Journal of Southeast University (English Edition) Vol. 31, No. 1, pp. 80 - 86 Mar. 2015 ISSN 1003—7985

# Quality prediction of batch process using the global-local discriminant analysis based Gaussian process regression model

Lu Chunhong Gu Xiaofeng

(Key Laboratory of Advanced Process Control for Light Industry of Ministry of Education, Jiangnan University, Wuxi 214122, China)

**Abstract**: The conventional single model strategy may be ill-suited due to the multiplicity of operation phases and system uncertainty. A novel global-local discriminant analysis (GLDA) based Gaussian process regression (GPR) approach is developed for the quality prediction of nonlinear and multiphase batch processes. After the collected data is preprocessed through batchwise unfolding, the hidden Markov model (HMM) is applied to identify different operation phases. A GLDA algorithm is also presented to extract the appropriate process variables highly correlated with the quality variables, decreasing the complexity of modeling. Besides, the multiple local GPR models are built in the reduced-dimensional space for all the identified operation phases. Furthermore, the HMM-based state estimation is used to classify each measurement sample of a test batch into a corresponding phase with the maximal likelihood estimation. Therefore, the local GPR model with respect to specific phase is selected for online prediction. The effectiveness of the proposed prediction approach is demonstrated through the multiphase penicillin fermentation process. The comparison results show that the proposed GLDA-GPR approach is superior to the regular GPR model and the GPR based on HMM (HMM-GPR) model.

**Key words**: quality prediction; global-local discriminant analysis; Gaussian process regression; hidden Markov model; soft sensor

**doi**: 10.3969/j.issn.1003-7985.2015.01.014

Recently, soft sensor technique is a promising solution towards online quality prediction[1-2]. The collected historical data which involve high-dimensional, highly correlated and redundant variables lead to a difficulty in building the prediction model[3]. A global-local discriminant analysis (GLDA) is proposed for the Gaussian process regression (GPR) in the nonlinear projection space. Its goal is to find a set of the most important variables which integrate the global and local discriminant information by maximizing the ratio of distances between samples with large differences in output values and those with small differences in output values for dimension reduction. In addition, the batch processes encounter multiple operating phases, which are probably driven by feedstock, production rate, temperature and pressure. Such characteristics of batch processes make the single regression model ill-suited and result in unreliable quality prediction. The hidden Markov model (HMM) is a good candidate for batch processes with system uncertainty due to its strong stochastic and inferential characteristics[4]. Therefore, the advantages of the HMM for phase identification are combined with the merits of GLDA for dimensionality reduction and GPR for quality prediction.

**Received** 2014-06-04.

**Biographies**: Lu Chunhong (1982—), female, graduate; Gu Xiaofeng (corresponding author), male, doctor, professor, xgu@jiangnan.edu.cn.

**Foundation items**: The Fundamental Research Funds for the Central Universities (No. JUDCF12027, JUSRP51323B), the Scientific Innovation Research of College Graduates in Jiangsu Province (No. CXLX12_0734).

**Citation**: Lu Chunhong, Gu Xiaofeng. Quality prediction of batch process using the global-local discriminant analysis based Gaussian process regression model[J]. Journal of Southeast University (English Edition), 2015, 31(1): 80 - 86. [doi: 10.3969/j.issn.1003-7985.2015.01.014]

## 1 Preliminaries

### 1.1 Multiway hidden Markov model

The HMM is an extension of Markov chains that generates a sequential observation depending on the underlying process state. In the HMM, the actual sequence of states is not directly observable but is hidden from the output measurements. The complete specification of an HMM model includes the following key elements: 1) Hidden states, $\boldsymbol{S}=\{S_1, S_2, \cdots, S_N\}$, where $N$ is the number of states; 2) Observable measurement variables at each state, $\boldsymbol{O}=\{O_1, O_2, \cdots, O_M\}$, where $M$ is the number of observation variables per state; 3) State transition probability distribution, $\boldsymbol{A}=\{a_{ij}\}$ is an $N\times N$ state transition probability matrix, where $a_{ij}=P(q_{t+1}=S_j \mid q_t=S_i)$ and $q_t$ represents the hidden state at time $t$; 4) Observation probability distribution, $\boldsymbol{B}=\{b_j(k)\}$, where $b_j(k)=P(o_{k,t} \mid q_t=S_j)$, $1\leqslant j\leqslant N$, $1\leqslant k\leqslant M$ and $o_{k,t}$ denotes the value of the $k$-th observation variable at sampling time $t$; 5) Probability distribution, $\pi=\{\pi_i\}$, where $\pi_i=P(q_{t=1}=S_i)$, $1\leqslant i\leqslant N$.

Thus, the HMM model parameters $H=f(\boldsymbol{A}, \boldsymbol{B}, \pi)$ are calculated from the following measurement sequence:

$$\boldsymbol{O}=\{\boldsymbol{O}_1,\boldsymbol{O}_2,\cdots,\boldsymbol{O}_T\}=\{\{o_{1,1},o_{2,1},\cdots,o_{M,1}\},\{o_{1,2},o_{2,2},\cdots,o_{M,2}\},\cdots,\{o_{1,T},o_{2,T},\cdots,o_{M,T}\}\} \quad (1)$$

For batch processes with multiple operation phases, the HMM is applied to estimate the probabilities of measurement outputs at the invisible states. The multiphase

process is assumed to follow Markov property that the current state only depends on the previous state instead of the older ones.

