Chapter 14

Probabilistic Graphical Model for Continuous Variables

Most of the sampled data in complex industrial processes are sequential in time. Therefore, the traditional BN learning mechanisms have limitations on the value of probability and cannot be applied to the time series. The model established in Chap. 13 is a graphical model similar to a Bayesian network, but its parameter learning method can only handle the discrete variables. This chapter aims at the probabilistic graphical model directly for the continuous process variables, which avoids the assumption of discrete or Gaussian distributions.

This chapter expands the previous work in Chap. 13 from the random discrete variables to the random continuous variables. In addition to enhancing the effect of causal structure and parameter learning on the continuous variables, kernel density estimation is used to construct the node association strength of the causal graph network in the form of probability density. The conditional probability density is obtained from the mathematical operation between the low-dimensional probability density and the high-dimensional joint probability density. This non-parametric estimation method directly estimates the probability density of continuous variables and avoids the limitations of traditional Gaussian assumptions. Moreover, this chapter strictly derives the evaluation indicators for the KDE estimation quality. The proposed causal learning mechanism does not have any restrictions, such as linear, nonlinear, or distribution functions. It establishes an accurate causal probability graphical model to detect faults and find the root cause of the fault.

14.1 Construction of Probabilistic Graphical Model

14.1.1 Multivariate Casual Structure Learning

The first step of building a graphical model is to construct a causal topological relationship. The causal hypothesis model is a post-nonlinear model. It can determine the

causal relationship between multiple variables through hypothesis testing. Detailed information can be found in Chap. 13 (Chen et al. 2018).

Consider a model which represents the causal relationship between variables. Here a generative model is used to explain the data generation process. When the existing mechanism of the data model cannot be determined, the hypothetical model should be sufficiently versatile so that it can be adapted to approximate the actual data generation process. In addition, the model should be identified so that cause and effect can be distinguished.

In order to discover the causality of multiple variables in a complex system, a more generalized multivariable nonlinear acyclic causal model with internal additive noise is given same as Chap. 13. The model adopts the form of graph theory and Bayesian network structure. Assume that a directed acyclic graph (DAG) represents the relationship between multiple observed variables. Select a pair of variables X_i and $X_j, i, j = \{1, 2, \dots, n\}$ from the system, respectively. If X_i is X_j 's parent node and its data generating process is described in a post-nonlinear(PNL) mixing model. The generation process of X_i is $X_j = f_{j,2}(f_{j,1}(X_i) + e_j)$, where $f_{i,1}$ denotes the nonlinear effect of the causes, and $f_{i,2}$ denotes the invertible post-nonlinear distortion in variable X_i . e_j is the independent disturbance. Here it is applicable to a combination of hypothesis testing and nonlinear independent component analysis (ICA) to solve this problem (Shimizu et al. 2011). To describe in simplified language, it can be divided into two steps:

1. The nonlinear ICA method with constraints is used to calculate the interference e_j corresponding to the assumed causality $X_i \rightarrow X_j$;
2. The statistical independence test is used to determine the independent relationship between the estimated interference e_j and the assumed cause X_i .

For any pair of variables in the system, two causal assumptions can be made. The causality is assumed positive and negative, and the direction of the causality is determined by comparing the statistical information obtained by calculation. After $n(n-1)$ hypotheses and tests, the causality of all system variables is determined finally. Therefore, this multivariate nonlinear acyclic causal modeling method will not have the limitation of Bayesian network structure learning. It can effectively establish the causal structure of the process.

14.1.2 Probability Density Estimation

Section 14.1.1 completed the construction of the causal structure of the model. The complete graph model also should include the quantitative relationships between nodes which is described as probabilistic connection of nodes here. The probability density of the node variable is determined by the non-parametric probability density estimation method. Because the child node is affected by its parent node, the probabilistic connection relationship manifests itself in the conditional probability

density. Kernel Density Estimation (KDE) is a prominent method to estimate the non-parametric probability density. The explicit form of the density function is the main advantage of KDE method (Chen et al. 2018).

