Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

Identification of important factors influencing nonlinear counting systems*

Xinmin ZHANG‡, Jingbo WANG, Chihang WEI, Zhihuan SONG

State Key Laboratory of Industrial Control Technology, College of Control Science and Engineering, Zhejiang University, Hangzhou 310027, China

E-mail: xinminzhang@zju.edu.cn; wangjingbobo@zju.edu.cn; chhwei@zju.edu.cn; songzhihuan@zju.edu.cn

Received July 6, 2020; Revision accepted Jan. 6, 2021; Crosschecked Sept. 2, 2021

Abstract: Identifying factors that exert more influence on system output from data is one of the most challenging tasks in science and engineering. In this work, a sensitivity analysis of the generalized Gaussian process regression (SA-GGPR) model is proposed to identify important factors of the nonlinear counting system. In SA-GGPR, the GGPR model with Poisson likelihood is adopted to describe the nonlinear counting system. The GGPR model with Poisson likelihood inherits the merits of nonparametric kernel learning and Poisson distribution, and can handle complex nonlinear counting systems. Nevertheless, understanding the relationships between model inputs and output in the GGPR model with Poisson likelihood is not readily accessible due to its nonparametric and kernel structure. SA-GGPR addresses this issue by providing a quantitative assessment of how different inputs affect the system output. The application results on a simulated nonlinear counting system and a real steel casting-rolling process have demonstrated that the proposed SA-GGPR method outperforms several state-of-the-art methods in identification accuracy.

Key words: Important factors; Nonlinear counting system; Generalized Gaussian process regression; Sensitivity analysis; Steel casting-rolling process

https://doi.org/10.1631/FITEE.2000324 **CLC number:** TP277

1 Introduction

Evaluating factors that have an impact on system output is critical for decision-makers to identify critical control points (CCPs). For example, the steel industry is committed to reducing defects in steel products based on the defect analysis critical control point (DACCP) system. One step in the DACCP system is to determine CCPs where defect management efforts can be focused.

The study of important factor identification from observational data is usually based on the supervised learning model. Supervised learning is a class of systems that determine a predictive model using labeled data (Mohri et al., 2018). Linear regression is a supervised learning technique typically used in predicting and finding relationships among quantitative data (Talabis et al., 2014; Sugiyama, 2015). The popular linear regression model is partial least squares (PLS) regression, which has been widely used in various fields (Wold et al., 2001; Abdi, 2010; Kano and Ogawa, 2010; Shao and Tian, 2015; Ge et al., 2017; Zhang et al., 2017, 2019, 2020a; Ge, 2018). In PLS regression, PLS-Beta and PLS-VIP have been widely used to identify important factors (Wang et al., 2015). In PLS-Beta, the identification of important factors is based on the regression coefficients of the PLS model. PLS-VIP is based on the variable importance in the projection (VIP) score.

‡ Corresponding author

* Project supported by the National Natural Science Foundation of China (Nos. 62003301 and 61833014) and the Natural Science Foundation of Zhejiang Province, China (No. LQ21F030018)

ORCID: Xinmin ZHANG, https://orcid.org/0000-0002-4761-3969

Nevertheless, simple parametric models lack expressive power for complex nonlinear processes. Compared with simple parametric models, nonparametric regression models, such as random forest (RF) (Cutler et al., 2012) and Gaussian process regression (GPR) (Rasmussen and Williams, 2006), are more powerful in handling complex nonlinear processes. RF is a nonlinear ensemble learning method that constructs a number of decision trees on various subsamples of the dataset and uses averaging to improve the prediction accuracy. The identification of important factors in RF can be realized by the permutation importance criterion and out-of-bag (OOB) error estimates (referred to as RF-PI) (Biau, 2012). GPR is a kernel-based nonlinear regression method. Because the implicit feature mapping is used in GPR through the kernel function, the evaluation of important factors in GPR is not easily accessible. To solve this issue, Blix et al. (2017) proposed the Gaussian process sensitivity analysis and applied it to solve the oceanic chlorophyll problem. Zhang et al. (2020b) proposed a GPR and Hilbert-Schmidt independence criterion based identification method. However, the GPR model is designed for continuous real-valued outputs with a Gaussian assumption, which does not hold in some engineering application studies. For example, causal analysis of defects in steel products is to discover the factors that affect the number of defects, which is the count data output; the Gaussian assumption is invalid and the GPR model cannot be directly applied.