Consider a state sequence $\boldsymbol{Q} = \{q_1, q_2, \cdots, q_t, \cdots, q_T\}$ as the operation phases. At any sampling time, the value of state variable $q_t$ can be one of the $N$ different hidden states as $q_t \in \{S_i, 1 \leqslant i \leqslant N\}$. A preprocessing step of data unfolding prior to further analysis is required due to the collected historical data in the form of a three-way observation matrix $\boldsymbol{O} = \{o_{i,j}^{(l)}\}$, where $1 \leqslant i \leqslant M$ and $M$ denotes the number of measurement variables; $1 \leqslant j \leqslant T$ and $T$ represents the number of sampling instants; $1 \leqslant l \leqslant L$ and $L$ corresponds to the number of batches. This three-way matrix in the $l$-th batch is rearranged in the format of sequences as

$$\underbrace{\{o_{1,1}^{(l)}, o_{2,1}^{(l)}, \cdots, o_{M,1}^{(l)}\}}_{t=1}, \underbrace{\{o_{1,2}^{(l)}, o_{2,2}^{(l)}, \cdots, o_{M,2}^{(l)}\}}_{t=2}, \cdots, \underbrace{\{o_{1,T}^{(l)}, o_{2,T}^{(l)}, \cdots, o_{M,T}^{(l)}\}}_{t=T} \tag{2}$$

Here, the sequence can be considered as $T$ sequential blocks. Then the two-way data matrix through batchwise unfolding can be given as

$$\widetilde{\boldsymbol{O}} = \begin{bmatrix} O^{(l=1)} \\ O^{(l=2)} \\ \vdots \\ O^{(l=L)} \end{bmatrix} = \left[\begin{array}{ccc|ccc|c|ccc} o_{1,1}^{(1)} & \cdots & o_{M,1}^{(1)} & o_{1,2}^{(1)} & \cdots & o_{M,2}^{(1)} & \cdots & o_{1,T}^{(1)} & \cdots & o_{M,T}^{(1)} \\ o_{1,1}^{(2)} & \cdots & o_{M,1}^{(2)} & o_{1,2}^{(2)} & \cdots & o_{M,2}^{(2)} & \cdots & o_{1,T}^{(2)} & \cdots & o_{M,T}^{(2)} \\ \vdots & & \vdots & \vdots & & \vdots & & \vdots & & \vdots \\ o_{1,1}^{(L)} & \cdots & o_{M,1}^{(L)} & o_{1,2}^{(L)} & \cdots & o_{M,2}^{(L)} & \cdots & o_{1,T}^{(L)} & \cdots & o_{M,T}^{(L)} \end{array}\right] \tag{3}$$

The corresponding state matrix is expressed as

$$\boldsymbol{Q} = \begin{bmatrix} q_1^{(l=1)} & q_2^{(l=1)} & \cdots & q_T^{(l=1)} \\ q_1^{(l=2)} & q_2^{(l=2)} & \cdots & q_T^{(l=2)} \\ \vdots & \vdots & & \vdots \\ q_1^{(l=L)} & q_2^{(l=L)} & \cdots & q_T^{(l=L)} \end{bmatrix} \tag{4}$$

With the observation sequences $\widetilde{\boldsymbol{O}}$ and corresponding state sequences $\boldsymbol{Q}$ in the unfolded training data set, the HMM can be first trained to estimate different operation phases by the Baum-Welch algorithm. Given the observed measurements $\widetilde{\boldsymbol{O}}$, the corresponding optimum state sequences $\boldsymbol{Q}^* = \{q_1, q_2, \cdots, q_T\}$ can be solved by using the Vierbi algorithm as follows[5]:

$$\boldsymbol{Q}^* = \arg\max_{Q} P(\boldsymbol{Q} \mid \widetilde{\boldsymbol{O}}, H) \tag{5}$$

Furthermore, for any test batch $\boldsymbol{O}^{(\mathrm{te})}$, its observation sequence is rearranged as

$$\widetilde{\boldsymbol{O}}^{(\mathrm{te})} = \{\{o_{1,1}^{(\mathrm{te})}, o_{2,1}^{(\mathrm{te})}, \cdots, o_{M,1}^{(\mathrm{te})}\}, \{o_{1,2}^{(\mathrm{te})}, o_{2,2}^{(\mathrm{te})}, \cdots, o_{M,2}^{(\mathrm{te})}\}, \cdots, \{o_{1,T}^{(\mathrm{te})}, o_{2,T}^{(\mathrm{te})}, \cdots, o_{M,T}^{(\mathrm{te})}\}\} \tag{6}$$

It can be categorized into its underlying operation phase by maximizing its likelihood,

$$i^* = \arg\max_{1 \leqslant i \leqslant N} P(\widetilde{\boldsymbol{O}}^{(\mathrm{te})} \mid S_i) \tag{7}$$

### 1.2 Gaussian process regression model

Available historical data are presented as matrices $\boldsymbol{X} \in \mathbf{R}^{d \times n}$ and $\boldsymbol{Y} \in \mathbf{R}^{1 \times n}$, where $\boldsymbol{X}$ contains $n$ input samples with $d$ easy-to-measure variables, and $\boldsymbol{Y}$ contains $n$ output samples with difficult-to-measure variables. The aim of the data-driven soft sensor is to obtain the process model between $\boldsymbol{X}$ and $\boldsymbol{Y}$. A Gaussian process is a collection of random variables[6], any finite number of which follows a joint Gaussian distribution[7]. In a linear Gaussian process regression model, the output variable is assumed to be a linear combination of input variables and an additive noise.

$$y_i = f(\boldsymbol{x}_i) + \varepsilon_i = \boldsymbol{x}_i^{\mathrm{T}} \boldsymbol{\beta} + \varepsilon_i \tag{8}$$

where $x_i$ is a measurement of input variable; $y_i$ is the output variable value; $f$ is the regression function; $\varepsilon_i$ is the noise following a Gaussian distribution with zero mean and variance $\sigma_\varepsilon^2$; and $\boldsymbol{\beta}$ is the regression coefficient vector and follows a Gaussian distribution with zero mean and covariance of $\boldsymbol{\Sigma}_\beta$, $\boldsymbol{\beta} \sim N(0, \boldsymbol{\Sigma}_\beta)$.

