Intelligent Sensing, Fusion, and Processing in Cyber-Physical-Social Systems - Research Article

International Journal of Distributed Sensor Networks

International Journal of Distributed Sensor Networks
2021, Vol. 17(11)
© The Author(s) 2021
DOI: 10.1177/15501477211055931
journals.sagepub.com/home/dsn
SAGE

Quality-related fault diagnosis based on *k*-nearest neighbor rule for non-linear industrial processes

Zelin Ren[1,2], Yongqiang Tang[1] and Wensheng Zhang[1,2]

Abstract

The fault diagnosis approaches based on *k*-nearest neighbor rule have been widely researched for industrial processes and achieve excellent performance. However, for quality-related fault diagnosis, the approaches using *k*-nearest neighbor rule have been still not sufficiently studied. To tackle this problem, in this article, we propose a novel quality-related fault diagnosis framework, which is made up of two parts: fault detection and fault isolation. In the fault detection stage, we innovatively propose a novel non-linear quality-related fault detection method called kernel partial least squares-*k*-nearest neighbor rule, which organically incorporates *k*-nearest neighbor rule with kernel partial least squares. Specifically, we first employ kernel partial least squares to establish a non-linear regression model between quality variables and process variables. After that, the statistics and thresholds corresponding to process space and predicted quality space are appropriately designed by adopting *k*-nearest neighbor rule. In the fault isolation stage, in order to match our proposed non-linear quality-related fault detection method kernel partial least squares-*k*-nearest neighbor seamlessly, we propose a modified variable contributions by *k*-nearest neighbor (VCkNN) fault isolation method called modified variable contributions by *k*-nearest neighbor (MVCkNN), which elaborately introduces the idea of the accumulative relative contribution rate into VC *k*-nearest neighbor, such that the smearing effect caused by the normal distribution hypothesis of VC *k*-nearest neighbor can be mitigated effectively. Finally, a widely used numerical example and the Tennessee Eastman process are employed to verify the effectiveness of our proposed approach.

Keywords

Fault detection, fault diagnosis, quality-related, non-linear industrial process, *k*-nearest neighbor rule

Date received: 19 August 2021; accepted: 4 October 2021

Handling Editor: Yanjiao Chen

Introduction

With the rapid development of industry, modern industrial systems are expanding toward the direction of large scale and complexity. To ensure safety and reliability in industrial production, multivariate statistical process monitoring (MSPM) as a kind of data-driven approach has been extensively studied and successfully applied to actual industrial processes.[1–3] In MSPM, process-related fault detection is a popular research task, of which the mainstream method is principal component analysis (PCA).[4] Many researchers have improved PCA for better application in fault detection.

[1]Research Center of Precision Sensing and Control, Institute of Automation, Chinese Academy of Sciences, Beijing, China
[2]University of Chinese Academy of Sciences, Beijing, China

Corresponding author:
Wensheng Zhang, Research Center of Precision Sensing and Control, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China.
Email: zhangwenshengia@hotmail.com

For example, Xiu et al.[5] proposed a novel Laplacian regularized robust PCA method that can effectively capture the intrinsic non-linear geometric information. Other process-related fault detection methods have canonical correlation analysis (CCA),[6] non-negative matrix factorization (NMF),[7,8] and so on. Another particularly important research direction for MSPM is quality-related fault diagnosis,[9–11] in which quality-related fault detection and fault isolation are two key tasks. Quality-related fault detection belongs to a supervised learning task in machine learning area,[12] and it aims to detect whether a fault that affects product quality occurs in the industrial system. When a fault detection algorithm indicates some faults exist in the system, fault isolation attempts to locate the faulty sensors. By the diagnosis of quality-related faults, unnecessary downtime and cost brought by the quality-unrelated faults can be greatly reduced, and the risky faulty sensors can also be located as quickly as possible. Therefore, quality-related fault diagnosis has been a research hotspot recently.[13–15]

Compared with process samples in real industry, quality samples usually have a large time lag to collect and a relatively rare quantity. Hence, the direct use of quality samples cannot meet the requirement of real-time online monitoring. To tackle this situation, the commonly adopted idea is to first establish a regression model between process variables and quality variables, and then extract the quality-related features from process variables, which will replace quality variables to realize quality-related online fault detection. Currently, two mainstream frameworks based on this idea are least squares (LS)–based and partial least squares (PLS)–based approaches. Zhou et al.[16] comprehensively analyzed the defect of PLS[17] for quality-related fault detection and proposed a total projection to latent structures (TPLS) model.[18] Yin et al.[19,20] performed singular value decomposition (SVD) method on coefficient matrix of LS and PLS separately, and presented modified partial least squares (MPLS) and improved partial least squares (IPLS). Since all above are linear methods which are unsuitable for non-linear processes, the kernel trick has been widely adopted for non-linear fault detection. Its main idea is to map the original process variables into Reproducing Kernel Hilbert Space (RKHS) through some kernel function, thus making the process variables linearly separable in such kernel space.[21] By introducing the kernel trick, many linear methods can be transformed into non-linear versions.[14,22,23] However, all of the above methods design statistics without considering the local characteristics among samples.

k-Nearest neighbor (*k*-NN) rule is a classical machine learning method, which is usually adopted as a classifier.[24,25] Because it is capable of mining local characteristics between near neighbors,[26] *k*-NN rule has been modified to propose a fault detection method based on the *k*-NN rule, namely, FD-*k*-NN.[27] It provides a promising direction for the solution of the above problems to some extent. The statistics of FD-*k*-NN are designed by fully considering the Euclidean distance measurement among local neighbor samples. Due to its excellent performance, *k*-NN-based fault detection methods have been studied extensively.[28–30] Although these methods have been employed into various tasks, such as multirate sampling process[31] and multimode process,[32] quality-related fault detection methods based on *k*-NN rule have still not been well-established so far.

Fault isolation is a successor task of fault detection, which is utilized to locate fault sensor variables. Many classical fault isolation approaches have been proposed. Contribution plot and reconstruction-based contribution (RBC) are two most commonly used isolation methods, but they have relatively obvious smearing effect. To deal with this problem, many other fault isolation techniques have been addressed in previous works.[33–36] Unfortunately, these methods fail to be used in *k*-NN-based fault detection. To handle the problem, Zhou et al.[37] proposed a novel fault isolation method based on *k*-NN rule, VC*k*-NN, which makes it possible to determine the failed sensors after detecting faults using FD-*k*-NN. Compared with previous methods, VC*k*-NN suffers from less effect of fault smearing. Nevertheless, VC*k*-NN assumes that the process samples follow a multivariate normal distribution, which greatly restricts its effect of fault isolation.

In this article, we propose a novel quality-related fault diagnosis framework, which is made up of two parts: non-linear quality-related fault detection and fault isolation. For quality-related fault detection task, the motivation of proposing kernel partial least squares (KPLS) by combining KPLS with *k*-NN rule is that (1) KPLS can effectively utilize process variables to obtain predictive quality variables, which will replace the actual quality variables that cannot get in real time, and (2) *k*-NN rule can mine information between a test sample and the nearest template samples for a better detection effect compared with only using a single test sample. For fault isolation task, the motivation to improve VC*k*-NN to MVC*k*-NN is that the hypothesis is sometimes not satisfied, and in this case, VC*k*-NN still has a smearing effect, so MVC*k*-NN is presented to deal with this problem by mitigating the influence of faulty variables on faultless variables. The main contributions of this article are summarized as follows:

- We propose a new quality-related non-linear fault diagnosis framework based on *k*-NN rule, including a new quality-related non-linear fault detection method KPLS-*k*-NN and a new fault isolation method modified VC*k*-NN.
- KPLS-*k*-NN is proposed by combining KPLS with *k*-NN rule. Quality-related statistics of

KPLS-k-NN take into full consideration the local neighbor information among predicted quality samples, which greatly improve the detection rate (DR) for quality-related faulty samples.
- Modified VCk-NN is established by introducing the idea of the relative variable contributions of accumulative relative contribution rate (ARCR) into VCk-NN, which does not need the assumption that the process samples obey a multivariate normal distribution and has more precise isolation results in identifying latent fault root cause than VCk-NN.

The rest of this article is arranged as follows: first, give some relevant preliminaries. Afterward, the k-NN scheme for quality-related non-linear fault diagnosis is proposed in a detailed presentation. Then, the simulation results are provided and discussed. Finally, we conclude the article and present our future work.

Preliminaries

Let the non-linear process contain an input data matrix $\mathbf{X}$ which records N samples of m process variables and an output data matrix $\mathbf{Y}$ including N samples of l quality variables, that is

$$\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_N]^{\mathrm{T}} \in \mathbb{R}^{N \times m} \tag{1}$$

$$\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_N]^{\mathrm{T}} \in \mathbb{R}^{N \times l} \tag{2}$$

where $\mathbf{x}_i \in \mathbb{R}^m$ and $\mathbf{y}_i \in \mathbb{R}^l (i = 1, ..., N)$ represent the ith sample of $\mathbf{X}$ and $\mathbf{Y}$, respectively. All samples are supposed to obey normal distribution.