Let $X_1, X_2, X_3, \dots, X_n$ be a set of samples of the random variable X . Its density function $f(x), x \in R, X$ is unknown. The distribution density function $f(x)$ can be derived from its corresponding cumulative distribution function $F(x)$,

$$f(x) = \frac{dF(x)}{dx} \approx \frac{F(x+h) - F(x-h)}{2h}, \quad (14.1)$$

where $h > 0$ is the window width. The empirical distribution function $F_n(x) = \frac{1}{n} \sum_i I(X_i \leq x)$ is used to estimate $F(x)$. Substitute it into (14.1),

$$\begin{aligned} \hat{f}(x) &= \frac{dF(x)}{dx} \approx \frac{F(x+h) - F(x-h)}{2h} \\ &= \frac{1}{2nh} \sum_i I(x-h < X_i \leq x+h) \\ &= \frac{1}{nh} \sum_i K_0\left(\frac{X_i - x}{h}\right). \end{aligned} \quad (14.2)$$

(14.2) gives the KDE for $f(x)$ with a window width h and a kernel function $K_0 = \frac{1}{2}I(|u| \leq 1)$.

The more general kernel density estimate is

$$\hat{f}(x) = \frac{1}{nh} \sum_i^n K\left(\frac{X_i - x}{h}\right), \quad (14.3)$$

where $\hat{f}(x)$ gives the estimate of the probability density function. n, h, K are the number of samples, window width and kernel function.

Conditional probability density calculation requires additional mathematical operations. Similarly, consider two random sample sets $X_1, X_2, X_3, \dots, X_n$ and $Y_1, Y_2, Y_3, \dots, Y_n$, where X is cause variable and Y is effect variable. The joint probability density of x and y is defined as

$$\hat{f}(x, y) = \frac{1}{n} \sum_{i=1}^n \frac{1}{h_1 h_2} K\left(\frac{x - X_i}{h_1}, \frac{y - Y_i}{h_2}\right), \quad (14.4)$$

where h_1 and h_2 are the window width corresponding to the cause variable x and the effect variable y , respectively.

According to the definition of conditional probability, the conditional density $f(y|x)$ is obtained as follows:

Table 14.1 Common kernel functions

Number	Kernel function	Expression
1	Uniform	$\frac{1}{2}I(u \leq 1)$
2	Triangle	$(1 - u)I(u \leq 1)$
3	Gaussian	$\frac{1}{\sqrt{2\pi}}\exp(-\frac{1}{2}\mu^2)$
4	Epanechnikov	$\frac{3}{4}(1 - \mu^2)I(u \leq 1)$

$$f(y|x) = \frac{f(x, y)}{f(x)}. \quad (14.5)$$

The kernel function here affects the precision of kernel density estimation. How to select an appropriate kernel function is an important issue. Usually, the following properties should be considered: symmetry, non-negative, and normality (Zeng et al. 2017). The mathematical description of common kernel functions is given in Table 14.1 (Jiang and Nicholas 2014).

It can be seen from the KDE expression that the kernel function K , sample size n and its window width h are the main contributing factors of $f(x)$. Once the number of samples n is fixed, K and h directly affect the accuracy of the system model parameters. Furthermore, the effectiveness of fault detection and root cause diagnosis will fluctuate directly. Therefore, in order to estimate the probability density more accurately and improve the estimation quality of KDE, a KDE evaluation criterion is given in the next section. There are already data showing that the choice of kernel function has a negligible effect on the result of kernel density estimation (Silverman 1998), so the optimization of K is not considered here.

14.1.3 Evaluation Index of Estimation Quality

According to the definition of kernel density, consider the following two cases: (1) the value of the window width h is very large. The average compression transformation $\frac{x-X_i}{h}$ can remove the local details of the probability density function, which results in the smoothness of probability density estimation curve. A relatively low resolution is shown at this case, and the estimation deviation is enlarged; (2) the value of the window width is very small. On the contrary, the influence of the randomness of probability density will increase, and the important characteristics of density will be masked. It causes the larger fluctuation of density estimation and the stability is easy to be deteriorated. The estimation variance is too large at this case (Jiang and Nicholas 2014).