A Gaussian process is completely determined by its mean $m(\boldsymbol{x})$ and covariance function $\boldsymbol{\Sigma}(\boldsymbol{x}_i, \boldsymbol{x}_j)$, where $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ represent the two input measurements. Given the process inputs and regression model parameters , then the conditional probability density function of the output variables is calculated as

$$P(\boldsymbol{Y} \mid \boldsymbol{X}, \boldsymbol{\beta}) = \prod_{i=1}^{n} P(y_i \mid \boldsymbol{x}_i, \boldsymbol{\beta}) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi}\sigma_\varepsilon} \exp\left(-\frac{(y_i - \boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{\beta})^2}{2\sigma_\varepsilon^2}\right) \sim N(\boldsymbol{X}^{\mathrm{T}}\boldsymbol{\beta}, \sigma_\varepsilon^2 \boldsymbol{I}) \tag{9}$$

where $\boldsymbol{I}$ is the $n \times n$ identity matrix. In the Bayesian estimation, the posterior probability of $\boldsymbol{\beta}$ is given by

$$P(\boldsymbol{\beta} \mid \boldsymbol{Y}, \boldsymbol{X}) = \frac{P(\boldsymbol{Y} \mid \boldsymbol{X}, \boldsymbol{\beta}) P(\boldsymbol{\beta})}{P(\boldsymbol{Y} \mid \boldsymbol{X})} = \frac{P(\boldsymbol{Y} \mid \boldsymbol{X}, \boldsymbol{\beta}) P(\boldsymbol{\beta})}{\int P(\boldsymbol{Y} \mid \boldsymbol{X}, \boldsymbol{\beta}) P(\boldsymbol{\beta}) \mathrm{d}\boldsymbol{\beta}} \tag{10}$$

According to Ref. [6], the mean vector of regression coefficients is expressed as

$$\bar{\boldsymbol{\beta}} = \sigma_\varepsilon^{-2} (\sigma_\varepsilon^{-2} \boldsymbol{X}\boldsymbol{X}^{\mathrm{T}} + \boldsymbol{\Sigma}_\beta^{-1})^{-1} \boldsymbol{X}\boldsymbol{Y} = \sigma_\varepsilon^{-2} \boldsymbol{\Sigma}_{\mathrm{h}}^{-1} \boldsymbol{X}\boldsymbol{Y} \tag{11}$$

with $\boldsymbol{\Sigma}_{\mathrm{h}} = \sigma_\varepsilon^{-2} \boldsymbol{X}\boldsymbol{X}^{\mathrm{T}} + \boldsymbol{\Sigma}_\beta^{-1}$. The posterior probability of $\boldsymbol{\beta}$ can be given by $P(\boldsymbol{\beta} \mid \boldsymbol{Y}, \boldsymbol{X}) \sim N(\bar{\boldsymbol{\beta}}, \boldsymbol{\Sigma}_{\mathrm{h}}^{-1})$. The predictive distribution can be written for any test sample $\boldsymbol{x}_{\mathrm{t}}$ and corresponding output $y_{\mathrm{t}}$ as

$$P(y_{\mathrm{t}} \mid \boldsymbol{x}_{\mathrm{t}}, \boldsymbol{X}, \boldsymbol{Y}) \sim N(\sigma_\varepsilon^{-2} \boldsymbol{x}_{\mathrm{t}}^{\mathrm{T}} \boldsymbol{\Sigma}_{\mathrm{h}}^{-1} \boldsymbol{X}\boldsymbol{Y}, \boldsymbol{x}_{\mathrm{t}}^{\mathrm{T}} \boldsymbol{\Sigma}_{\mathrm{h}}^{-1} \boldsymbol{x}_{\mathrm{t}}) \tag{12}$$

Then the output prediction is obtained as

$$\hat{y}_{\mathrm{t}} = \sigma_\varepsilon^{-2} \boldsymbol{x}_{\mathrm{t}}^{\mathrm{T}} \boldsymbol{\Sigma}_{\mathrm{h}}^{-1} \boldsymbol{X}\boldsymbol{Y} \tag{13}$$

The nonlinear version of GPR is adopted to enhance the capability of dealing with inherent nonlinearity in this study. With $\phi(\cdot)$ denoting the nonlinear mapping function, Eq. (12) is rewritten as

$$P(y_t \mid \boldsymbol{x}_t, \boldsymbol{X}, \boldsymbol{Y}) \sim N(\sigma_e^{-2}\phi(\boldsymbol{x}_t)^T \boldsymbol{\Sigma}_h^{-1}\phi(\boldsymbol{X})\boldsymbol{Y}, \phi(\boldsymbol{x}_t)^T \boldsymbol{\Sigma}_h^{-1}\phi(\boldsymbol{x}_t)) \quad (14)$$

A kernel function is defined as

$$\kappa(\boldsymbol{x}, \boldsymbol{x}') = \phi(\boldsymbol{x})^T \boldsymbol{\Sigma}_\beta \phi(\boldsymbol{x}')$$

Substituting $\phi(\cdot)$ function with this kernel function, Eq. (14) becomes

$$P(y_t \mid \boldsymbol{x}_t, \boldsymbol{X}, \boldsymbol{Y}) \sim N\{\kappa(\boldsymbol{x}_t, \boldsymbol{X})[\kappa(\boldsymbol{X}, \boldsymbol{X}) + \sigma_e^2 \boldsymbol{I}]^{-1}\boldsymbol{Y}, \kappa(\boldsymbol{x}_t, \boldsymbol{x}_t) - \kappa(\boldsymbol{x}_t, \boldsymbol{X})[\kappa(\boldsymbol{X}, \boldsymbol{X}) + \sigma_e^2 \boldsymbol{I}]^{-1}\kappa(\boldsymbol{X}, \boldsymbol{x}_t)\} \quad (15)$$

with the output prediction $\hat{y}_t = \kappa(\boldsymbol{x}_t, \boldsymbol{X})[\kappa(\boldsymbol{X}, \boldsymbol{X}) + \sigma_e^2 \boldsymbol{I}]^{-1}\boldsymbol{Y}$.