In this work, a novel method, called the sensitivity analysis of the generalized Gaussian process regression (SA-GGPR) model, is proposed to identify important factors of the nonlinear counting system. In SA-GGPR, the GGPR model with Poisson likelihood is adopted to describe the nonlinear counting system. The GGPR model with Poisson likelihood inherits the merits of nonparametric kernel learning and Poisson distribution, and can deal with complex nonlinear counting systems. Nevertheless, for the GGPR model with Poisson likelihood, the identification of model inputs that have a significant effect on the system output is not easily accessible due to its nonparametric and kernel structure. SA-GGPR deals with this issue by providing a quantitative assessment of how different inputs affect the system output in terms of sensitivity measure. The proposed method is first validated on a simulated nonlinear counting system and then applied to a real steel casting-rolling process. The results demonstrate the feasibility and reliability of the proposed SA-GGPR method.

2 Conventional methods

In this section, brief descriptions of PLS-Beta, PLS-VIP, and RF-PI are presented. PLS-Beta and PLS-VIP are widely used to identify important factors of linear systems, while RF-PI is widely used for nonlinear systems.

2.1 PLS-Beta

PLS regression is a popular supervised learning method that predicts the system output from a set of inputs by constructing a latent variable model (Abdi, 2010). Consider a training dataset with input $\boldsymbol{X}\in\mathbb{R}^{N\times M}$ and output $\boldsymbol{y}\in\mathbb{R}^{N}$, where N and M represent the number of samples and the number of input variables, respectively. In PLS regression, $\boldsymbol{X}\in\mathbb{R}^{N\times M}$ and $\boldsymbol{y}\in\mathbb{R}^{N}$ are decomposed as

$$\begin{cases}\boldsymbol{X}=\boldsymbol{T}\boldsymbol{P}^{\mathrm{T}}+\boldsymbol{E},\\ \boldsymbol{y}=\boldsymbol{T}\boldsymbol{q}+\boldsymbol{f},\\ \boldsymbol{T}=\boldsymbol{X}\boldsymbol{W}(\boldsymbol{P}^{\mathrm{T}}\boldsymbol{W})^{-1},\end{cases}\tag{1}$$

where $\boldsymbol{P}\in\mathbb{R}^{M\times R}$ and $\boldsymbol{T}\in\mathbb{R}^{N\times R}$ represent the loading and score matrices of $\boldsymbol{X}$, respectively. $\boldsymbol{q}\in\mathbb{R}^{R}$ represents the loading vector of $\boldsymbol{y}$, $\boldsymbol{W}\in\mathbb{R}^{M\times R}$ represents the weighting matrix, and $\boldsymbol{E}\in\mathbb{R}^{N\times M}$ and $\boldsymbol{f}\in\mathbb{R}^{N}$ are residuals. R represents the number of retained latent variables. The standard algorithm for constructing a PLS regression model is nonlinear iterative partial least squares (NIPALS) (Wold et al., 2001).

In PLS regression, the output estimate $\hat{\boldsymbol{y}}$ can be expressed as

$$\begin{cases}\hat{\boldsymbol{y}}=\boldsymbol{X}\boldsymbol{\beta}_{\mathrm{pls}},\\ \boldsymbol{\beta}_{\mathrm{pls}}=\boldsymbol{W}(\boldsymbol{P}^{\mathrm{T}}\boldsymbol{W})^{-1}\boldsymbol{q},\end{cases}\tag{2}$$

where $\boldsymbol{\beta}_{\mathrm{pls}}\in\mathbb{R}^{M}$ is a regression coefficient vector, indicating the importance of each input in describing the output. The absolute value of $\boldsymbol{\beta}_{\mathrm{pls}}$ is employed in PLS-Beta to identify important factors.

