Journal of Process Control 57 (2017) 15–25

Contents lists available at ScienceDirect

Journal of Process Control

journal homepage: www.elsevier.com/locate/jprocont

Probabilistic density-based regression model for soft sensing of nonlinear industrial processes

Xiaofeng Yuan [a], Yalin Wang [a,*], Chunhua Yang [a], Weihua Gui [a], Lingjian Ye [b]

[a] College of Information Science and Engineering, Central South University, Changsha 410083, Hunan, China
[b] Ningbo Institute of Technology, Zhejiang University, Ningbo 315100, Zhejiang, China

ARTICLE INFO

Article history:
Received 15 December 2016
Received in revised form 3 May 2017
Accepted 5 June 2017

Keywords:
Soft sensor
Quality prediction
Gaussian mixture model
Weighted Gaussian regression
Locally weighted learning

ABSTRACT

Process nonlinearity is a challenging issue for soft sensor modeling of industrial plants. Traditional nonlinear soft sensing methods are not achieved through the probabilistic manner, which only give single point estimation for output variables but do not provide the prediction uncertainty. To meet the probabilistic soft sensor requirement, a novel density-based regression method, which is called weighted Gaussian regression (WGR), is proposed in this paper. By taking the weights of training samples into consideration, a local weighted Gaussian model (WGM) is first built to model the joint density $P(x, y)$ of input and output variables around the query sample. Then, the output variables can be estimated by taking the conditional distribution $P(y|x)$. The new method can successfully approximate the nonlinear relationship between output and input variables. Moreover, WGR can provide more detailed information of uncertainty for the prediction. The effectiveness and flexibility of WGR are validated through a numerical example and an industrial debutanizer column process.

1. Introduction

Nowadays, online accurate identification of quality variables is of significant importance for successful monitoring and control of industrial processes [1–3]. In many situations, it is difficult to timely measure the quality variables due to the measurement difficulties like severe measuring circumstance, expensive analyzer cost and large time delay, etc. As a result, these measurement limitations may cause a lot of disadvantages on the process production, like product quality degradation, energy loss, toxic byproduct generation, and safety risk problem. Thus, soft sensors have been developed to estimate the difficult-to-measure variables y through those easy-to-measure process variables x by inferential models [4–7]. As alternatives for hardware sensors, soft sensors can provide highly efficient and low-cost prediction for the quality variables. In general, there are mainly three different categories of soft sensors, which are first-principle models [8–10], data-driven models [11–13] and hybrid models [14–16]. First-principle models are often difficult to obtain, especially for complicated modern industrial processes. Instead, data-driven soft sensors can be easily obtained thanks to the access of vast process data, which can largely reduce the amount of engineering knowledge. Hence, they have gained more and more popularity in both academia and industry.

For a soft sensor, the main task is to train a regression model $y = f(x)$ between y and x from existing datasets. Hence, the key factor to a successful soft sensor is choosing an appropriate regression model for the specified process. Multivariate statistical methods like principal component regression (PCR) [17] and partial least squares regression (PLSR) [18], artificial neural networks (ANN) [19] and support vector machine (SVM) [20] are some typical regression-based approaches. Numerous of successful applications have been reported in processes like chemical engineering, biochemical engineering, metallurgical engineering pharmaceutical industries by these approaches. [21–24] However, most of these methods are carried out in deterministic ways. The random noises and variable uncertainty are not taken into consideration for modeling. Hence, it can only give single point predictions for the outputs. For many applications, it is important to provide the quantification about the prediction uncertainty, like the probabilistic bounds. Actually, process data are often contaminated by random noises and uncertainty as a result of measurement variations and transmission disturbances. It is more reasonable to take these variables as random variables and carry out data modeling in the probabilistic framework. In this way, the vector regression function $y = f(x)$ is expressed by density-based methods [25–27].

* Corresponding author.
E-mail address: ylwang@csu.edu.cn (Y. Wang).

http://dx.doi.org/10.1016/j.jprocont.2017.06.002
0959-1524/

In the probabilistic modeling framework, input and output variables are described by means of the probability distribution or probability density. To estimate the function $y=f(x)$, the joint density of input and output $P(x,y)$ is first estimated in the density-based methods. Then, the conditional distribution of output over input variables can be obtained as $P(y|x)$. Given a particular input vector x_{new}, the output can be predicted from the conditional density $P(y_{new}|x_{new})$ by easily substituting x_{new} for x [25]. Different from the regression-based methods, which only give single point estimation for y_{new} , density-based approaches can provide a probabilistic distribution of the output variables y_{new}. If one wishes to obtain a single prediction of y_{new} by density-based methods, it is easily to calculate the expectation of y_{new} given x_{new}. That is $\hat{y}_{new} = E(y_{new}|x_{new})$. There are a lot of advantages to exploit density-based methods for data modeling. For example, it can handle dataset with missing values by expectation-maximization algorithm [28]. Also, it is very easy to estimate any relationship between two subsets of these variables in vector (x,y). Furthermore, as mentioned before, it can give not only a point estimation for the output, but also a probabilistic distribution for it.