## 2 Proposed Approach

### 2.1 Global-local discriminant analysis (GLDA)

The aim of the soft sensor based on the collected historical data is to build a mathematical model approximating the unknown functional dependence between process variables and the key quality variable. Historical data often contains a relatively large number of measurements of process variables. However, some of variables are unimportant and include little information about the output variable, which increases the model complexity. The useful variables that relate to the quality variable should be extracted before the soft sensor development[6]. Therefore, global-local discriminant analysis (GLDA) is presented for dimensionality reduction. By maximizing the ratio of distances of samples with large differences in the output value and those with small differences in the output value, the GLDA algorithm integrates global and local information to extract the most significant variables and remove the irrelevant or redundant variables. The presented algorithm is based on linear discriminant analysis for regression (LDAr)[8-9].

The principle of the LDAr algorithm is briefly described first. Given a set of input/output pairs $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^n, \boldsymbol{x}_i \in \mathbf{R}^{d\times 1}$, $y_i \in \mathbf{R}^{c\times 1}$ and $n$ denotes the number of the pairs. Here, $d$ is the number of original variables and $c$ is the dimension of the output vector which is often assumed to be 1. The within-class and between-class scatter matrices for LDAr are written as

$$\boldsymbol{S}_w = \frac{1}{n_{w(i,j)\in A_w}} \sum f_w(y_i, y_j)(\boldsymbol{x}_i - \boldsymbol{x}_j)(\boldsymbol{x}_i - \boldsymbol{x}_j)^T$$

$$\boldsymbol{S}_b = \frac{1}{n_{b(i,j)\in A_b}} \sum f_b(y_i, y_j)(\boldsymbol{x}_i - \boldsymbol{x}_j)(\boldsymbol{x}_i - \boldsymbol{x}_j)^T$$

where $A_w = \{(i,j) \mid |y_i - y_j| - \tau < 0, i < j\}$ is the membership set of nearby output pairs representing small differences; $A_b = \{(i,j) \mid |y_i - y_j| - \tau \geqslant 0, i < j\}$ is the membership set of faraway output pairs representing large differences; $n_w$ and $n_b$ are the cardinality of $A_w$ and $A_b$, respectively; the threshold parameter $\tau = e\sigma_y$, $\sigma_y$ representing the standard deviation of $y$, $e$ representing multiple factor ranged in $[0.1, 1.0]$. The function $f_w$ and $f_b$ are non-negative weight functions, defined as $f_w(x) = f_b(x) = ||x| - \tau|$. Although the extracted leading components by the LDAr algorithm show good performance[8-9], they can only capture the local data information. One critical drawback of the LDAr method is that it does not explicitly consider the global data information.

In this paper, the global weight matrix is defined as

$$\boldsymbol{S}_g = \frac{1}{n_b + n_{w(i,j)\in(A_b \cup A_w)}} \sum f_g(y_i - y_j)(\boldsymbol{x}_i - \boldsymbol{x}_j)(\boldsymbol{x}_i - \boldsymbol{x}_j)^T \quad (16)$$

where the function is defined as $f_g(x) = |x|$.

Likewise, the nonlinear mapping $\phi(\cdot)$ is adopted to transform input data in a $d$-dimensional space into an $r$-dimensional feature space in handling the nonlinear processes. Suppose that the mapping is centered, i.e., $\sum_{i=1}^n \phi(x_i) = 0$. The corresponding within-class scatter matrix, between-class scatter matrix and global weight matrix in the feature space are rewritten as follows:

$$\tilde{\boldsymbol{S}}_w = \frac{1}{n_{w(i,j)\in A_w}} \sum f_w(y_i, y_j)[\phi(\boldsymbol{x}_i) - \phi(\boldsymbol{x}_j)][\phi(\boldsymbol{x}_i) - \phi(\boldsymbol{x}_j)]^T = \frac{1}{n_w} \sum_{i=1}^n \sum_{j=1}^n [\phi(\boldsymbol{x}_i) - \phi(\boldsymbol{x}_j)] w_{ij} [\phi(\boldsymbol{x}_i)^T - \phi(\boldsymbol{x}_j)^T] = \frac{2}{n_w}\phi(\boldsymbol{X})(\boldsymbol{D}_w - \boldsymbol{W}_w)\phi(\boldsymbol{X})^T = \frac{2}{n_w}\phi(\boldsymbol{X})\boldsymbol{L}_w\phi(\boldsymbol{X})^T \quad (17)$$

where $\boldsymbol{X} = [\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n] \in \mathbf{R}^{d\times n}$ is an input matrix, $\phi(\boldsymbol{X}) = [\phi(\boldsymbol{x}_1), \cdots, \phi(\boldsymbol{x}_n)] \in \mathbf{R}^{r\times n}$ is a feature matrix, $\boldsymbol{W}_w = [w_{ij}] \in \mathbf{R}^{n\times n}$ is a symmetric matrix whose element is

$$w_{ij} = \begin{cases} f_w(y_i - y_j) & \text{if } (i,j) \in A_w \\ 0 & \text{otherwise} \end{cases}$$

$\boldsymbol{D}_w$ is a diagonal matrix whose diagonal elements are the column sum of $w_{ij}$, i.e., $D_{ii}^w = \sum_j w_{ij}$. $\boldsymbol{L}_w = \boldsymbol{D}_w - \boldsymbol{W}_w$, is called the local similarity matrix.

$$\tilde{\boldsymbol{S}}_b = \frac{2}{n_b}\phi(\boldsymbol{X})(\boldsymbol{D}_b - \boldsymbol{W}_b)\phi(\boldsymbol{X})^T = \frac{2}{n_b}\phi(\boldsymbol{X})\boldsymbol{L}_b\phi(\boldsymbol{X})^T \quad (18)$$

where $\boldsymbol{W}_b = [b_{ij}] \in \mathbf{R}^{n\times n}$ is a symmetric matrix whose element is

$$b_{ij} = \begin{cases} f_b(y_i - y_j) & \text{if } (i,j) \in A_b \\ 0 & \text{otherwise} \end{cases}$$