The requirements about the accurate estimation include much closer to the true values and remaining stable for different observations. These two attributes are described by the estimated deviation and variance which are given as

$$\begin{aligned}\text{Bias}\{\hat{f}(x)\} &= \mathbb{E}[\hat{f}(x)] - f(x) \\ \text{Var}\{\hat{f}(x)\} &= \mathbb{E}[\hat{f}(x)]^2 - [\mathbb{E}\hat{f}(x)]^2.\end{aligned}\tag{14.6}$$

The probability density function of the child nodes in the causal model is affected by the parent nodes. Its probability density usually is multidimensional. Consider a two-dimensional kernel density function $f(x, y)$ as an example. Its deviation and variance are

$$\begin{aligned}\text{Bias}\{\hat{f}(x, y)\} &= \mathbb{E}[\hat{f}(x, y)] - f(x, y) \\ \text{Var}\{\hat{f}(x, y)\} &= \mathbb{E}[\hat{f}(x, y)]^2 - [\mathbb{E}\hat{f}(x, y)]^2.\end{aligned}\tag{14.7}$$

Here the mean square integral error (MISE) is introduced as the evaluation index of KDE. The MISE index has a unique advantage to evaluate the difference between the estimated function and the true function. At the same time, it also guarantees the fitness and smoothness of kernel estimation.

One-dimensional MISE is defined as

$$\text{MISE}[\hat{f}(x)] = \mathbb{E} \int [\hat{f}(x) - f(x)]^2 dx.\tag{14.8}$$

Two-dimensional MISE is defined as

$$\text{MISE}[\hat{f}(x, y)] = \mathbb{E} \iint [\hat{f}(x, y) - f(x, y)]^2 dx dy.\tag{14.9}$$

The above MISE indices are simplified as, and the details can be found from the supporting information in Chen et al. (2018),

$$\begin{aligned}\text{MISE}[\hat{f}(x)] &= \int \text{Var}(\hat{f}(x)) + \int \text{Bias}^2(\hat{f}(x)) dx \\ &= \frac{1}{nh} \int K^2(t) dt + \frac{1}{4} h^4 \left[\int t^2 K(t) dt \right]^2 \int [f''(x)]^2 dx\end{aligned}\tag{14.10}$$

$$\begin{aligned}\text{MISE}[\hat{f}(x, y)] &= \frac{1}{nh_1 h_2} \int K^2(t) dt + \frac{1}{4} h_1^4 h_2^4 \\ &\quad \times \left[\int t^2 K^2(t) dt \right]^2 \iint (\nabla f(x, y))^2 dx dy.\end{aligned}\tag{14.11}$$

It is found from (14.10) and (14.11) that the values of the functions $\int K^2(t) dt$ and $\int t^2 K(t) dt$ are related to the kernel function K . They are not difficult to calculate if the mathematical expression of kernel function is substituted into the above equations. Generally speaking, window width h has a greater impact on MISE value, so optimizing h is critical. Here (14.10) and (14.11) are also used as optimization objectives to find the best window width h .

For one-dimensional probability density, let $d \left(\text{MISE} \left[\hat{f}(x) \right] \right) / dh = 0$. Then

$$h_{opt} = \sqrt[5]{\frac{\int K^2(t)dt}{n[\int t^2 K(t)dt]^2 \int f''(x)^2 dx}}. \quad (14.12)$$

For two-dimensional probability density, let

$$\begin{aligned} \frac{\partial \text{MISE}[\hat{f}(x, y)]}{\partial h_1} &= h_1^3 h_2^4 \left(\int t^2 K(t)dt \right)^2 \iint (\nabla f(x, y))^2 dx dy \\ &\quad - \frac{1}{n h_1^2 h_2} \int K^2(t)dt \\ &= 0, \\ \frac{\partial \text{MISE}[\hat{f}(x, y)]}{\partial h_2} &= h_2^3 h_1^4 \left(\int t^2 K(t)dt \right)^2 \iint (\nabla f(x, y))^2 dx dy \\ &\quad - \frac{1}{n h_2^2 h_1} \int K^2(t)dt \\ &= 0. \end{aligned} \quad (14.13)$$

