol. 61, no. 5, pp. 2441–2451, May 2014.

[41] H.-C. Yan, J.-H. Zhou, and C. K. Pang, "Gaussian mixture model using semisupervised learning for probabilistic fault diagnosis under new data categories," *IEEE Trans. Instrum. Meas.*, vol. 66, no. 4, pp. 723–733, Apr. 2017.

[42] J. Yu, "A nonlinear Kernel Gaussian mixture model based inferential monitoring approach for fault detection and diagnosis of chemical processes," *Chem. Eng. Sci.*, vol. 68, no. 1, pp. 506–519, Jan. 2012.

[43] Z. Ge, F. Gao, and Z. Song, "Mixture probabilistic PCR model for soft sensing of multimode processes," *Chemometric Intell. Lab. Syst.*, vol. 105, no. 1, pp. 91–105, Jan. 2011.

[44] F. Nie, S. Xiang, Y. Jia, and C. Zhang, "Semi-supervised orthogonal discriminant analysis via label propagation," *Pattern Recognit.*, vol. 42, no. 11, pp. 2615–2627, Nov. 2009.

[45] L. Jia, Z. Zhang, and W. Jiang, "Transductive classification by robust linear neighborhood propagation," in *Proc. Pacific Rim Conf. Multimedia*. Cham, Switzerland: Springer, 2016, pp. 296–305.

[46] J. Feng and K. Li, "MRS-kNN fault detection method for multirate sampling process based variable grouping threshold," *J. Process Control*, vol. 85, pp. 149–158, Jan. 2020.

[47] S. Zhang, C. Zhao, and F. Gao, "Incipient fault detection for multiphase batch processes with limited batches," *IEEE Trans. Control Syst. Technol.*, vol. 27, no. 1, pp. 103–117, Jan. 2019.

[48] R. Sun and Y. Zhang, "Fault diagnosis with between mode similarity analysis reconstruction for multimode processes," *Chemometric Intell. Lab. Syst.*, vol. 164, pp. 43–51, May 2017.

[49] J. Yu, "Multiway discrete hidden Markov model-based approach for dynamic batch process monitoring and fault classification," *AIChE J.*, vol. 58, no. 9, pp. 2714–2725, Sep. 2012.

YING XIE received the master's degree in control theory and control engineering from the Shenyang University of Chemical Technology, in 2011. She is currently pursuing the Ph.D. degree in control theory and control engineering with Northeastern University, Shenyang, China. Her research interests include data modeling, fault detection, and fault diagnosis for complex industry processes.

• • •