$\boldsymbol{D}_b$ is a diagonal matrix whose diagonal elements are the column sum of $b_{ij}$; i.e., $D_{ii}^b = \sum_j b_{ij}$. $\boldsymbol{L}_b = \boldsymbol{D}_b - \boldsymbol{W}_b$ is called the local dissimilarity matrix.

$$\tilde{\boldsymbol{S}}_g = \frac{2}{n_b + n_w}\phi(\boldsymbol{X})(\boldsymbol{D}_g - \boldsymbol{W}_g)\phi(\boldsymbol{X})^T \quad (19)$$

where $\boldsymbol{W}_g = [g_{ij}] \in \mathbf{R}^{n\times n}$ is a weighting matrix whose element is $g_{ij} = f_g(y_i - y_j)$; $\boldsymbol{D}_g$ is a diagonal matrix whose diagonal elements are the column sum of $g_{ij}$; i. e., $D_{ii}^g = \sum_j g_{ij}$. $\boldsymbol{L}_g = \boldsymbol{D}_g - \boldsymbol{W}_g$ is called the global dissimilarity matrix.

For the need of regression, a motivation of dimensional reduction is to find a projection that makes the samples with large differences between output values become more dissimilar, and simultaneously makes the samples with small differences between output values become more similar. Therefore, a good projection means that it can maximize the local and global dissimilar matrices and minimize the local similar matrix at the same time. If the nonlinear transformation can satisfy Eqs. (17), (18) and (19) simultaneously, the most important variables highly related to output variable will be extracted. Thus, the projection matrix can be found by maximizing the following objective function

$$\boldsymbol{\alpha} = \arg\max_{\boldsymbol{\alpha}} \frac{\boldsymbol{\alpha}^{\mathrm{T}}(\tilde{\boldsymbol{S}}_b + \tilde{\boldsymbol{S}}_p)\boldsymbol{\alpha}}{\boldsymbol{\alpha}^{\mathrm{T}}\tilde{\boldsymbol{S}}_w\boldsymbol{\alpha}} \tag{20}$$

It is difficult to solve directly the GED problem of Eq. (20). There are coefficients $\theta_i$ ($i = 1, 2, \cdots, n$) such that

$$\boldsymbol{\alpha} = \sum_{i=1}^{n} \theta_i \phi(\boldsymbol{x}_i) = \phi(\boldsymbol{X})\boldsymbol{\theta} \tag{21}$$

We define an $n \times n$ matrix $\boldsymbol{K}$ by $\boldsymbol{K} = \phi(\boldsymbol{X})^{\mathrm{T}}\phi(\boldsymbol{X})$, whose element is $k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \phi(\boldsymbol{x}_i)^{\mathrm{T}}\phi(\boldsymbol{x}_j)$. Centralize the kernel matrix $\boldsymbol{K} = \boldsymbol{K} - \mathbf{1}_n\boldsymbol{K} - \boldsymbol{K}\mathbf{1}_n + \mathbf{1}_n\boldsymbol{K}\mathbf{1}_n$, where all the elements of $\mathbf{1}_n$ is $1/n$. Substituting Eq. (21) into Eq. (20), the numerator and denominator of Eq. (20) are

$$\begin{aligned}\boldsymbol{\alpha}^{\mathrm{T}}(\tilde{\boldsymbol{S}}_b + \tilde{\boldsymbol{S}}_p)\boldsymbol{\alpha} &= \frac{2}{n_b}\boldsymbol{\theta}^{\mathrm{T}}\phi(\boldsymbol{X})^{\mathrm{T}}\phi(\boldsymbol{X})\boldsymbol{L}_b\phi(\boldsymbol{X})^{\mathrm{T}}\phi(\boldsymbol{X})\boldsymbol{\theta} + \\ &\quad \frac{2}{n_b + n_w}\boldsymbol{\theta}^{\mathrm{T}}\phi(\boldsymbol{X})^{\mathrm{T}}\phi(\boldsymbol{X})\boldsymbol{L}_g\phi(\boldsymbol{X})^{\mathrm{T}}\phi(\boldsymbol{X})\boldsymbol{\theta} = \\ &\quad \frac{2}{n_b}\boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{K}\boldsymbol{L}_b\boldsymbol{K}\boldsymbol{\theta} + \frac{2}{n_b + n_w}\boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{K}\boldsymbol{L}_g\boldsymbol{K}\boldsymbol{\theta}\end{aligned} \tag{21}$$

$$\boldsymbol{\alpha}^{\mathrm{T}}\tilde{\boldsymbol{S}}_w\boldsymbol{\alpha} = \frac{2}{n_w}\boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{K}\boldsymbol{L}_w\boldsymbol{K}\boldsymbol{\theta} \tag{22}$$

The objective function of the GLDA approach can be rewritten as

$$\boldsymbol{\theta} = \arg\max_{\boldsymbol{\theta}} \frac{\boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{K}(\boldsymbol{L}_b + \boldsymbol{L}_g)\boldsymbol{K}\boldsymbol{\theta}}{\boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{K}\boldsymbol{L}_w\boldsymbol{K}\boldsymbol{\theta}} \tag{23}$$

When the denominator of Eq. (23) is nonsingular, the projection vector $\boldsymbol{\theta}$ is given by solving GED problem:

$$\boldsymbol{K}(\boldsymbol{L}_b + \boldsymbol{L}_g)\boldsymbol{K}\theta_m = \lambda_m\boldsymbol{K}\boldsymbol{L}_w\boldsymbol{K} \qquad \lambda_1 \geqslant \lambda_2 \geqslant \cdots \geqslant \lambda_r \tag{24}$$

By the projection matrix $\boldsymbol{P} = [\boldsymbol{\theta}_1, \cdots, \boldsymbol{\theta}_r] \in \mathbf{R}^{n\times r}$, the nonlinear feature vector $\boldsymbol{z}$ of a new sample $\boldsymbol{x}$ is obtained, $\boldsymbol{z} = \boldsymbol{P}^{\mathrm{T}}k(\boldsymbol{X}, \boldsymbol{x})$. The matrix $\boldsymbol{K}\boldsymbol{L}_w\boldsymbol{K}$ is sometimes singular for predicting problems of batch process. As an effective preprocessing method, PCA is first applied to project the dataset into the subspace so that the matrix $\boldsymbol{K}\boldsymbol{L}_w\boldsymbol{K}$ is nonsingular.