Then

$$\begin{aligned} h_1^{opt} &= \sqrt[5]{\frac{\int K^2(t)dt}{n h_2^5 (\int t^2 K(t)dt)^2 \iint (\nabla f(x, y))^2 dx dy}} \\ h_2^{opt} &= \sqrt[5]{\frac{\int K^2(t)dt}{n h_1^5 (\int t^2 K(t)dt)^2 \iint (\nabla f(x, y))^2 dx dy}}. \end{aligned} \quad (14.14)$$

If the kernel function is predetermined, $\frac{\int K^2(t)dt}{(\int t^2 K(t)dt)^2} = C(k)$ is a constant. Usually the true probability density functions $f(x)$ and $f(x, y)$ are unknown. The estimated probability density function (14.3) and (14.4) are substituted into (14.12) and (14.14), respectively. Then the optimal parameter h for one-dimensional estimation or h_1 and h_2 for two-dimensional estimation are obtained.

14.2 Dynamic Threshold for the Fault Detection

Generally speaking, the process variables show obvious difference in their measurements in the normal operation and faulty operation. Then the measurement difference must be reflected in the probability density distribution. System failure detection is to find their differences based on the appropriate thresholds. Here, it is not feasible to use the confidence interval of the normal state to directly distinguish the fault. The actual process data are usually accompanied by a lot of noise, the distribution is not

ideal even in the normal operation. Therefore, its confidence cannot be completely described as a constant horizontal line. The constant confidence line is further difficult to distinguish the normal operation and the fault operation. Therefore, the idea of dynamic threshold is introduced. Fused Lasso (FL) method is common to denoise in the field of signal processing. Here it is used to design the dynamic confidence limits. It can provide the required reasonable range for each node based on the normal data.

The Fused Lasso Signal Approximator (FLSA) aims at eliminating noise and smoothing data (Bensi et al. 2013). The real-valued observations $\mathbf{y} = \beta\mathbf{x}$ is obtained by finding the sequence $\beta_1, \dots, \beta_N$ that minimizes the criterion,

$$J_{FL} = \frac{1}{2} \sum_{k=1}^N (\mathbf{y}_k - \beta_k \mathbf{x}_k)^2 + \lambda_1 \sum_{k=1}^N |\beta_k| + \lambda_2 \sum_{k=2}^N |\beta_k - \beta_{k-1}|, \quad (14.15)$$

where λ_1 and λ_2 are tuning parameters, $\mathbf{x}_1, \dots, \mathbf{x}_N$ is the feature variables. The objective of J_{FL} consists of three parts: $\frac{1}{2} \sum_{k=1}^N (\mathbf{y}_k - \beta_k \mathbf{x}_k)^2$ is the traditional index of the least squares algorithm. It strives for the regression accuracy of the model for all the existed measurements $[\mathbf{x}_k, \mathbf{y}_k]$. The last two parts $\lambda_1 \sum_{k=1}^N |\beta_k| + \lambda_2 \sum_{k=2}^N |\beta_k - \beta_{k-1}|$ encourages the sparsity of regression coefficients and their differences. The parameters λ_1 and λ_2 are adjusted to trade-off the regression accuracy and denoising power. (14.15) is totally a denoising problem if $\lambda_1 = 0$.

Here the hidden Markov model (HMM) and the maximum likelihood estimation method are used for optimization calculation. The HMM posits an emission probability $Pr(\mathbf{y}_k | \beta_k)$ that is a standard normal distribution, and a transition probability $Pr(\beta_{k+1} | \beta_k)$ that is double exponential with parameter λ_2 (where Pr denotes probability).