The implementation of KPLS method is divided into two steps as follows. First, kernel trick[21] is introduced into KPLS model to effectively deal with the non-linear relationship among variables. Given a kernel function ϕ, it maps the original samples $x_i (i = 1, 2, \ldots, N)$ into a high-dimension kernel space $\mathcal{F}$, which is defined as $\mathbf{x}_i \in \mathbb{R}^m \rightarrow \phi(\mathbf{x}_i) \in \mathbb{R}^f$, where f is the dimension of $\mathcal{F}$. Thus, the process matrix $\mathbf{X}$ is transformed into feature matrix $\mathbf{\Phi}$ as

$$\mathbf{\Phi} = [\phi(\mathbf{x}_1), \phi(\mathbf{x}_2), \ldots, \phi(\mathbf{x}_N)]^{\mathrm{T}} \in \mathbb{R}^{N \times f} \tag{3}$$

As a necessary step, $\phi(\mathbf{x}_i)$ needs to be processed to zero mean vector $\overline{\phi}$, that

$$\overline{\phi}(\mathbf{x}_i) = \phi(\mathbf{x}_i) - \overline{\phi} \tag{4}$$

$$\overline{\phi} = \frac{1}{N} \sum_{i=1}^{N} \phi(\mathbf{x}_i) = \frac{1}{N} \mathbf{\Phi}^{\mathrm{T}} \mathbf{1}_N \tag{5}$$

where $\mathbf{1}_N = [1, 1, \ldots, 1]^{\mathrm{T}} \in \mathbb{R}^N$. Hence, the zero mean matrix of $\mathbf{\Phi}$ can be obtained by

$$\begin{aligned} \bar{\mathbf{\Phi}} &= \left[\overline{\phi}(\mathbf{x}_1), \overline{\phi}(\mathbf{x}_2), \ldots, \overline{\phi}(\mathbf{x}_N)\right]^{\mathrm{T}} \\ &= \left(\mathbf{I}_N - \frac{1}{N} \mathbf{1}_N \mathbf{1}_N^{\mathrm{T}}\right) \mathbf{\Phi} \end{aligned} \tag{6}$$

where $\mathbf{I}_N$ is the identity matrix. The concrete form of $\mathbf{\Phi}$ is unknowable, but its kernel matrix $\mathbf{K} = \mathbf{\Phi}\mathbf{\Phi}^{\mathrm{T}} \in \mathbb{R}^{N \times N}$ can be artificial setting. In this article, the radial basis function (RBF) kernel is used to calculate the elements in $\mathbf{K}$, which is

$$\mathbf{K}_{i,j} = \exp\left(-\frac{\left\|\mathbf{x}_i - \mathbf{x}_j\right\|^2}{2\sigma^2}\right) (i,j = 1, 2, \ldots, N) \tag{7}$$

where σ is the kernel parameter that needs to be set according to experience. The zero mean of $\mathbf{K}$ is calculated as follows

$$\mathbf{K} = \bar{\mathbf{\Phi}}\bar{\mathbf{\Phi}}^{\mathrm{T}} = \left(\mathbf{I}_N - \frac{1}{N} \mathbf{1}_N \mathbf{1}_N^{\mathrm{T}}\right) \mathbf{K} \left(\mathbf{I}_N - \frac{1}{N} \mathbf{1}_N \mathbf{1}_N^{\mathrm{T}}\right) \tag{8}$$

Second, PLS model is established between $\bar{\mathbf{\Phi}}$ and $\mathbf{Y}$ in space $\mathcal{F}$. Thus, the KPLS model is constructed as

$$\begin{cases} \bar{\mathbf{\Phi}} = \mathbf{T}\mathbf{P}^{\mathrm{T}} + \bar{\mathbf{\Phi}}_r \\ \mathbf{Y} = \mathbf{U}\mathbf{Q}^{\mathrm{T}} + \mathbf{Y}_r \end{cases} \tag{9}$$

where $\mathbf{T} \in \mathbb{R}^{N \times a}$ and $\mathbf{P} \in \mathbb{R}^{f \times a}$ are the score matrix and the loading matrix of $\bar{\mathbf{\Phi}}$, respectively, $\mathbf{U} \in \mathbb{R}^{N \times a}$ and $\mathbf{Q} \in \mathbb{R}^{l \times a}$ are the score matrix and the loading matrix of $\mathbf{Y}$, respectively, $\bar{\mathbf{\Phi}}_r$ and $\mathbf{Y}_r$ are the residual matrices, and a represents the number of latent variables.

The iterative calculation algorithm of KPLS has been elaborated in Jiao et al.,[14] in which we can get the score matrix $\mathbf{U}$ and $\mathbf{R}$ that is $\mathbf{R} = \bar{\mathbf{\Phi}}^{\mathrm{T}}\mathbf{U}(\mathbf{T}^{\mathrm{T}}\bar{\mathbf{K}}\mathbf{U})^{-1}$. For the score matrix $\mathbf{T}$ and the loading matrix $\mathbf{Q}$, the following equations hold

$$\mathbf{T} = \bar{\mathbf{\Phi}}\mathbf{R} \tag{10}$$

$$\mathbf{Q} = \mathbf{Y}^{\mathrm{T}}\mathbf{T} \tag{11}$$

Obviously, the score vector $\mathbf{t}_{new}$ of $\overline{\phi}(\mathbf{x}_{new})$ is

$$\mathbf{t}_{new} = \mathbf{R}^{\mathrm{T}}\overline{\phi}(\mathbf{x}_{new}) \tag{12}$$

Methodology

FD-k-NN as a popular fault detection approach has ability in determining whether a fault has occurred in the process, but it cannot estimate whether the fault occurred will have an impact on the production quality.

Therefore, in this section, we propose k-NN-based fault diagnosis scheme. Its fault detection scheme KPLS-k-NN is designed as follows: first, FD-k-NN is employed to monitor the process space. Then, KPLS is adopted to obtain the predicted quality samples $\mathbf{Y}_p$ of training process samples $\mathbf{X}$ in predicted quality space, where quality-related fault detection will be carried out. When KPLS-k-NN indicates that the system has failed, a new fault isolation method MVCk-NN is given for better locating the faulty sensor variables.

The proposed KPLS-k-NN for fault detection

Fault detection in process space. FD-k-NN is designed through following the principle: any normal samples should be close to other normal samples to some extent, while for a faulty sample, it should deviate from normal samples. Usually, the degree of deviation is measured by k-NN distance, which is defined as the average square distance between the test sample and its k-NNs from the training normal samples. When the k-NN distance of a sample exceeds the threshold, it is considered as a faulty sample, otherwise, it is judged as a normal sample. The details of the algorithm are as follows.

Model building. Given the training samples $\mathbf{X}$, the model is built according to the following procedures:

1. Find the k-NNs for each sample $\mathbf{x}_i$ in $\mathbf{X}$ and compute all the Euclidean distance, that

$$d_{ij} = \left\|\mathbf{x}_i - \mathbf{x}_j\right\|_2, j \in k\text{−NNs}(\mathbf{x}_i) \tag{13}$$

where k−NNs$(\mathbf{x}_i)$ represents the set of k-NNs of $\mathbf{x}_i$.

2. Calculate the k-NN distance of $\mathbf{x}_i$. k-NN distance is adopted as the statistics $D^2_{\mathbf{x}}(\mathbf{x}_i)$ as

$$D^2_{\mathbf{x}}(\mathbf{x}_i) = \frac{1}{k}\sum_{j=1}^{k} d^2_{ij} \tag{14}$$

3. Determine the threshold $D^2_{\mathbf{x},\alpha}$:

$D^2_{\mathbf{x}}$ is rearranged in descending order as $D^2_{\mathbf{x}\_arrange}$, of which $(1-\alpha)$−empiricalquartile is chosen as $D^2_{\mathbf{x},\alpha}$ in Zhou et al.,[37] that is, $D^2_{\mathbf{x}\_arrange}(\lfloor N(1-\alpha)\rfloor)$.

Fault detection. For a new incoming test sample $\mathbf{x}_{new}$, the sample categories are inferred as the following steps:

1. Find $\mathbf{x}_{new}$'s k-NNs from $\mathbf{X}$.
2. Compute $\mathbf{x}_{new}$'s k-NN distance $D^2_{\mathbf{x}}(\mathbf{x}_{new})$.
3. Compare $D^2_{\mathbf{x}}(\mathbf{x}_{new})$ with the threshold $D^2_{\mathbf{x},\alpha}$:

If $D^2_{\mathbf{x}}(\mathbf{x}_{new}) > D^2_{\mathbf{x},\alpha}$, then $\mathbf{x}_{new}$ is considered as a faulty sample. Otherwise, $\mathbf{x}_{new}$ is a normal sample.

Fault detection in predicted quality space. Given the training process samples $\mathbf{X}$ and training quality samples $\mathbf{Y}$. To obtain the predicted quality samples $\mathbf{Y}_p$ of training process samples $\mathbf{X}$, KPLS[41] is adopted to obtain $\mathbf{Y}_p$ (the prediction value of $\mathbf{Y}$), since it has the ability to utilize $\mathbf{X}$ to effectively mine the essential information of $\mathbf{Y}$. We set the coefficient matrix of $\mathbf{X}$ and $\mathbf{Y}_p$ as $\mathbf{M}$, which is

$$\mathbf{M} = \mathbf{U}(\mathbf{T}^{\mathrm{T}}\bar{\mathbf{K}}\mathbf{U})^{-1}\mathbf{T}^{\mathrm{T}}\mathbf{Y} \tag{15}$$

According to the calculation of equations (10) and (11), the predictive output $\mathbf{Y}_p$ is calculated by

$$\mathbf{Y}_p = \mathbf{T}\mathbf{Q}^{\mathrm{T}} = \bar{\mathbf{\Phi}}\bar{\mathbf{\Phi}}^{\mathrm{T}}\mathbf{U}(\mathbf{T}^{\mathrm{T}}\bar{\mathbf{K}}\mathbf{U})^{-1}\mathbf{T}^{\mathrm{T}}\mathbf{Y} = \bar{\mathbf{K}}\mathbf{M} \tag{16}$$

Similar to process space, we call the space where $\mathbf{y}_p(i = 1, \ldots, N) \in \mathbf{Y}_p$ is located predicted quality space.