2.2 PLS-VIP

PLS-VIP identifies important factors in terms of VIP score, which measures the importance of each

input in the projection used in a PLS model (Wang et al., 2015). The VIP score of the m^{th} variable is expressed as

$$\mathrm{VIP}_m = \sqrt{\frac{M\sum_{r=1}^{R}\left[q_r^2\boldsymbol{t}_r^{\mathrm{T}}\boldsymbol{t}_r(w_{m,r}/\parallel \boldsymbol{w}_r \parallel)^2\right]}{\sum_{r=1}^{R} q_r^2\boldsymbol{t}_r^{\mathrm{T}}\boldsymbol{t}_r}}, \quad (3)$$

where $\boldsymbol{t}_r$ and $\boldsymbol{w}_r$ denote the r^{th} column vectors of $\boldsymbol{T}$ and $\boldsymbol{W}$, respectively, q_r represents the r^{th} element of $\boldsymbol{q}$, and $w_{m,r}$ represents the m^{th} element of $\boldsymbol{w}_r$.

2.3 RF-PI

RF (Cutler et al., 2012) is a nonlinear ensemble learning method describing the input-output relationship by constructing a set of decision trees. Each tree is built on the bootstrap subset of the dataset. During the tree-growing process, the best split of each node is calculated from the randomly selected subset of the total input variables.

RF uses the permutation importance criterion (referred to as RF-PI) to identify important factors of nonlinear systems (Cutler et al., 2012). The idea of RF-PI is that if one input variable is not important, the model accuracy will not deteriorate when the value of that input variable is permuted. Mathematically, the importance score VI_m for the m^{th} input variable is calculated by averaging the difference in OOB errors before and after the permutation over all trees (Bühlmann, 2012). Let $\hat{f}^b(\cdot)$ denote the tree grown on the b^{th} bootstrap subset ($b = 1, 2, \ldots, B$) and let OOB_b denote the OOB observation corresponding to the b^{th} bootstrap subset. A step-by-step procedure for calculating VI_m is presented as follows:

1. For $b = 1$, search for OOB_b.

2. Calculate OOB error $\mathrm{err}_b^{\mathrm{OOB}}$ by $\hat{f}^b(\cdot)$ over OOB_b:

$$\mathrm{err}_b^{\mathrm{OOB}} = \frac{1}{n_{\mathrm{OOB}_b}} \sum_{\substack{i=1 \\ (i\in \mathrm{OOB}_b)}}^{N} \left(y_i - \hat{f}^b(\boldsymbol{x}_i)\right)^2, \quad (4)$$

where n_{OOB_b} is the number of observations in OOB_b, y_i is the measured value, and $\hat{f}^b(\boldsymbol{x}_i)$ is the output estimate.

3. For the m^{th} variable, $m = 1, 2, \ldots, M$:

(1) Permutate the value of the m^{th} input variable in OOB_b, and the permuted OOB_b is denoted as $\mathrm{OOB}_{b,m}$.

(2) Predict $\mathrm{OOB}_{b,m}$ using $\hat{f}^b(\cdot)$, and then calculate $\mathrm{err}_{b,m}^{\mathrm{OOB}}$.