Hence, the most important thing of density-based methods is to find the approximate density model to describe the data structure. As a basic density function, Gaussian probabilistic distribution is the most commonly used data density model for continuous variables in reality. For many process data, they can be roughly approximated by Gaussian distribution. Since Gaussian distribution is simple to implement, it can largely reduce the complexity of modeling procedure. Although Gaussian density has a lot of successful applications, it suffers from significant limitations when it is used to model complicated industrial data due to its linear assumption of data relationship. As most industrial processes are nonlinear naturally, approximation by simple Gaussian model is not sufficient to capture data structure accurately. Hence, it is desirable to exploit a more robust density model to handle process nonlinearity. To deal with this problem, Gaussian mixture model (GMM) [29] has been developed to better characterize data structure by simple linear superposition of Gaussian components. GMM can give rise to very complex data distributions. Moreover, it can approximate almost any continuous density to arbitrary accuracy by using a sufficient number of Gaussians, and adjusting their means and covariances as well as the coefficients in the linear combination. Therefore, GMM is a useful modeling tool for data description with nonlinear structure [30–33]. Nonetheless, there are still some limitations for the application of GMM. First, it is usually very difficult to determine the approximate number of Gaussian components. If the number of components is too small, it cannot provide a satisfactory approximation for the real dataset. Otherwise, too many components will result in over-fitting problems. Second, to meet the predefined approximation accuracy, a large number of components may be required for GMM. This can largely increase the computing burden and model complexity. Moreover, it may cost a lot of time to train the model when the number of Gaussian components is very large.

To alleviate the aforementioned problems, a novel weighted Gaussian model (WGM) is proposed for nonlinear data description in this paper, which is conceptually simple to understand and quite easy to implement. This method has been motivated by locally weighted learning (LWL) [34,35]. LWL is based on the idea that the complicated nonlinear surface can be approximated by nearby points using distance weighted regression. In the proposed WGM, the distances will first be calculated between the historical samples and the query sample when the output prediction is required for the query sample. Then, the weights for historical samples can be computed according to the calculated distances. Usually, the weight is a decreasing function with the distance since distant points have less relevance with the query while close ones should occupy more relevance. After that, a weighted Gaussian model is built around the query sample that can capture the local data structure around the query sample with great accuracy. This is achieved by designing a weighted log-likelihood function for the joint input and output dataset. By maximizing the weighted log-likelihood function, a local Gaussian model is trained to locally fit the complex surface. At last, by taking the conditional density function of outputs given the inputs, the output distribution for the query data can be estimated. In this paper, this Gaussian density-based approach is referred to as weighted Gaussian regression (WGR).

The remaining parts of this paper are organized as follows. In Section 2, preliminaries about Gaussian model and mixture Gaussian model are simply revisited. Then, the weighted Gaussian model and weighted Gaussian regression are introduced in detail in Section 3. After that, two case studies are carried out to validate the effectiveness and flexibility of the proposed method in Section 4. At last, Section 5 provides some conclusions.

2. Preliminaries

Gaussian density distribution, or Gaussian model, is the most widely used method for data structure description. Assume the input vector is $x \in R^n$ and the output vector is $y \in R^m$, where n and m are the dimensions of input and output variables, respectively. The joint vector of input and output is denoted as $z = [x^T, y^T]^T$, where $z \in R^{n+m}$. For simplicity, we denote $d = n + m$. The observed historical data samples are $(X, Y) = \{(x_i, y_i)\}_{i=1,2,\ldots,N}$. Correspondingly, the joint vector data are $Z = \{z_i\}$, $_{i=1,2,\ldots,N}$. If Gaussian model is used to represent the joint data distribution, it is expressed with the following formula

$$P(z|\theta) = \frac{1}{\sqrt{(2\pi)^d |\Sigma|}} \exp\left\{-\frac{1}{2}(z-\mu)^T \Sigma^{-1}(z-\mu)\right\} \quad (1)$$

where μ is the mean vector and Σ is the covariance matrix, and $\theta = \{\mu, \Sigma\}$ are the parameters to determine a Gaussian distribution. To obtain the parameters of the Gaussian model, one can simply maximize the following log-likelihood function of observed data.

$$L(Z, \theta) = \sum_{i=1}^{N} \ln P(z_i|\theta) \quad (2)$$

However, Gaussian model is not sufficient to describe nonlinear data distribution. In this way, Gaussian mixture model is exploited to better characterize data structure. GMM is constructed by simple linear superposition of Gaussian components, which is expressed as

$$P(z|\Omega) = \sum_{k=1}^{K} \lambda_k P(z|\theta_k) \quad (3)$$

where K is the number of Gaussian components in GMM. λ_k is the probabilistic ratio of the kth Gaussian component and subjects to condition of $\lambda_k \geq 0, \sum_{k=1}^{K} \lambda_k = 1$. $\theta_k = \{\mu_k, \Sigma_k\}$ represent the parameters that define the kth Gaussian component, which are the mean vector μ_k and the covariance matrix $\sum_k$. Then the total parameters in the complete GMM with K components can be defined as $\Omega = \{\{\omega_1, \mu_1, \sum_1\}, \{\omega_2, \mu_2, \sum_2\}, \cdots \{\omega_K, \mu_K, \sum_K\}\}$, which involve both the Gaussian model parameters θ_k and the mixing probabilities ω_k $(1 \leq k \leq K)$. The parameter set of GMM can be estimated by maximizing the following log-likelihood function using the expectation-maximization algorithm

$$L(Z, \Omega) = \sum_{n=1}^{N} \log\left(\sum_{k=1}^{K} \omega_k f(z_n|\theta_k)\right) \quad (4)$$