At this point, FD-k-NN is employed to perform the detection of quality-related faults in predicted quality space. We need to find the k-NNs for each sample $\mathbf{y}_p$ in $\mathbf{Y}_p$, and obtain the $\mathbf{y}_p$'s k-NN distance $D^2_{\mathbf{y}}(\mathbf{y}_p)$ according to equation (14). Notice that since $\mathbf{X}$ and $\mathbf{Y}_p$ generally do not obey the same distribution, for $\mathbf{x}$ and $\mathbf{y}_p$ in the same point, their corresponding k-NNs might not be the samples in the same points. Hence, the k-NNs for $\mathbf{x}$ and $\mathbf{y}_p$ should be calculated separately. In addition, our KPLS-k-NN method does not need to carry out any variable transformation in the original space, and it directly depends on the Euclidian distance of the variables in the original space as an index to quantify the discrepancy between test samples and normal samples.

Different from the threshold in Zhou et al.,[37] in this article, kernel density estimation (KDE)[42], as a non-parameter probability density estimation method of random variable, is utilized to determine the threshold for two monitoring spaces, which can be referred in detail in Parzen.[38] Thus, corresponding to $D^2_{\mathbf{x}}$ and $D^2_{\mathbf{y}}$, we will get their thresholds $D^2_{\mathbf{x},\alpha}$ and $D^2_{\mathbf{y},\alpha}$.

For a new incoming test sample $\mathbf{x}_{new}$, its predicted quality $\mathbf{y}_{p,new}$ is calculated by equations (12) and (15) as follows

$$\mathbf{y}_{p,new} = \mathbf{Q}\mathbf{t}_{new} = \mathbf{M}^{\mathrm{T}}\bar{\mathbf{\Phi}}\bar{\phi}(\mathbf{x}_{new}) = \mathbf{M}^{\mathrm{T}}\bar{\mathbf{k}}_{new} \tag{17}$$

To determine whether $\mathbf{y}_{p,new}$ is a faulty sample, we find $\mathbf{y}_{p,new}$'s k-NNs from $\mathbf{Y}_p$, and compute $\mathbf{y}_{p,new}$'s k-NN distance $D^2_{\mathbf{y}}(\mathbf{y}_{p,new})$.

Finally, detection logic is performed by comparing $D^2_{\mathbf{y}}(\mathbf{y}_{p,new})$ with the threshold $D^2_{\mathbf{y},\alpha}$: if $D^2_{\mathbf{y}}(\mathbf{y}_{p,new}) > D^2_{\mathbf{y},\alpha}$, then $\mathbf{x}_{new}$ is considered as a quality-related faulty sample. Otherwise, $\mathbf{x}_{new}$ is a normal sample or a quality-unrelated faulty sample.

The whole scheme of KPLS-k-NN. By combining fault detection in process space as well as in predicted quality space, our proposed the whole KPLS-k-NN non-linear quality-related fault detection scheme is summarized as follows:

Offline modeling.
1. Normalize the training process sample $\mathbf{X}$ and training quality samples $\mathbf{Y}$ to the zero mean and unit variance.
2. Obtain the coefficient matrix $\mathbf{M}$ of KPLS by equation (15) and get $\mathbf{Y}_p$ using equation (16).
3. Calculate $\mathbf{x}$'s k-NN distance $D^2_{\mathbf{x}}(\mathbf{x})$ and $\mathbf{y}_p$'s k-NN distance $D^2_{\mathbf{y}}(\mathbf{y}_p)$ to get the thresholds $D^2_{\mathbf{x},\alpha}$ and $D^2_{\mathbf{y},\alpha}$.

Online detection. For a new incoming test sample $\mathbf{x}_{new}$:

1. Obtain $\mathbf{y}_{p,new}$ using equation (17).
2. Compute the statistics $D^2_{\mathbf{x}}(\mathbf{x}_{new})$ and $D^2_{\mathbf{y}}(\mathbf{y}_{p,new})$.

Detection logic.
1. If $D^2_{\mathbf{x}_{new}} \leqslant D^2_{\mathbf{x},\alpha}$ and $D^2_{\mathbf{y}_{new}} \leqslant D^2_{\mathbf{y},\alpha}$, the system is fault-free.
2. If $D^2_{\mathbf{y}_{new}} > D^2_{\mathbf{y},\alpha}$, the system has quality-related faults.
3. If $D^2_{\mathbf{y}_{new}} \leqslant D^2_{\mathbf{y},\alpha}$ and $D^2_{\mathbf{x}_{new}} > D^2_{\mathbf{x},\alpha}$, the system has some quality-unrelated faults.

Notice that our KPLS-k-NN is essentially a supervised quality-related fault detection method, which is designed by combining KPLS with FD-k-NN. The above seems to be a simple combination, but FD-k-NN as an unsupervised method is successfully applied to complete a supervised fault detection task.

The proposed MVCk-NN for fault isolation

The KPLS-k-NN-based fault detection approach has been presented in the above section, which can effectively judge whether there are some faults in the process and whether the faults are related to product quality. Next, when KPLS-k-NN indicates the system exists some faults, a fault isolation method matched with KPLS-k-NN will be needed to locate faulty variables. Here, we propose a new fault isolation method MVCk-NN in detail.

In general, the sensor fault as a kind of system faults is classified as the additive fault. Hence, a process sample $\mathbf{x} \in \mathbb{R}^m$ can be expressed as

$$\mathbf{x} = \mathbf{x}^* + \mathbf{f} = \mathbf{x}^* + \sum_{i=1}^{m} \xi_i f_i \tag{18}$$

where $\mathbf{x}^* \in \mathbb{R}^m$ denotes the fault-free component of $\mathbf{x}$ and $\mathbf{f} \in \mathbb{R}^m$ denotes the fault component. If $\mathbf{f} = \mathbf{0}$, $\mathbf{x}$ is a normal sample, otherwise a faulty sample. $\xi_i \in \mathbb{R}^m (i = 1, \ldots m)$ is the fault-direction indicator vector of the ith process variable among m variables, of which the ith element is 1 and other elements are 0.

To locate which sensors cause the statistics $D^2_{\mathbf{x}}$ to alarm faults, the contributions from all variables of $\mathbf{x}$ to $D^2_{\mathbf{x}}$ need to be calculated. We decompose $D^2_{\mathbf{x}}$ as follows

$$D^2_{\mathbf{x}} = \frac{1}{k}\sum_{j=1}^{k} \|\mathbf{x} - \mathbf{x}_j\|_2^2 = \sum_{i=1}^{m} \left\{ \frac{1}{k}\sum_{j=1}^{k} \left[\xi_i^T (\mathbf{x} - \mathbf{x}_j)\right] \right\} \tag{19}$$

Then, define the contribution from ith variable of $\mathbf{x}$ to $D^2_{\mathbf{x}}$ as

$$C(\mathbf{x}, i) = \frac{1}{k}\sum_{j=1}^{k} \left[\xi_i^T (\mathbf{x} - \mathbf{x}_j)\right] \tag{20}$$

Obviously, by equations (19) and (20), $D^2_{\mathbf{x}}$ is the sum of the contributions of all variables, namely, $D^2_{\mathbf{x}} = \sum_{i=1}^{m} C(\mathbf{x}, i)$.

Next, we discuss the influence of the fault magnitude on variable contributions. Set $\mathbf{x}$ include at least one fault. Without loss of generality, we assume the ith variable be faulty, and others be normal or faulty, that

$$\mathbf{x} = \mathbf{x}^* + \xi_i f_i + \sum_{j=1, j\neq i}^{m} \xi_j f_j (f_i \neq 0) \tag{21}$$

By equations (20) and (21), we have

$$C(\mathbf{x}, i) = \frac{1}{k}\sum_{j=1}^{k} \left[\xi_i^T \left(\mathbf{x}^* - \mathbf{x}_j + \xi_i f_i + \sum_{j=1, j\neq i}^{m} \xi_j f_j \right) \right] \tag{22}$$

According to neighborhood relationship, a reasonable hypothesis is given[37] as follows.

Hypothesis 1. Any variable's fault magnitude is much larger than the Euclidean distance between the variable and its neighbors on the same dimension, that

$$\|f_i\|_2 \geqslant \left\| [\mathbf{x}^*]_i - [\mathbf{x}_j]_i \right\|_2 \tag{23}$$

where $[\cdot]_i$ represents the ith component of vector.

Due to $\xi_i^T \xi_i = 1$, $\xi_i^T \xi_j = 0, \forall j \neq i$, and Hypothesis 1, it can be obtained that

$$C(\mathbf{x}, i) \approx \frac{1}{k}\sum_{j=1}^{k} f_i^2 = f_i^2 \tag{24}$$

It indicates that the contribution of each fault variable to the statistics is approximately equal to the square of the fault magnitude, and this method hardly suffers from smearing effect when the Hypothesis 1 is satisfied.