4. Repeat steps 1–3 for $b = 2, 3, \ldots, B$.

5. Calculate VI_m as

$$\mathrm{VI}_m = \frac{1}{B}\sum_{b=1}^{B}(\mathrm{err}_{b,m}^{\mathrm{OOB}} - \mathrm{err}_b^{\mathrm{OOB}})^2. \quad (5)$$

3 Sensitivity analysis of generalized Gaussian process regression

In this section, a new method called SA-GGPR is presented to identify important factors of the nonlinear counting system. In SA-GGPR, the GGPR model with Poisson likelihood is adopted to describe the nonlinear counting system. The GGPR model with Poisson likelihood inherits the merits of nonparametric kernel learning and Poisson distribution, and can deal with complex nonlinear counting systems. However, it is not intuitive to understand the relationship between model inputs and output in GGPR with Poisson likelihood due to its nonparametric kernel structure. To solve this problem, SA-GGPR is proposed in this work. SA-GGPR determines the factors that have a significant effect on the system output in terms of the sensitivity measure.

3.1 Generalized Gaussian process regression

GGPR constructs flexible nonparametric Bayesian models in which the observation likelihood is parameterized by an exponential family distribution (EFD) and the latent function is related to the output distribution via a link function (Chan and Dong, 2011). Specifically, GGPR consists of the following three components:

1. Random component

The output variable y follows an EFD, with a probability density function (or probability mass function) taking the form of

$$p(y|\,\theta,\phi) = h(y,\phi)\exp\left\{\frac{1}{a(\phi)}[y\theta - b(\theta)]\right\}, \quad (6)$$

where θ is the natural parameter of EFD and ϕ is the dispersion parameter. $h(y,\ \phi)$ and $a(\phi)$ are known functions, and $b(\theta)$ is a log-partition function normalizing the distribution. The mean and variance of y are functions of $b(\theta)$ and $a(\phi)$:

$$\mu = \mathbb{E}[y] = b'(\theta),\ \mathrm{var}(y) = b''(\theta)a(\phi), \quad (7)$$

where $b'(\theta)$ and $b''(\theta)$ are the first and second derivatives of b with respect to θ, respectively. EFD provides a general framework for selecting a specific

parametric distribution in terms of output domain (e.g., continuous, discrete, and count-type).

2. Latent function

A zero-mean Gaussian process is placed on the latent function, $\eta(\boldsymbol{x}) \sim \mathrm{GP}\big(0, k(\boldsymbol{x}, \boldsymbol{x}')\big)$, where $k(\boldsymbol{x}, \boldsymbol{x}')$ is the covariance function (e.g., squared exponential).

3. Link function

A monotonic and differentiable function called the link function, $g(\cdot)$, is introduced to relate the mean of the output distribution with the latent function, $\eta(\boldsymbol{x}) = g(\mu)$.

Formally, the GGPR model is specified by

$$\begin{aligned} &\eta(\boldsymbol{x}) \sim \mathrm{GP}\big(0, k(\boldsymbol{x}, \boldsymbol{x}')\big),\ y \sim p(y|\,\theta, \phi),\\ &g(\mathbb{E}[y|\,\theta]) = \eta(\boldsymbol{x}). \end{aligned} \tag{8}$$

Based on the monotonic property of $g(\cdot)$, the functional relationship between the mean of the output distribution and the latent function can be rewritten as

$$\mu = g^{-1}(\eta(\boldsymbol{x})), \tag{9}$$

where $g^{-1}(\cdot)$ is called the inverse-link function.

By integrating Eq. (7) into Eq. (8), we can obtain

$$\eta(\boldsymbol{x}) = g(\mathbb{E}[y|\,\theta]) = g(b'(\theta)). \tag{10}$$

Furthermore, we can obtain the functional relationship between the parameter θ and the latent function $\eta(\boldsymbol{x})$:

$$\theta(\eta(\boldsymbol{x})) = [b']^{-1}\big(g^{-1}(\eta(\boldsymbol{x}))\big). \tag{11}$$