The Viterbi algorithm is a typical dynamic programming algorithm for this HMM problem, which the detailed description be found in (Rabiner et al. 1989). The objective function (14.15) is rewritten as maximization in a more general form,

$$J_{FL} = \sum_{k=1}^N e_k(\beta_k) - \lambda_2 \sum_{k=2}^N d(\beta_k, \beta_{k-1}), \quad (14.16)$$

where $e_k(\mathbf{b}) = \sum_{i=1}^R \mathbf{y}_{ik} v_i(\mathbf{b})$.

Denote the variable sequences $(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k)$ as the shorthand $\mathbf{x}_{1:k}$. Rewrite the criterion (14.16) as follows:

$$\begin{aligned} J_{FL} &= \max_{\beta_{1:N}} \left[\sum_{k=1}^N e_k(\beta_k) - \lambda_2 \sum_{k=2}^N d(\beta_k, \beta_{k-1}) \right] \\ &= \max_{\beta_N} [e_N(\beta_N)] + \max_{\beta_{1:(N-1)}} \left[\sum_{k=1}^{N-1} e_k(\beta_k) - \lambda_2 \sum_{k=2}^N d(\beta_k, \beta_{k-1}) \right] \end{aligned} \quad (14.17)$$

and

$$\begin{aligned}
f_N(\beta_N) &:= \max_{\beta_{1:(N-1)}} \left[\sum_{k=1}^{N-1} e_k(\beta_k) - \lambda_2 \sum_{k=2}^N d(\beta_k, \beta_{k-1}) \right] \\
&= \max_{\beta_{N-1}} [e_{N-1}(\beta_{N-1}) + \lambda_2 d(\beta_N, \beta_{N-1})] \\
&\quad + \max_{\beta_{1:(N-2)}} \left[\sum_{k=1}^{N-2} e_k(\beta_k) - \lambda_2 \sum_{k=2}^{N-1} d(\beta_k, \beta_{k-1}) \right].
\end{aligned} \tag{14.18}$$

The definitions of functions $f_{N-1}(\beta_{N-1}), f_{N-2}(\beta_{N-2}), \dots, f_2(\beta_2)$ are similar to $f_N(\beta_N)$. The maximization problem is solved further iteratively. It is summarized by introducing the intermediate functions with k ranging from 2 to N ,

$$\begin{aligned}
\delta_1(\mathbf{b}) &:= e_1(\mathbf{b}) \\
\psi_k(\mathbf{b}) &:= \arg \max_{\tilde{\mathbf{b}}} [\delta_{k-1}(\tilde{\mathbf{b}}) - \lambda_2 |b - \tilde{\mathbf{b}}|] \\
f_k(\mathbf{b}) &:= \delta_{k-1}(\psi_k(\mathbf{b})) - \lambda_2 |b - \psi_k(\mathbf{b})| \\
\delta_k &:= e_k(\mathbf{b}) + f_k(\mathbf{b}).
\end{aligned} \tag{14.19}$$

The functions $\psi_k(\cdot)$ take part in the backward pass of the algorithm. This backward pass computes $\hat{\beta}_1, \dots, \hat{\beta}_N$ through a recursion identical to that of the Viterbi algorithm for HMMs:

$$\begin{aligned}
\hat{\beta}_N &= \arg \max_{\mathbf{b}} \{\delta_N(\mathbf{b})\} \\
\hat{\beta}_k &= \psi_{k+1}(\hat{\beta}_{k+1}) \quad \text{for } k = N-1, N-2, \dots, 1.
\end{aligned} \tag{14.20}$$

So far, the above FL theory is implemented to obtain the dynamic threshold of the data model. During the process of fault detection, the KDE estimated probability values are the input variable of the FLSA algorithm for smoothing. The influence of data noise on the estimated probability density function is eliminated and a credible threshold is found to distinguish the normal operation and the faulty operation.