The above are VCk-NN-based variable contributions, but there exists a drawback that Hypothesis 1 sometimes cannot be enough satisfied, which causes that the contribution of fault variables may not be significantly different from that of normal variables, so as to increase the smearing effect.

Therefore, our proposed MVCk-NN introduces the idea of ARCR[35] into VCk-NN to obtain the relative variable contributions instead of absolute ones, so that the smearing effect is further eliminated. The detailed steps are illustrated as follows:

1. For a new sample $\mathbf{x}_{new} \in \mathbb{R}^m$, when $m = 1$, it means process variable is a single source variable. In this case, once a fault occurs, the faulty variable must be this process variable. However, when $m \geqslant 2$, we need to execute the following Steps 2–5 to isolate faulty process variables.
2. Normalize each $C(\mathbf{x}_{new}, i)$ obtained by equation (20) to guarantee that all variables contribute more or less the same to $\mathbf{x}_{new}$

$$\hat{C}(\mathbf{x}_{new}, i) = \frac{C(\mathbf{x}_{new}, i)}{\frac{1}{N}\sum_{j=1}^{N} C(\mathbf{x}_j, i)} \tag{25}$$

3. To eliminate the smearing effect, $\hat{C}(\mathbf{x}_{new}, i)$ is divided by the sum of all the m normalized contributions to obtain relative variable contribution $C_r(\mathbf{x}_{new}, i)$

$$C_r(\mathbf{x}_{new}, i) = \frac{\hat{C}(\mathbf{x}_{new}, i)}{\sum_{i=1}^{m} \hat{C}(\mathbf{x}_{new}, i)} \tag{26}$$

4. The recommended experience threshold is given in Peng et al.[35]

$$\theta = \frac{\|C_r(\mathbf{x}_{new})\|_2 + \frac{1}{m}}{2} \tag{27}$$

where $C_r(\mathbf{x}_{new}) = [C_r(\mathbf{x}_{new}, 1), \ldots, C_r(\mathbf{x}_{new}, m)]^T$.

5. If $C_r(\mathbf{x}_{new}, i)$ is more than θ, the ith variable of $\mathbf{x}_{new}$ can be considered as a root cause of faults, otherwise not.

Through the above derivation, the flow chart of the proposed k-NN-based quality-related non-linear fault diagnosis scheme is summarized in Figure 1.

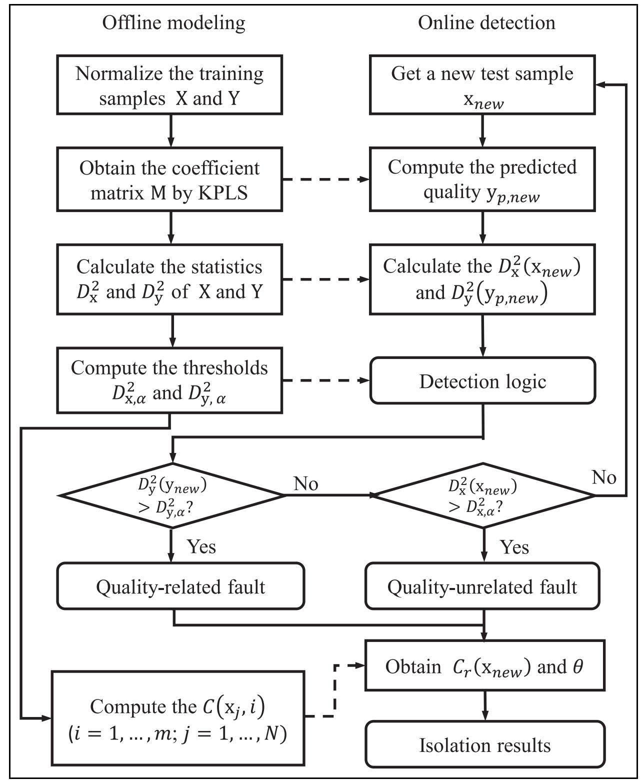

Figure 1. The quality-related non-linear fault diagnosis framework of k-NN.

Case study

This section applies a widely typical numerical and a real industrial Tennessee Eastman (TE) process benchmark to validate the effectiveness of our proposed method. Two fault evaluation indexes are adopted for performance evaluation. In the fault detection stage, our method KPLS-k-NN will be compared with the state-of-the-art approach total kernel projection to latent structures (TKPLS)[22] and the most recent SVD-based non-linear method modified kernel least squares (MKLS)[23] to show its superiorities. In the fault isolation stage, our MVCk-NN will be compared with VCk-NN. Besides, in the experiment, the confidence level α is set to 0.99.

Evaluation index

Two evaluation indexes—that is, the fault DR and the false alarm rate (FAR)—are used in our experiments, which are defined as follows

$$\mathrm{DR} = \frac{\text{Number of effective alarms}}{\text{Total faulty samples}} \times 100\% \tag{28}$$

$$\mathrm{FAR} = \frac{\text{Number of false alarms}}{\text{Total faulty samples}} \times 100\% \tag{29}$$

where an effective alarm represents a quality-related faulty sample is detected, while a false alarm represents a quality-unrelated faulty sample is detected. For performance evaluation, when a quality-related fault occurs, DR is adopted as the key indicator, while when a quality-unrelated fault happens, then FAR is used as

Table 1. Detection results of KPLS-*k*-NN, MKLS, and TKPLS for quality-related Fault 1 and Fault 2 (%).

Fault 1								Fault 2						
f	KPLS-*k*-NN		MKLS		TKPLS		f	KPLS-*k*-NN		MKLS		TKPLS	
	$D_{\mathbf{y}}^2$	$D_{\mathbf{x}}^2$	T_{mkls}^2	Q_{mkls}	$T_{ky}^2\&Q_{kr}$	$T_{ko}^2\&T_{kr}^2$		$D_{\mathbf{y}}^2$	$D_{\mathbf{x}}^2$	T_{mkls}^2	Q_{mkls}	$T_{ky}^2\&Q_{kr}$	$T_{ko}^2\&T_{kr}^2$
0.2	$(100)^a$	$(100)^a$	$(100)^a$	$(100)^a$	$(100)^a$	$(100)^a$	0.002	$(92)^a$	$(93.5)^a$	$(95)^a$	$(93)^a$	$(94.5)^a$	$(86.5)^a$
0.4	$(100)^a$	$(100)^a$	$(100)^a$	$(100)^a$	$(100)^a$	$(100)^a$	0.003	$(96)^a$	$(96.5)^a$	$(97)^a$	$(96.5)^a$	$(96.5)^a$	$(93)^a$
0.6	$(100)^a$	$(100)^a$	$(100)^a$	$(100)^a$	$(100)^a$	$(100)^a$	0.004	$(97.5)^a$	$(97.5)^a$	$(98.5)^a$	$(98)^a$	$(98)^a$	$(92.5)^a$
0.8	$(100)^a$	$(100)^a$	$(100)^a$	$(100)^a$	$(100)^a$	$(100)^a$	0.005	$(97)^a$	$(97.5)^a$	$(99)^a$	$(97.5)^a$	$(98.5)^a$	$(95.5)^a$

KPLS-*k*-NN: kernel partial least squares-*k*-nearest neighbor.
where "*a*" refers to DRs.

the key indicator. More details on DR and FAR can be referred to Wang and Jiao.[23]

Typical numerical example

The following numerical example introduced in Peng et al.[22] is applied

$$\begin{cases} x_1 \sim N\left(1, 0.01^2\right), x_2 \sim N\left(1, 0.01^2\right) \\ x_3 = \sin(x_1) + e_1 \\ x_4 = x_1^2 - 3x_1 + 4 + e_2 \\ x_5 = x_2^2 + \cos\left(x_2^2\right) + 1 + e_3 \\ y = x_3^2 + x_3 x_4 + x_1 + v \end{cases} \tag{30}$$

where $e_i \sim N(0, 0.001^2)(i = 1, 2, 3)$, $v \sim N(0, 0.005^2)$, and e_i and v denote the noises. $\mathbf{x} = [x_1, x_2, x_3, x_4, x_5]^{\mathrm{T}}$ is the process variable, and y is the quality variable.

From the above equation, we can see that y can only be affected by x_1, x_3, and x_4, but not by x_2 and x_5. Hence, when a fault occurs in x_1, it will have influence on quality variable y. While a fault happens in x_2, it will not do anything to y. We generate 400 normal samples as training samples for offline modeling, 400 normal samples as validation samples for selecting the hyper-parameters, and 400 test samples which include 200 normal samples and 200 faulty samples for online detection. The fault scenarios are as follows:

Fault 1: step bias occurs in x_1: $x_1 = x_1^* + f$.
Fault 2: ramp change occurs in x_1: $x_1 = x_1^* + (t - 200)f$.
Fault 3: step bias occurs in x_2: $x_2 = x_2^* + f$.
Fault 4: ramp change occurs in x_2: $x_2 = x_2^* + (t - 200)f$, where x_1^* and x_2^* are the normal values of x_1 and x_2, respectively, t $(201 \leqslant t \leqslant 400)$ is the sequence number of the tth sample, and f is the fault magnitude. It can be seen that Fault 1 and Fault 2 are quality-related faults, while Fault 3 and Fault 4 are quality-unrelated faults.