Using Eq. (11), another form of GGPR is obtained:

$$\begin{aligned} &\eta(\boldsymbol{x}) \sim \mathrm{GP}\big(0, k(\boldsymbol{x}, \boldsymbol{x}')\big),\ y \sim p\big(y|\,\theta(\eta(\boldsymbol{x})), \phi\big),\\ &\theta(\eta(\boldsymbol{x})) = [b']^{-1}\big(g^{-1}(\eta(\boldsymbol{x}))\big). \end{aligned} \tag{12}$$

For the GGPR inference, given a set of training samples $(\boldsymbol{X},\ \boldsymbol{y})$, the distribution of the latent values $\boldsymbol{\eta} = \big[\eta(\boldsymbol{x}_1),\ \eta(\boldsymbol{x}_2),\ \ldots,\ \eta(\boldsymbol{x}_N)\big]$ corresponding to $\boldsymbol{X}$ is jointly Gaussian $\boldsymbol{\eta}|\boldsymbol{X} \sim N(\boldsymbol{0}, \boldsymbol{K})$, where $\boldsymbol{K} \in \mathbb{R}^{N\times N}$ is the kernel or covariance matrix with entries $[\boldsymbol{K}]_{ij} = k(\boldsymbol{x}_i, \boldsymbol{x}_j)$. The posterior probability distribution of $\boldsymbol{\eta}$ given the observed output $\boldsymbol{y}$ can be calculated with Bayes' theorem:

$$p(\boldsymbol{\eta}|\boldsymbol{y}, \boldsymbol{X}) = \frac{p(\boldsymbol{y}|\boldsymbol{\theta}(\boldsymbol{\eta}))p(\boldsymbol{\eta}|\boldsymbol{X})}{p(\boldsymbol{y}|\boldsymbol{X})}, \tag{13}$$

where $p\big(\boldsymbol{y}|\boldsymbol{X}\big)$ denotes the marginal likelihood given by

$$p(\boldsymbol{y}|\boldsymbol{X}) = \int p(\boldsymbol{y}|\boldsymbol{\theta}(\boldsymbol{\eta}))p(\boldsymbol{\eta}|\boldsymbol{X})\mathrm{d}\boldsymbol{\eta}. \tag{14}$$

Given a novel input $\boldsymbol{x}_q$, the posterior distribution of the novel latent value $\eta_q = \eta(\boldsymbol{x}_q)$ is obtained by marginalization over the posterior distribution in Eq. (13) (i.e., averaging over all possible latent functions):

$$p\big(\eta_q|\boldsymbol{X}, \boldsymbol{x}_q, \boldsymbol{y}\big) = \int p\big(\eta_q|\boldsymbol{\eta}, \boldsymbol{X}, \boldsymbol{x}_q\big)p\big(\boldsymbol{\eta}|\boldsymbol{X}, \boldsymbol{y}\big)\mathrm{d}\boldsymbol{\eta}, \tag{15}$$

where $p(\eta_q|\,\boldsymbol{\eta}, \boldsymbol{X}, \boldsymbol{x}_q) = N(\eta_q|\,\boldsymbol{k}_q\boldsymbol{K}^{-1}\boldsymbol{\eta}, k_{qq} - \boldsymbol{k}_q\boldsymbol{K}^{-1}\boldsymbol{k}_q^{\mathrm{T}})$ with $\boldsymbol{k}_q = [k(\boldsymbol{x}_q, \boldsymbol{x}_i)]$ and $k_{qq} = k(\boldsymbol{x}_q, \boldsymbol{x}_q)$. According to the Gaussian approximation inference (Nickisch and Rasmussen, 2008), the approximate posterior for η_q is given by

$$p(\eta_q|\boldsymbol{X}, \boldsymbol{x}_q, \boldsymbol{y}) = N(\eta_q|\hat{\mu}_\eta, \hat{\sigma}_\theta^2), \tag{16}$$

where the mean and variance are

$$\hat{\mu}_\eta = \boldsymbol{k}_q(\boldsymbol{K} + \tilde{\boldsymbol{W}})^{-1}\tilde{\boldsymbol{t}}, \tag{17}$$

$$\hat{\sigma}_\eta^2 = k_{qq} - \boldsymbol{k}_q(\boldsymbol{K} + \tilde{\boldsymbol{W}})^{-1}\boldsymbol{k}_q^{\mathrm{T}}. \tag{18}$$