14.3 Forward Fault Diagnosis and Reverse Reasoning

Detailed theoretical supports have been supplemented enough in last section, including the construction of probability graph models, the selection of probability density estimation evaluation indicators and parameter optimization, and the setting of dynamic thresholds for fault detection. The established model structure is determined by the causal direction between operating units, which represents the qualitative relationship between nodes. The non-parametric KED estimation is used to obtain the parameters of the graph model, i.e., the causal probability relationship. Probability can quantitatively describe the dependence between process variables. The evalu-

Fig. 14.1 The overall framework

ation index of the probability relationship estimation is derived and calculated to ensure the accuracy of the graphical model.

Now this section combines and implements the above theoretical methods into a certain fault detection and diagnosis framework, which can be used to diagnose abnormal events in the system and locate the root cause of the fault. The overall framework of the proposed method is represented in Fig. 14.1.

The main steps for fault detection and root tracing are summarized based on the detail flow chart in Fig. 14.2,

1. Construct a cause-effect network structure for the selected process variables from the industrial process;
2. List all the probability density functions that need to be estimated, including one-dimensional densities for root nodes, multidimensional joint densities, or the corresponding conditional probability densities for child nodes;
3. Estimate the (conditional) probability densities of each node based on KDE method;
4. Calculate the dynamic threshold for the health status of each node by input all the density values to FLSA;
5. Collect test data and detect whether faults occur compared with the dynamic threshold;
6. Reverse reasoning based on the graph model in the case of failure. Starting from the faulty node, check which parent nodes of the faulty node is faulty in turn. Remove all non-faulty parent nodes and clarify the fault propagation path until the fault root is found.

Fig. 14.2 Flowchart for detecting and tracing faults

Fig. 14.3 The casual structure of partial TE process

14.4 Case Study: Application to TEP

The proposed methods are verified on Tennessee Eastman (TE) process simulator. The TE process contains a total of 52 process variables and measurement variables. Eight variables in the reactor module are selected to test the causal structure, same as Chap. 13. The physical meanings of these variables are listed in Table 14.2. According to the causal analysis method, it is not difficult to obtain the causal relationship between eight variables (the detail analysis also can be found in Chap. 13). The corresponding topology is shown in Fig. 14.3.

Table 14.2 Process manipulated variables

Variable (Symbol in the Fig. 3)	Physical meaning	Units
$x_1(v_5)$	Recycle flow	km^3/h
$x_2(v_6)$	Reactor feed rate	km^3/h
$x_3(v_7)$	Reactor pressure	kPa
$x_4(v_8)$	Reactor level	%
$x_5(v_9)$	Reactor temperature	$^{\circ}\text{C}$
$x_6(v_{12})$	Product separator level	%
$x_7(v_{20})$	Compress work	KW
$x_8(v_{21})$	Reactor cooling water outlet temperature	$^{\circ}\text{C}$

List all the probability density function and conditional probability density of nodes in the causal graph. In total, $f(x_2)$, $f(x_8)$, $f(x_4|x_2)$, $f(x_5|x_8)$, $f(x_7|x_5)$, $f(x_3|x_5)$, $f(x_1|x_3)$, $f(x_6|x_3)$ need to be estimated. Here the root nodes x_2 and x_8 have one-dimensional probability density function. Optimize the window width h to obtain an accurate probability estimate.

The training data set contains 960 samples in the normal operation. These data are used to obtain the KDE of the model. Combine the causal structure constructed in the previous step to get a complete graphical model. Fault IDV(4) is a minor fault which is used as a test sample to verify the effectiveness and sensitiveness of the proposed method to minor faults. The fault IDV(4), a step change of the reactor cooling water inlet temperature, is introduced in the middle of the reaction. Then 960 samples are obtained as the testing data set, in which the first 480 samples are normal and the following 480 are faulty data.

In order to be able to trace the root cause of the fault, the child nodes must be selected here to test the fault. Randomly select one of the child nodes x_7 of the graphical model as the experimental object. According to the causal structure, it is easy to see that x_7 is directly related to x_5 . Here x_5 is the parent node of x_7 , so first calculate the conditional probability density $f(x_7|x_5)$. Figure 14.4 gives the graphical representation of the probability relationship between these two variables. Figure 14.4a depicts the probability density of normal data and fault data as a function of sampling time. Based on the fusion lasso method, the obtained KDE estimation is used as a rough signal for denoising and restoration. The crossed line in Fig. 14.4b represents the KDE recovered after denoising, which is set as the dynamical threshold. It can be clearly seen that after about 480 samples, the conditional probability of x_7 has exceeded the normal limit. Based on the FLSA method, the obtained KDE estimation is used as a rough signal for denoising and restoration.