### 2.2 GLDA-based GPR model (GLDA-GPR) for online quality prediction

In this work, the Gaussian kernel function is adopted. The model parameters are tuned by the cross-validation procedure. The detailed procedure is summarized below:

1) Collect the input and output data of the batch process.

2) Conduct batchwise unfolding on the multiway data and scale the formed two-dimension matrix with zero mean and unit variance along all the process variables at each sampling instant.

3) Use the unfolded input data in the training set to estimate the HMM through the Baum-Welch algorithm and identify all different operating phases.

4) Compute the three matrices, global dissimilarity matrix, local similarity and dissimilarity matrices, respectively.

5) Specify the kernel function and then compute the kernel matrix and normalize it.

6) Conduct generalized eigenvalue decomposition to obtain the eigenvalues and corresponding eigenvectors. Form a reduced-dimensional discriminant subspace $P_r$.

7) Build the multiple local GPR models in the subspace using each pair of input and output block matrices.

8) Unfold each new measurement sample in the test batches and normalize them. Then obtain the corresponding nonlinear feature vector of the test sample. Furthermore, identify the operation phase to which the batch belongs through the Viterbi algorithm with the maximized likelihood estimation.

9) Estimate the quality variable value of each test sample using the local GPR model with respect to the identified phase.

## 3 Case study

### 3.1 Fed-batch penicillin fermentation process

A simulated fed-batch penicillin fermentation process[10] is adopted to demonstrate the online measurement capability of the GLDA-GPR approach. The accurate and reliable online estimation of the quality variable can provide real-time feedback of the model-based advanced control design in batch processes[11]. The evaluating results are compared to the regular GPR model and the HMM-GPR model. The appropriate kernel parameters are selected to avoid the models overfitting or underfitting.

In the batch process simulation, the microorganisms are grown during the initial 40 h until the cell density is maximized. Then the fermenter process is switched to the fed-batch mode in order to boost the synthesis of penicillin.

The entire process for each batch has the duration of 400 h with a sampling interval of 0.5 h. Two cascade controllers are implemented in the process to track pH and temperature set points. The substrate of glucose and oxygen are continuously fed into the fermenter for cell growth and penicillin formation. The process exhibits nonlinear dynamics, multiphase feature and system uncertainty. In this paper, ten measurement variables including the aeration rate, agitator power, $CO_2$ concentration, substrate feed temperature, substrate concentration, pH, dissolved oxygen concentration, fermenter temperature, cooling water flow rate and generated heat are used as input variables, while three quality variables including penicillin concentration $O_1$, biomass concentration $O_2$ and substrate feed rate $O_3$ are selected as outputs. The initial and operational parameter settings of the process operating conditions are given in Tab. 1 and Tab. 2.

**Tab. 1** Initial conditions of operational parameters in the fed-batch penicillin fermentation process

| Variable | Initial condition |
|---|---|
| Substrate concentration/($g \cdot L^{-1}$) | 14 to 16 |
| Dissolved oxygen concentration/($g \cdot L^{-1}$) | 1.05 to 1.25 |
| Biomass concentration/($g \cdot L^{-1}$) | 0.05 to 0.15 |
| Penicillin concentration/($g \cdot L^{-1}$) | 0 |
| Culture volume/L | 99 to 102 |
| $CO_2$ concentration/($g \cdot L^{-1}$) | 0.5 to 0.8 |
| pH | 4.5 to 5.5 |
| Fermenter temperature/K | 297 to 304 |
| Generated heat/kJ | 0 |

**Tab. 2** Set points of operational parameters in the fed-batch penicillin fermentation process

| Variable | Set point |
|---|---|
| Aeration rate/($g \cdot L^{-1}$) | 8 to 9 |
| Agitator power/W | 28 to 32 |
| Substrate feed flow rate/($mL \cdot h^{-1}$) | 35 to 45 |
| Substrate feed temperature/K | 295 to 298 |
| Fermenter temperature/K | 297 to 301 |
| pH | 4.8—5.1 |

A total of 30 batches as training set are collected for building the soft sensor model and the additional 10 batches in test set for predicting quality variable in terms of the accuracy and reliability. The following root-mean-square error (RMSE) and $R^2$ indices are used to evaluate the learning and prediction capability of soft sensor:

$$\mathrm{RMSE}(i) = \sqrt{\frac{\sum_{l=1}^{L}\sum_{j=1}^{T}(\hat{y}_{ijl}^{(\mathrm{te})} - y_{ijl}^{(\mathrm{te})})^2}{TL}} \tag{25}$$

$$R^2(i) = 1 - \frac{\sum_{l=1}^{L}\sum_{j=1}^{T}(\hat{y}_{ijl}^{(\mathrm{te})} - y_{ijl}^{(\mathrm{te})})^2}{\sum_{l=1}^{L}\sum_{j=1}^{T}(\bar{y}_{i}^{(\mathrm{te})} - y_{ijl}^{(\mathrm{te})})^2} \tag{26}$$

where $L$ and $T$ denote the number of batches and sampling instants in the test set; $\bar{y}_i^{(\mathrm{te})}$ denotes the mean value of the $i$-th output variable; $y_{ijl}^{(\mathrm{te})}$ and $\hat{y}_{ijl}^{(\mathrm{te})}$ are the actual and predicted measurements of the the $i$-th output variable for the $l$-th batch and the $j$-th sampling instant, respectively.