The model parameters are $N = 400$, $m = 5$, and $l = 1$. For the choice of hyper-parameters, that is, A, c, and k, we utilize cross-validation method to determine them. In detail, in the normal state, several sets of parameters with the minimum of sum of FARs of process variables and quality variables will be selected as the alternative hyper-parameters. In addition, we also need to consider the predictive performance of KPLS meanwhile. Therefore, we select one from the alternative hyper-parameters that has the minimum prediction error as final hyper-parameters. In this experiment, the hyper-parameters are set as $A = 2$, $c = 2 \times 10^4$, and $k = 5$. The significance level α is 0.01. To make the experiment more comprehensive, the magnitude of Fault 1 is changed in turn that $f = 0.2, 0.4, 0.6, 0.8$, while for Fault 2, $f = 0.002, 0.003, 0.004, 0.005$.

The results of quality-related faults

Fault detection. The detection results for Fault 1 and Fault 2 of KPLS-*k*-NN, MKLS, and TKPLS are displayed in Table 1. As shown in Table 1, the DRs of three methods to detect Fault 1 are all 100%, and the DRs of three methods for Fault 2 are almost above 90%. So, three methods all give right detection results.

Furthermore, without loss of generally, we set $f = 0.4$ in Fault 1 and observe the statistics values of three methods. From Figure 2, it is obvious that for these three methods, almost of all statistics of normal samples are below the threshold, while those faulty samples are significantly above the threshold. Therefore, three methods all have a good detection ability for quality-related faults.

Fault isolation. When $f = 0.4$, the Fault 1 isolation results of MVC*k*-NN and VC*k*-NN are shown in Figure 3. According to the setting of Fault 1, variable x_1 is the root cause of Fault 1, while x_3 and x_4 are also affected by x_1, but they are not root causes. In Figure 3(a), VC*k*-NN regards x_1, x_3, and x_4 as the root cause of Fault 1, while from Figure 3(b), MVC*k*-NN can only identify x_1 as the root cause of Fault 1. By contrast with VC*k*-NN, our proposed MVC*k*-NN has higher accuracy in the fault isolation task.

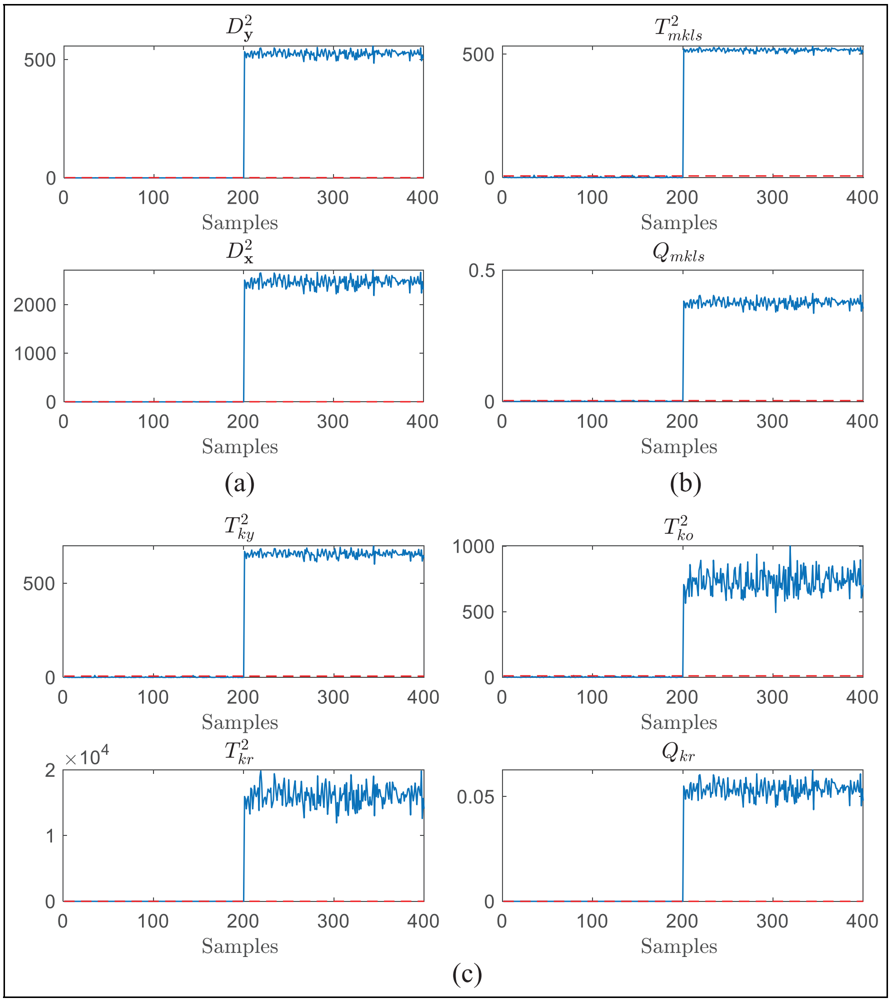

Figure 2. Detection results of quality-related Fault 1 with $f = 0.4$ by (a) KPLS-*k*-NN, (b) MKLS, and (c) TKPLS.

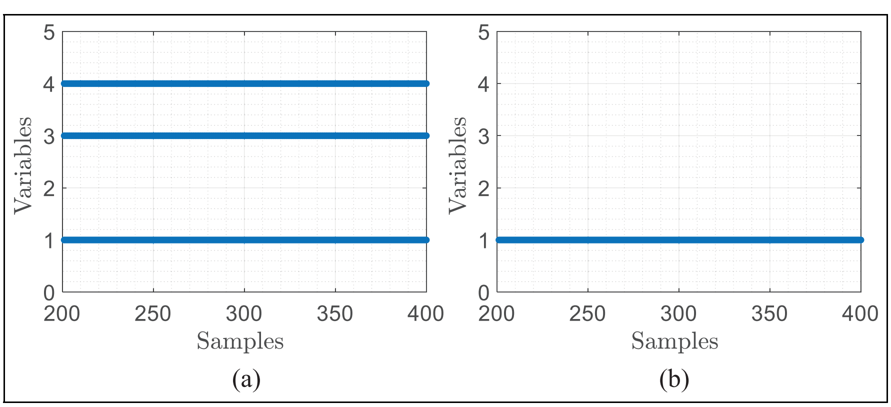

Figure 3. Isolation results of quality-related Fault 1 with $f = 0.4$ by (a) MVC*k*-NN and (b) VC*k*-NN.
A blue "·" represents that the variable is the root cause of the fault.

The results of quality-unrelated faults

Fault detection. The results to detect Fault 3 and Fault 4 using three methods are shown in Table 2. Because the quality-unrelated statistics of these three approaches are all over 90%, they can all warn that some process faults have happened in the system. The FARs of the statistics using TKPLS are over 90%, so TKPLS fails to identify the true natures of these faults.

For MKLS, although its FARs of the statistics T^2_{mkls} are all less than 1%, they are still not equal to 0. However, using KPLS-k-NN, the FARs of the statistics D^2_{y} are all 0, so it achieves the best performance in comparison methods.

We set $f = 0.003$ in Fault 2 to valid the fault detection and isolation performance for quality-unrelated faults. The values of three methods' statistics are shown in Figure 4. From Figure 4, when Fault 2 occurs, quality-unrelated statistics of three methods are all over their own thresholds, so they all indicate some faults have occurred. Because Q^2_{kr} of TKPLS is almost higher than its threshold in all faulty samples, it misunderstands that the faults are quality-related, so it fails. Since quality-related statistics of both the KPLS-k-NN method and MKLS are lower than the thresholds, they both can give the correct results: the system exists some quality-unrelated faults. Compared with T^2_{mkls} of MKLS, it is very obvious that D^2_{y} of KPLS-k-NN can achieve a better margin, because it is much lower than the threshold. Therefore, for quality-unrelated faults, KPLS-k-NN can make a clearer division between quality-related and quality-unrelated samples.

Fault isolation. When $f = 0.003$, the Fault 2 isolation results of two methods are displayed in Figure 5. Variable x_2 is the root cause of Fault 2. Although x_5 is not root cause, it will be affected by x_2. In Figure 3, VCk-NN considers x_2 and x_5 both as root cause variables, while our MVCk-NN only determines that x_2 is the root cause of Fault 2, which is consistent with the fact.

Through the above analysis of the experimental results, our method KPLS-k-NN does a much excellent job for quality-related fault detection task than other comparison methods. Besides, when KPLS-k-NN indicates some faults happen, our proposed MVCk-NN has high accuracy in fault root cause diagnosis and has prominent advantages over VCk-NN.

TE benchmark

TE process is a real simulation benchmark of industrial process, which has been widely utilized for the simulation and verification of various control and MSPM approaches,[39] and its structure flowchart is shown in Figure 6. The variables in this process contain two blocks of variables: the XMV block of 11 manipulated variables and the XMEAS block of 41 measured variables which include 22 process and 19 analysis variables. In this simulation, 22 process variables (XMEAS (1–22)) and 11 manipulated variables (XMV (1–11)) are chosen to be process input $\mathbf{X}$, and select purge gas analysis component G (XMEAS (35)) as the quality output $\mathbf{Y}$.

The training data set and validation data set contain 500 and 960 fault-free data samples, respectively. For the test data set, it is composed of 21 different fault sets, each of which includes 960 data samples and they are displayed in Table 3. The fault categories can be roughly divided into the following ones: step faults, random variation faults, slow drift faults, sticking faults, constant position faults, and some unknown faults. The detailed fault information is described in Downs and Vogel[40] and the website (http://depts.washington.edu/control/LARRY/TE/download.html).