Fault tracing refers to finding the root cause of failure in x_7 . The existing graph model can clearly show the causal relationship between nodes, so the propagation path of the fault can be easily analyzed. Carry out the reverse reasoning based on the established causal structure parameter model. Start from the failure variable and

(a) 3D plot of joint probability density of x_7 and x_5 (b) Conditional probability of x_7 under x_5 corresponds to the sampling time**Fig. 14.4** Conditional probability of x_7 under x_5

calculate the probability density function of its parent node in turn. The probability density curves obtained under normal and fault conditions are compared to determine whether the variables on each path are faulty. Continue this step until finding the root cause of the failure. In order to conversely infer the roots of fault x_7 , it is necessary to calculate $f(x_5|x_8)$, $f(x_5|x_2)$, $f(x_2)$, $f(x_8)$ separately. Simulation results are shown in Fig. 14.5.

From the detection result graph, the true propagation path of the fault can be analyzed. The test shows that the root of the fault is x_8 . Corresponding to the physical meaning of the variable, the root cause is the temperature of the cooling water, and fault IDV(4) is a step change in the temperature of the cooling water. The result is consistent with the actual process.

Fig. 14.5 Conditional probability densities of x_5 under x_8 , x_5 under x_2 ; probability densities of x_2, x_8

(a) Conditional probability density of x_5 under x_8

(b) Conditional probability density of x_5 under x_2

(c) Probability density of x_8

(d) Probability density of x_2

14.5 Conclusions

This chapter proposes a probability graph model directly for the continuous process variables aiming at the fault detection and root tracing. The model structure is determined by the causal relationship, and the probability relationship in the model is determined by the KDE method. For the child nodes in the causal structure, i.e., variables affected by other nodes, the conditional probability density functions are calculated based on the multidimensional joint probability density and the low-dimensional probability density. It reflects the strength relationship of the causal connection between the variables. An MISE index is rigorously derived to evaluate the estimation accuracy of KDE and optimize the KDE parameters. A dynamic threshold is constructed based on the FLSA algorithm to check the change of probability density, further to detect the fault. The experiment results in the TE process show that the proposed method not only accurately detects the occurrence of the failure, but also succeeds in finding its root cause.

References

- Bensi M, Kiureghian AD, Straub D (2013) Efficient bayesian network modeling of systems. *Reliab Eng Syst Saf* 112:200–213
- Chen X, Wang J, Zhou J (2018) Process monitoring based on multivariate causality analysis and probability inference. *IEEE Access* 6:6360–6369
- Chen X, Wang J, Zhou J (2018) Probability density estimation and bayesian causal analysis based fault detection and root identification. *Ind Eng Chem Res* 57(43):14656–14664
- Jiang W, Nicholas Z (2014) A probabilistic graphical model based stochastic input model construction. *J Comput Phys* 272(10):664–685
- Rabiner LR (1989) A tutorial on hidden markov models and selected applications in speech recognition. *Proc IEEE* 77:257–286
- Shimizu S, Inazumi T, Sogawa Y, Hyvarinen A, Kawahara Y, Washio T, Hoyer PO, Bollen K (2011) DirectLiNGAM: a direct method for learning a linear non-gaussian structural equation model. *J Mach Learn Res* 12(2):1225–1248
- Silverman BW (1998) *Density estimation for statistics and data analysis*. Routledge, Boca Raton
- Zeng J, Luo S, Cai J, Krüge U (2017) Nonparametric density estimation of hierarchical probabilistic graph models for assumption-free monitoring. *Ind Eng Chem Res* 56(5):1278–1287

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.