### 3.2 Online soft sensor prediction results

First, three inherent operating phases including lag phase, growth phase, and saturation phase are identified using the HMM conducted on the training batches, which coincides with the actual operation shifts. The obtained identification result is given in Fig. 1. The fermentation process begins and continues for 48 h in the first phase. Next, it is shifted to the second phase and remains for 292 h. The production saturation of penicillin corresponds to the last phase.

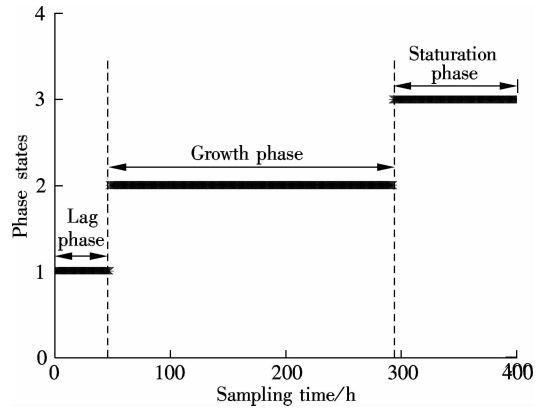


**Fig. 1** Phase identification of the simulated process

Then these training batches are projected onto the reduced-dimensional latent variable space and the extracted directions describe the most significant information about the quality variable among the identified phases. Furthermore, the multiple local GPR models are built in the projection space. Each measurement sample of test batches can be categorized into an individual operating condition so that the local GPR model can be adaptively chosen for the online quality variable estimation. The predicted time-series trends of the three output variables by averaging over the 10 test batches are plotted in Figs. 2 to 4. Meanwhile, the quantitative comparison of different soft sensor modeling and prediction results are listed in Tab. 3. It can be observed that the regular GPR model leads to the worst prediction of three output variables. As shown in Fig. 2, a significant offset from the actual measurement trajectory, especially between the shift phases, denotes that the GPR model has fairly poor prediction of the penicillin concentration. This is because the model cannot effectively capture some of the local switching, and this attributed to multiple operating phases of batch process. These pattern switches across different phases reveal that the regular GPR model is not sensitive to the shifting dynamics within various phases. As a result, the performance signifi-

cantly deteriorates. Similar prediction trends of biomass concentrations and substrate feed rate are exhibited in Figs. 3 and 4. On the other hand, the HMM-GPR method is superior to the regular GPR model in terms of evaluating performance because the HMM model is available to handle the multiple operation phases in the original measurement space. The HMM model can effectively estimate the probabilities of the process operation with respect to different phases in a stochastic fashion so that the local GPR model can obtain more accurate predictions than the regular GPR model. Consequently, the HMM-GPR method can perform well compared to the GPR model.

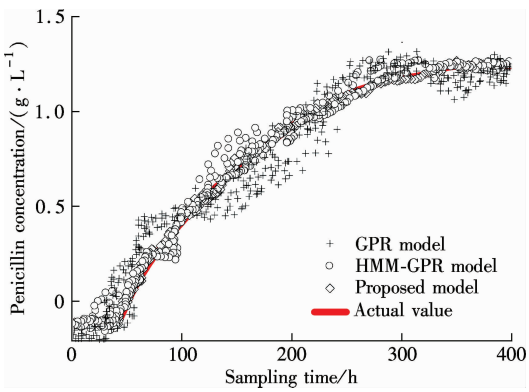


**Fig. 2** Comparison of penicillin concentration predictions by three different methods

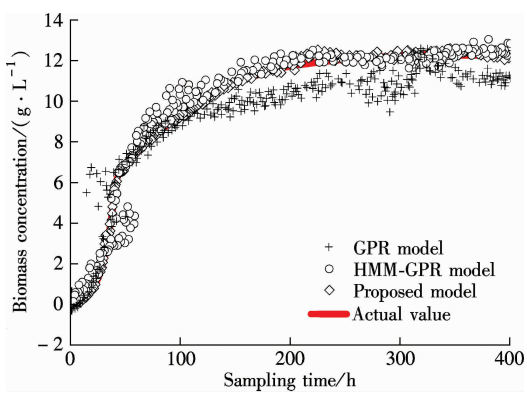


**Fig. 3** Comparison of biomass concentration predictions by three different methods

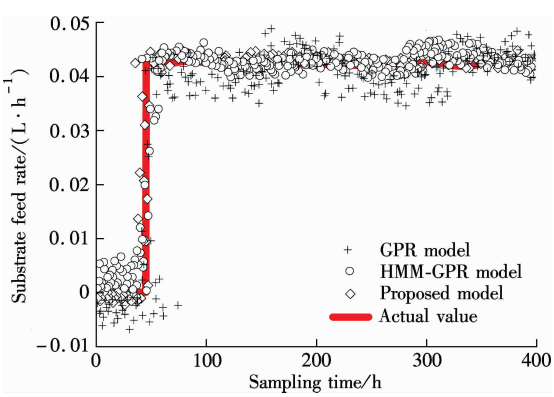


**Fig. 4** Comparison of substrate feed rate predictions by three different methods

**Tab. 3** Comparison of soft sensor modeling and prediction results using regular GPR, HMM-GPR and proposed methods

| Output variable | Method | RMSE | | $R^2$ | |
|---|---|---|---|---|---|
| | | Training | Prediction | Training | Prediction |
| $O_1$ | GPR | 0.127 | 0.158 | 0.892 | 0.853 |
| | HMM-GPR | 0.086 | 0.102 | 0.937 | 0.914 |
| | Proposed | 0.053 | 0.091 | 0.962 | 0.925 |
| $O_2$ | GPR | 0.524 | 0.812 | 0.923 | 0.912 |
| | HMM-GPR | 0.378 | 0.634 | 0.960 | 0.934 |
| | Proposed | 0.097 | 0.185 | 0.974 | 0.961 |
| $O_3$ | GPR | 0.726 | 0.895 | 0.885 | 0.863 |
| | HMM-GPR | 0.582 | 0.694 | 0.906 | 0.892 |
| | Proposed | 0.105 | 0.233 | 0.932 | 0.921 |