To classify the process faults into the category of affecting $\mathbf{Y}$ and that of not affecting, the criterion of Zhou et al.[16] is adopted, that is, if $n_y/n_t > 10\%$, then the faults are considered to be quality-related, otherwise quality-unrelated, where n_y/n_t denotes the affected rate of $\mathbf{Y}$. According to the validation data set, the model parameters are $N = 500$, $m = 33$, and $l = 1$ and the hyper-parameters are set as $A = 10$, $c = 2 \times 10^4$, and $k = 10$.

The results of fault detection. For the quality-related faults, detection results of KPLS-k-NN, MKLS and TKPLS are presented in Table 4. We can see that TKPLS performs better than KPLS-k-NN and MKLS since all its FDRs are higher than the corresponding ones of the other methods. However, KPLS-k-NN and MKLS still provide satisfactory results, with most of their statistics above being 10%. The DRs of KPLS-k-NN are significantly superior to MKLS in IDV(6) and IDV(18).

For the quality-unrelated faults, Table 5 gives the detection results. We can see the corresponding statistics D^2_{x}, Q_{mkls}, and $T^2_{ko}\&T^2_{kr}$ of all comparative methods have very close values, indicating that the three statistics are equally effective to detect system faults. TKPLS fails to judge the correlation between faults and quality, because its statistics for these faults are all above 10%. For difficult-to-detect faults IDV(9), IDV(15)–IDV(17), both KPLS-k-NN and MKLS fail, but they enable to give correct results for the other faults. Compared with MKLS, KPLS-k-NN performs much better, because the FARs of KPLS-k-NN is much less than MKLS. Because KPLS-k-NN has good detection ability for different 21 kinds of faults, it indicates that the model trained by KPLS-k-NN has perfect generalization ability.

The results of fault isolation. We select IDV(1) to verify the fault isolation effect of MVCk-NN by compared with VCk-NN. IDV(1) is a step fault which represents the change of A/C feed ration. The root cause of IDV(1) is the variable of x_{25}. When IDV(1) happens, variables x_1, x_{26}, x_4, and x_{18} will deviate from the normal operation state in turn because of the influence of x_{25}. Figure 7

Table 2. Detection results of KPLS-*k*-NN, MKLS, and TKPLS for quality-unrelated Fault 3 and Fault 4 (%).

Fault 1								Fault 2						
f	KPLS-k-NN		MKLS		TKPLS		f	KPLS-k-NN		MKLS		TKPLS	
	$D_{\mathbf{y}}^2$	$D_{\mathbf{x}}^2$	T_{mkls}^2	Q_{mkls}	$T_{ky}^2\&Q_{kr}$	$T_{ko}^2\&T_{kr}^2$		$D_{\mathbf{y}}^2$	$D_{\mathbf{x}}^2$	T_{mkls}^2	Q_{mkls}	$T_{ky}^2\&Q_{kr}$	$T_{ko}^2\&T_{kr}^2$
0.2	$(0)^b$	$(100)^a$	$(0)^b$	$(100)^a$	$(100)^b$	$(100)^a$	0.002	$(0)^b$	$(92)^a$	$(1)^b$	$(92)^a$	$(91.5)^b$	$(92)^a$
0.4	$(0)^b$	$(100)^a$	$(0.5)^b$	$(100)^a$	$(100)^b$	$(100)^a$	0.003	$(0)^b$	$(96)^a$	$(0.5)^b$	$(96.5)^a$	$(96)^b$	$(96)^a$
0.6	$(0)^b$	$(100)^a$	$(0.5)^b$	$(100)^a$	$(100)^b$	$(100)^a$	0.004	$(0)^b$	$(95.5)^a$	$(0.5)^b$	$(95.5)^a$	$(95.5)^b$	$(96)^a$
0.8	$(0)^b$	$(100)^a$	$(0)^b$	$(100)^a$	$(100)^b$	$(100)^a$	0.005	$(0)^b$	$(97)^a$	$(0.5)^b$	$(97)^a$	$(96.5)^b$	$(97.5)^a$

KPLS-*k*-NN: kernel partial least squares-*k*-nearest neighbor.
where "*a*" refers to DRs and "*b*" refers to FARs.

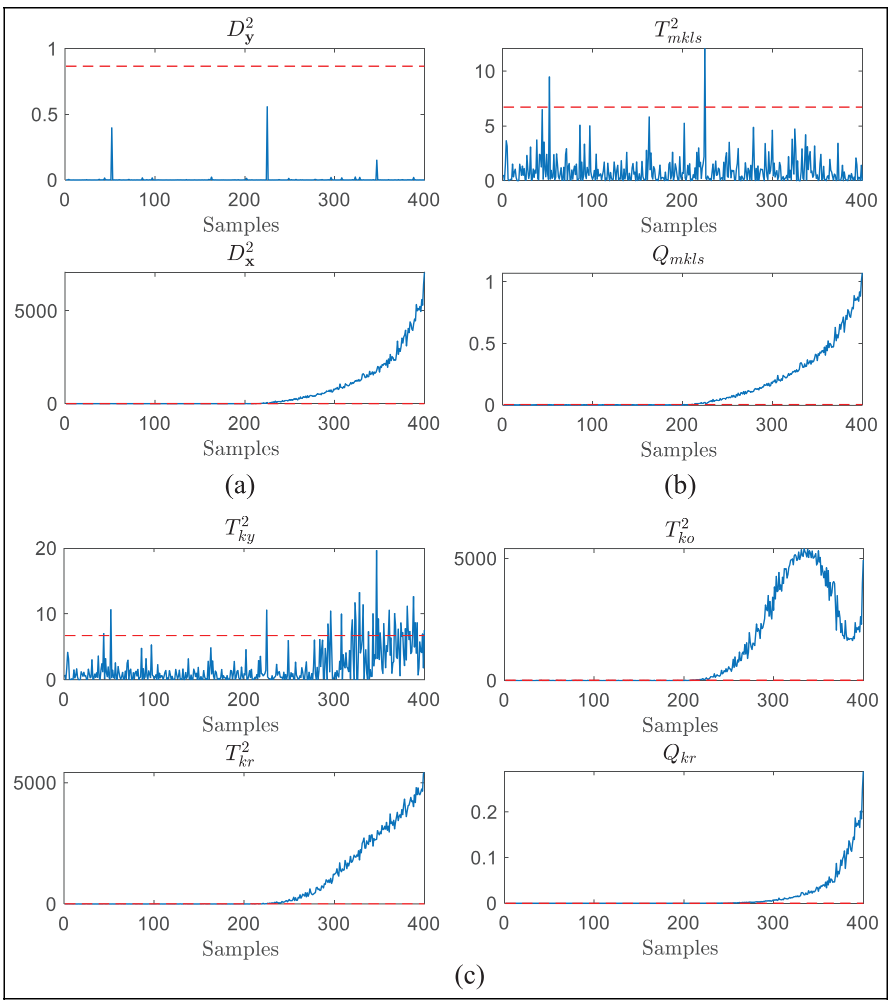

Figure 4. Detection results of quality-unrelated Fault 2 with $f = 0.003$ by (a) KPLS-*k*-NN, (b) MKLS, and (c) TKPLS.

displays the relative variable contributions of all process variables from 161th sample point to 960th sample point. From Figure 7(a), we can see that for the most part, VC*k*-NN identifies x_{25}, x_1, x_4, and x_{18} as the root cause of IDV(1). Whereas in Figure 7(b), MVC*k*-NN only needs to consider variables x_{25} and x_1 as the root cause. Therefore, our proposed MVC*k*-NN has a relatively better ability to locate the variables causing faults than VC*k*-NN.

According to the above results, we can conclude that the proposed method is effective in quality-related nonlinear fault diagnosis and has prominent advantages over traditional methods. Compared with the SVD-based methods, KPLS-*k*-NN directly utilizes the

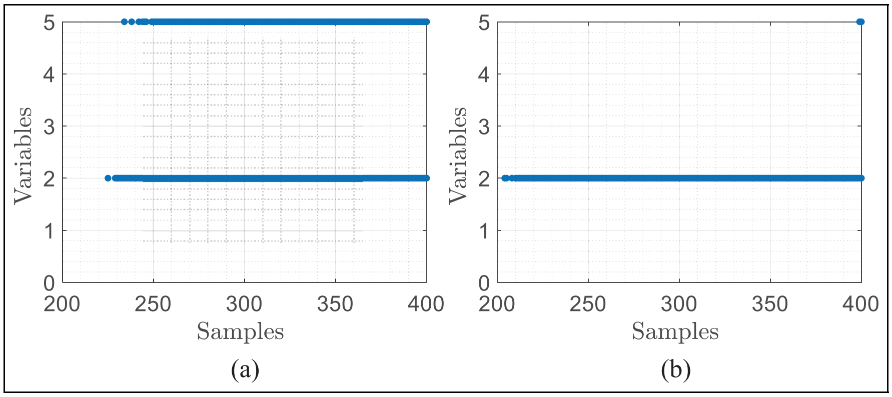

Figure 5. Isolation results of quality-related Fault 2 with $f = 0.003$ by (a) MVCk-NN and (b) VCk-NN.
A blue "·" represents that the variable is the root cause of the fault.

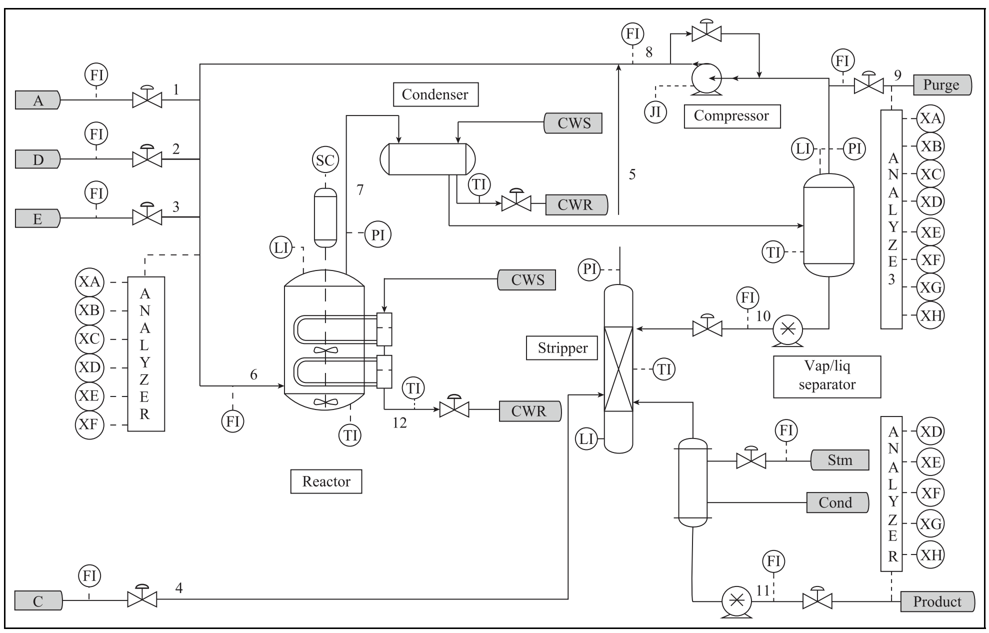

Figure 6. The structure flowchart of the TE process.

relationship among the predicted quality samples to design the statistics, which considers the information of the nearest neighbor structure. Besides, KPLS-k-NN directly detects the process space, avoiding the problem that the residual statistics may have a large component of variance, such as MKLS. Moreover, compared with VCk-NN, the proposed MVCk-NN has advantages in both variable contribution and threshold setting.

Conclusion

In this article, we present a novel quality-related non-linear fault diagnosis method based on k-NN, which is especially suitable for non-linear industrial processes with relatively small training samples. It consists of a new quality-related fault detection method KPLS-k-NN and a new fault isolation method MVCk-NN. First, KPLS is applied to establish a regression model between process variables and quality variables in order to obtain predictive quality samples. Then, FD-k-NN method is used to design two corresponding statistics, that is, $D_{\mathbf{x}}^2$ and $D_{\mathbf{y}}^2$ for the process space and the predicted quality space, respectively. They will make a judgment on whether there is a fault happening in the system, and whether the fault is related to quality when a fault occurs. When KPLS-k-NN detects faults, in order to locate root cause variables of faults, MVCk-NN is proposed by introducing the idea of ARCR into

Table 3. Fault types in the TE process.

Fault	Description	Type
IDV(1)	A/C Feed ratio, B composition constant (Stream 4)	Step
IDV(2)	B composition, A/C ratio constant (Stream 4)	Step
IDV(3)	D feed temperature (Stream 2)	Step
IDV(4)	Reactor cooling water inlet temperature	Step
IDV(5)	Condenser cooling water inlet temperature	Step
IDV(6)	A feed loss (Stream 1)	Step
IDV(7)	C header pressure loss (Stream 4)	Step
IDV(8)	A, B, C feed composition (Stream 4)	Random variation
IDV(9)	D feed temperature (Stream 2)	Random variation
IDV(10)	C feed temperature (Stream 4)	Random variation
IDV(11)	Reactor cooling water inlet temperature	Random variation
IDV(12)	Condenser cooling water inlet temperature	Random variation
IDV(13)	Reactor kinetics	Slow drift
IDV(14)	Reactor cooling water valve	Sticking
IDV(15)	Condenser cooling water valve	Sticking
IDV(16)	Unknown	Unknown
IDV(17)	Unknown	Unknown
IDV(18)	Unknown	Unknown
IDV(19)	Unknown	Unknown
IDV(20)	Unknown	Unknown
IDV(21)	Valve (Stream 4)	Constant position

TE: Tennessee Eastman.

Table 4. Detection results of KPLS-*k*-NN, MKLS, and TKPLS for quality-related faults in the TE process (%).

Fault number	n_y/n_t	KPLS-*k*-NN		MKLS		TKPLS	
		$D_{\mathbf{y}}^2$	$D_{\mathbf{x}}^2$	T_{mkls}^2	Q_{mkls}	$T_{ky}^2\&Q_{kr}$	$T_{ko}^2\&T_{kr}^2$
IDV(1)	22.72	$(82.25)^a$	$(99.75)^a$	$(34.88)^a$	$(99.88)^a$	$(99.75)^a$	$(99.75)^a$
IDV(2)	72.16	$(85.75)^a$	$(98.63)^a$	$(86.88)^a$	$(98.50)^a$	$(98.50)^a$	$(98.75)^a$
IDV(5)	14.98	$(13.88)^a$	$(36.63)^a$	$(23.50)^a$	$(33.38)^a$	$(33.38)^a$	$(33.25)^a$
IDV(6)	95.63	$(94.13)^a$	$(100)^a$	$(17.75)^a$	$(100)^a$	$(100)^a$	$(99.25)^a$
IDV(7)	18.23	$(29.88)^a$	$(100)^a$	$(32.63)^a$	$(100)^a$	$(100)^a$	$(100)^a$
IDV(8)	58.93	$(62.25)^a$	$(98.63)^a$	$(79.00)^a$	$(97.75)^a$	$(98.50)^a$	$(97.75)^a$
IDV(10)	11.99	$(60.12)^a$	$(63.88)^a$	$(26.75)^a$	$(59.38)^a$	$(73.25)^a$	$(87.25)^a$
IDV(12)	61.67	$(67.38)^a$	$(99.63)^a$	$(76.63)^a$	$(99.25)^a$	$(99.50)^a$	$(99.13)^a$
IDV(13)	66.67	$(83.00)^a$	$(95.38)^a$	$(79.13)^a$	$(95.25)^a$	$(95.50)^a$	$(94.75)^a$
IDV(18)	85.39	$(86.00)^a$	$(90.88)^a$	$(24.38)^a$	$(90.50)^a$	$(91.13)^a$	$(89.88)^a$
IDV(20)	12.48	$(10.00)^a$	$(65.25)^a$	$(28.38)^a$	$(63.50)^a$	$(59.75)^a$	$(50.25)^a$
IDV(21)	24.97	$(46.88)^a$	$(50.75)^a$	$(30.50)^a$	$(48.63)^a$	$(64.00)^a$	$(42.88)^a$

KPLS-*k*-NN: kernel partial least squares-*k*-nearest neighbor.
where "*a*" refers to DRs.

Table 5. Detection results of KPLS-*k*-NN, MKLS, and TKPLS for quality-unrelated faults in the TE process (%).

Fault number	n_y/n_t	KPLS-*k*-NN		MKLS		TKPLS	
		$D_{\mathbf{y}}^2$	$D_{\mathbf{x}}^2$	T_{mkls}^2	Q_{mkls}	$T_{ky}^2\&Q_{kr}$	$T_{ko}^2\&T_{kr}^2$
IDV(3)	7.74	$(1.13)^b$	$(14.37)^a$	$(7.25)^b$	$(14.00)^a$	$(15.38)^b$	$(6.88)^a$
IDV(4)	6.74	$(1.88)^b$	$(99.75)^a$	$(5.63)^b$	$(98.50)^a$	$(87.50)^b$	$(99.50)^a$
IDV(9)	4.99	$(0.75)^b$	$(11.13)^a$	$(5.63)^b$	$(10.88)^a$	$(12.13)^b$	$(7.12)^a$
IDV(11)	9.99	$(3.13)^b$	$(78.00)^a$	$(9.50)^b$	$(74.00)^a$	$(70.13)^b$	$(75.63)^a$
IDV(14)	6.24	$(7.63)^b$	$(100)^a$	$(3.38)^b$	$(100)^a$	$(100)^b$	$(100)^a$
IDV(15)	6.24	$(3.13)^b$	$(19.25)^a$	$(10.50)^b$	$(14.63)^a$	$(18.38)^b$	$(12.63)^a$
IDV(16)	3.75	$(39.25)^b$	$(59.88)^a$	$(20.00)^b$	$(52.50)^a$	$(63.13)^b$	$(74.38)^a$
IDV(17)	9.49	$(32.75)^b$	$(95.50)^a$	$(13.25)^b$	$(93.50)^a$	$(95.25)^b$	$(90.38)^a$
IDV(19)	8.36	$(0.50)^b$	$(26.13)^a$	$(1.00)^b$	$(19.88)^a$	$(43.13)^b$	$(13.63)^a$

KPLS-*k*-NN: kernel partial least squares-*k*-nearest neighbor.
where "*a*" refers to DRs and "*b*" refers to FARs.