Moreover, the composition estimation using the GLDA-GPR approach can not only capture the shifted relationships and track each change in the different operation phases, but also extract the most influential process variables which are highly correlated with the quality variable. Multiple phases have no obvious effect on the performance of the model. A minimal variability across all three variables indicates that the online prediction of the model matches well their actual values. The proposed method provides the most accurate prediction of these three quality variables since it gives the smallest RMSE index values among the three methods. In addition, we examine the uncertainty information of the proposed GLDA-GPR model by the output result of biomass concentration in terms of predictive variance. As shown in Fig. 5, the predictive variance changes greatly in different phases, but slightly inside each phase. In other words, most of samples are in control and do not violate the model since the values of predictive variance are relatively small. Thus, it can be inferred that the presented approach has reliable performance for online prediction. In a word, the proposed approach leads to different local GPR models in the discriminant subspace with respect to various phases and provides a relatively accurate prediction of quality variables in the batch processes.

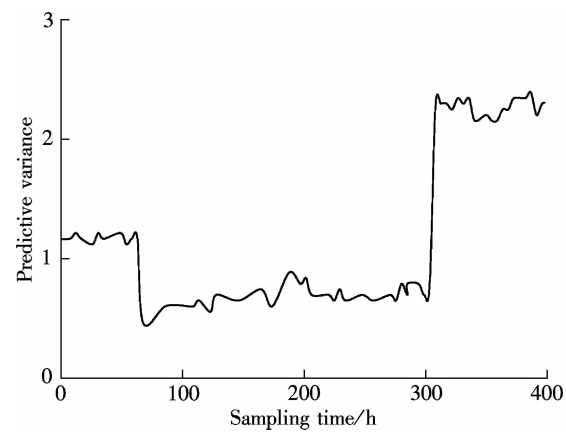


**Fig. 5** Predictive variance of the proposed method for biomass concentration

## 4 Conclusion

A nonlinear probabilistic soft sensor is constructed for the online prediction of the quality variable in the form of the Gaussian distribution. The proposed approach integrates the HMM to automatically segment multiple phases of batch process. Some important process variables are highly related to the quality variable are extracted through the GLDA method. In this way, some crucial information can be retained in the reduced-dimensional subspace, serving the subsequent regression modeling. Moreover, multiple local GPR models are constructed in the obtained space to characterize the shifting process dynamics within each individual phase. The proposed approach is applied to the fed-batch penicillin cultivation process to provide accurate measurements of quality variables. Compared to the regular GPR and HMM-GPR methods, the GLDA-GPR approach has better prediction performance.

## References

[1] Kadlec P, Gabrys B, Strandt S. Data driven soft sensor in the process industry [J]. *Computers and Chemical Engineering*, 2009, **33**(4): 795 - 814.

[2] Lin B, Recke J, Knudsen K H, et al. A systematic approach for soft sensor development [J]. *Computers and Chemical Engineering*, 2007, **31**(5): 419 - 425.

[3] Sliškovic D, Grbic R, Nyarko E K. Data preprocessing in data based process modeling [C]//*Proc of the IFAC International Conference on Intelligent Control Systems and Signal Processing*. Istanbul, Turkey, 2009: 559 - 565.

[4] Chen J H, Hsu T-Y, Chen C-C, et al. Online predictive monitoring using dynamic imaging of furnaces with the combinational method of multiway principal component analysis and hidden Markov model [J]. *Industrial & Engineering Chemistry Research*, 2011, **50**(5): 2946 - 2958.

[5] Viterbi A J. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm [J]. *IEEE Transactions on Information Theory*, 1976, **13**(2): 260 - 269.

[6] Grbic R, Sliškovic D, Kadlec P. Adaptive soft sensor for online prediction and process monitoring based on a mixture of Gaussian process models [J]. *Computers and Chemical Engineering*, 2013, **58**:84 - 97.

[7] Rasmussen C E, Williams C K I. *Gaussian processes in machine learning* [M]. Cambridge, MA, USA: The MIT Press, 2006.

[8] Kwak N. Kernel discriminant analysis for regression problems [J]. *Pattern Recognition*, 2012, **45**(5): 2019 - 2031.

[9] Kwak N, Lee J W. Feature extraction based on subspace methods for regression problems [J]. *Neurocomputers*, 2010, **73**(10/11/12):1740 - 1751.

[10] Birol G, Undey C, Cinar A. A modular simulation package for fed-batch fermentation: penicillin production [J]. *Computers and Chemical Engineering*, 2002, **26**(11): 1553 - 1565.

[11] Alford J S. Bioprocess control: advances and challenges [J]. *Computers and Chemical Engineering*, 2006, **30**(10/11/12):1464 - 1475.

# 基于全局局部鉴别分析的高斯回归模型的间歇过程质量预测

卢春红　顾晓峰

(江南大学轻工过程先进控制教育部重点实验室,无锡 214122)

**摘要**:由于过程操作阶段的复杂性及系统的不确定性使得传统的单模态模型策略为病态,因此提出了一种全局局部鉴别分析(GLDA)的高斯过程回归(GPR)方法用于非线性多阶段暂态过程的质量预测. 首先,将采集数据按批次方向展开,并采用隐马尔科夫模型(HMM)识别不同的操作阶段. 其次,利用 GLDA 算法提取与质量变量高度相关的过程变量,降低建模的复杂度. 在该降维后的子空间,为所有识别出的操作阶段建立多个局部 GPR 模型. 利用 HMM 状态估计将测试批次的每个测量样本以最大似然估计的方式划分到对应的阶段中. 最后,选出与具体阶段相对应的局部 GPR 模型进行在线预测. 利用多阶段的青霉素发酵过程验证了所提预测方法的有效性. 结果表明,与常规的 GPR 模型及基于 HMM 的 GPR 模型相比,提出的 GLDA-GPR 方法更具优势.

**关键词**:质量预测;全局局部鉴别分析;高斯过程回归;隐马尔科夫模型;软测量

**中图分类号**:TP273