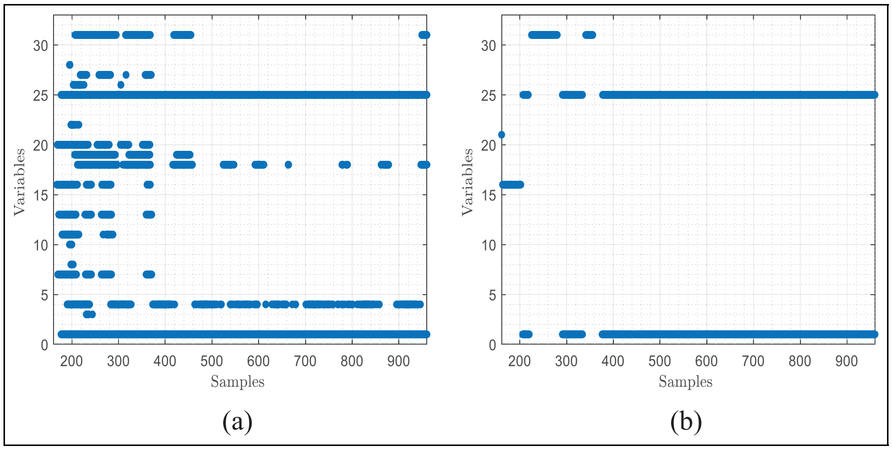

Figure 7. Fault isolation results for IDV(1) by (a) MVC*k*-NN and (b) VC*k*-NN.
A blue "·" represents that the variable is the root cause of the fault.

VC*k*-NN. Finally, the proposed method is applied to detect the faults in case study and the effectiveness is validated. Due to the superior fitting ability of the deep neural networks in comparison to KPLS, further work will focus on how to combine deep neural networks with *k*-NN rule to detect quality-related faults.

Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was supported by the National Natural Science Foundation of China (No. 61976213, No. 61772525 and No. 61906191).

ORCID iD

Zelin Ren https://orcid.org/0000-0001-7885-2658

References

1. Jiang Q, Yan X and Huang B. Review and perspectives of data-driven distributed monitoring for industrial plant-wide processes. *Ind Eng Chem Res* 2019; 58(29): 12899–12912.
2. Yin S, Ding SX, Xie X, et al. A review on basic data-driven approaches for industrial process monitoring. *IEEE Trans Ind Electron* 2014; 61(11): 6418–6428.
3. Zhang K, Peng K, Chu R, et al. Implementing multivariate statistics-based process monitoring: a comparison of basic data modeling approaches. *Neurocomputing* 2018; 290: 172–184.
4. Wise BM, Richer NL, Veltkamp DF, et al. A theoretical basis for the use of principal component models for monitoring multivariate processes. *Proc Contr Qual* 1990; 1(1): 41–51.
5. Xiu X, Yang Y, Kong L, et al. Laplacian regularized robust principal component analysis for process monitoring. *J Proc Contr* 2020; 92: 212–219.
6. Xiu X, Yang Y, Kong L, et al. Data-driven process monitoring using structured joint sparse canonical correlation analysis. *IEEE Trans Circ Syst II Exp Brief* 2021; 68(1): 361–365.
7. Ren Z, Zhang W and Zhang Z. A deep nonnegative matrix factorization approach via autoencoder for nonlinear fault detection. *IEEE Trans Ind Inform* 2020; 16(8): 5042–5052.
8. Xiu X, Fan J, Yang Y, et al. Fault detection using structured joint sparse nonnegative matrix factorization. *IEEE Trans Instrum Meas* 2021; 70: 1–11.
9. Yan S and Yan X. Quality-driven autoencoder for nonlinear quality-related and process-related fault detection based on least-squares regularization and enhanced statistics. *Ind Eng Chem Res* 2020; 59(26): 12136–12143.
10. Zhang K, Shardt YAW, Chen Z, et al. A KPI-based process monitoring and fault detection framework for large-scale processes. *ISA Trans* 2017; 68: 276–286.
11. Wang Y, Zhou D, Chen M, et al. Weighted part mutual information related component analysis for quality-related process monitoring. *J Proc Contr* 2020; 88: 111–123.
12. Chen Y and De Luca G. Technologies supporting artificial intelligence and robotics application development. *J Artif Intell Technol* 2021; 1(1): 1–8.
13. Ma L, Dong J and Peng K. A novel hierarchical detection and isolation framework for quality-related multiple faults in large-scale processes. *IEEE Trans Ind Electron* 2020; 67(2): 1316–1327.
14. Jiao J, Zhao N, Wang G, et al. A nonlinear quality-related fault detection approach based on modified kernel partial least squares. *ISA Trans* 2017; 66: 275–283.
15. Yao L, Shao W and Ge Z. Hierarchical quality monitoring for large-scale industrial plants with big process data. *IEEE Trans Neur Netw Learn Syst* 2021; 32(8): 3330–3341.

16. Zhou D, Li G and Qin SJ. Total projection to latent structures for process monitoring. *AIChE J* 2010; 56(1): 168–178.
17. MacGregor JF, Jaeckle C, Kiparissides C, et al. Process monitoring and diagnosis by multiblock PLS methods. *AIChE J* 1994; 40(5): 826–838.
18. Dong J, Zhang K, Huang Y, et al. Adaptive total PLS based quality-relevant process monitoring with application to the Tennessee Eastman process. *Neurocomputing* 2015; 154: 77–85.
19. Yin S, Ding SX, Zhang P, et al. Study on modifications of PLS approach for process monitoring. *IFAC Proc Vol* 2011; 44(1): 12389–12394.
20. Yin S, Zhu X and Kaynak O. Improved PLS focused on key-performance-indicator-related fault diagnosis. *IEEE Trans Ind Electron* 2015; 62(3): 1651–1658.
21. Schölkopf B, Smola AJ and Müller KR. Nonlinear component analysis as a kernel eigenvalue problem. *Neur Comput* 1998; 10: 1299–1319.
22. Peng K, Zhang K and Li G. Quality-related process monitoring based on total kernel PLS model and its industrial application. *Math Probl Eng* 2013; 2013: 707953.
23. Wang G and Jiao J. A kernel least squares based approach for nonlinear quality-related fault detection. *IEEE Trans Ind Electron* 2017; 64(4): 3195–3204.
24. Satapathy S, Loganathan D, Kondaveeti HK, et al. Performance analysis of machine learning algorithms on automated sleep staging feature sets. *CAAI Trans Intell Technol* 2021; 6(2): 155–174.
25. Ullah H, Ahmad B, Sana I, et al. Comparative study for machine learning classifier recommendation to predict political affiliation based on online reviews. *CAAI Trans Intell Technol* 2021; 6(3): 251–264.
26. Doreswamy, Hooshmand MK and Gad I. Feature selection approach using ensemble learning for network anomaly detection. *CAAI Trans Intell Technol* 2020; 5(4): 283–293.
27. He QP and Wang J. Fault detection using the *k*-nearest neighbor rule for semiconductor manufacturing processes. *IEEE Trans Semiconduct Manuf* 2007; 20(4): 345–354.
28. Zhang Z, Jiang T, Li S, et al. Automated feature learning for nonlinear process monitoring–an approach using stacked denoising autoencoder and *k*-nearest neighbor rule. *J Proc Contr* 2018; 64: 49–61.
29. Zhang C, Gao X, Li Y, et al. Fault detection strategy based on weighted distance of *k*-nearest neighbors for semiconductor manufacturing processes. *IEEE Trans Semiconduct Manuf* 2019; 32(1): 75–81.
30. Harrou F, Taghezouit B and Sun Y. Improved *k*NN-based monitoring schemes for detecting faults in PV systems. *IEEE J Photovolt* 2019; 9(3): 811–821.
31. Feng J and Li K. MRS-kNN fault detection method for multirate sampling process based variable grouping threshold. *J Proc Contr* 2020; 85: 149–158.
32. Song B, Tan S, Shi H, et al. Fault detection and diagnosis via standardized *k*-nearest neighbor for multimode process. *J Tai Inst Chem Eng* 2020; 106: 1–8.
33. Kariwala V, Odiowei PE, Cao Y, et al. A branch and bound method for isolation of faulty variables through missing variable analysis. *J Proc Contr* 2010; 20(10): 1198–1206.
34. Yan Z, Yao Y, Huang TB, et al. Reconstruction-based multivariate process fault isolation using Bayesian Lasso. *Ind Eng Chem Res* 2018; 57(30): 9779–9787.
35. Peng K, Zhang K, Li G, et al. Contribution rate plot for nonlinear quality-related fault diagnosis with application to the hot strip mill process. *Control Eng Pract* 2013; 21(4): 360–369.
36. Wang G, Jiao J and Yin S. Efficient nonlinear fault diagnosis based on kernel sample equivalent replacement. *IEEE Trans Ind Inform* 2019; 15(5): 2682–2690.
37. Zhou Z, Wen C and Yang C. Fault isolation based on *k*-nearest neighbor rule for industrial processes. *IEEE Trans Ind Electron* 2016; 63(4): 2578–2586.
38. Parzen E. On estimation of a probability density function and mode. *Ann Math Stat* 1962; 33(3): 1065–1076.
39. Chiang LH, Russell EL and Braatz RD. *Fault detection and diagnosis in industrial systems*. Berlin: Springer Science + Business Media, 2000.
40. Downs JJ and Vogel EF. A plant-wide industrial process control problem. *Comput Chem Eng* 1993; 17(3): 245–255.
41. Rosipal R. Kernel partial least squares for nonlinear regression and discrimination. *Neur Netw World* 2003; 13: 291–300.
42. Mugdadi AR and Ahmad IA. A bandwidth selection for kernel density estimation of functions of random variables. *Comput Stat Data Anal* 2004; 47(1): 49–62.