Here, $\tilde{\boldsymbol{W}} = \mathrm{diag}(\tilde{w}_1, \tilde{w}_2, \ldots, \tilde{w}_n)$ is a positive definite diagonal matrix, and $\tilde{\boldsymbol{t}}$ is a target vector. $\tilde{\boldsymbol{W}}$ and $\tilde{\boldsymbol{t}}$ are determined by the Taylor approximation algorithm (Chan and Dong, 2011).

3.2 SA-GGPR

Although GGPR is a flexible nonparametric Bayesian regression model, the evaluation of important factors in GGPR is not easily accessible due to the implementation of implicit feature mapping. SA-GGPR is proposed to solve this problem. SA-GGPR determines how different values of an input variable affect the output. Mathematically, the measure of the sensitivity of variable m is given as

$$s_m = \int \left(\frac{\partial\phi(\boldsymbol{x})}{\partial\boldsymbol{x}_j}\right)^2 p(\boldsymbol{x})\mathrm{d}\boldsymbol{x}, \tag{19}$$

where $\phi(\boldsymbol{x})$ denotes the objective function and $p(\boldsymbol{x})$ is the probability density function. The calculation of sensitivity involves taking the partial derivative of $\phi(\boldsymbol{x})$ with respect to the input factor $\boldsymbol{x}_m$. In this work, the predictive mean function μ_f of GGPR is specified as the objective function $\phi(\boldsymbol{x})$. To simplify

the calculation, the objective function is rewritten as

$$\begin{aligned}\phi_\eta(\boldsymbol{x}) &= \hat{\mu}_\eta \\ &= \boldsymbol{k}_q(\boldsymbol{K}+\tilde{\boldsymbol{W}})^{-1}\tilde{\boldsymbol{t}} \\ &= \boldsymbol{k}_q\boldsymbol{\alpha}_\eta \\ &= \sum_{p=1}^{N}\alpha_{\eta,p}k(\boldsymbol{x}_p,\boldsymbol{x}_q),\end{aligned} \tag{20}$$

where $\boldsymbol{\alpha}_\eta$ represents a weight vector. Then, an empirical estimate of s_m can be calculated by

$$\begin{aligned}\hat{s}_m =& \frac{1}{N}\sum_{q=1}^{N}\left(\frac{\partial\phi_\eta(\boldsymbol{x}_q)}{\partial x_{q,m}}\right)^2 \\ =& \frac{1}{N}\sum_{q=1}^{N}\left(\frac{\partial\sum_{p=1}^{N}\alpha_{\eta,p}k(\boldsymbol{x}_p,\boldsymbol{x}_q)}{\partial x_{q,m}}\right)^2 \\ =& \frac{1}{N}\sum_{q=1}^{N}\left(\sum_{p=1}^{N}\frac{\alpha_{\eta,p}(x_{p,m}-x_{q,m})}{\lambda^2}k(\boldsymbol{x}_p,\boldsymbol{x}_q)\right)^2.\end{aligned} \tag{21}$$

It is worth noting that the values of the positive definite diagonal matrix $\tilde{\boldsymbol{W}}$ and the target vector $\tilde{\boldsymbol{t}}$ in Eq. (20) depend on the choice of the type of the observation likelihood. Because the focus of this work is on the identification of important factors of the counting system, the Poisson likelihood is selected. As a discrete probability distribution, the Poisson likelihood is suitable for applications that involve counting the number of occurrences of random events (Hutchinson and Holtman, 2005; Coxe et al., 2009). The probability mass function of Poisson likelihood is defined as

$$p(y|\lambda) = \frac{1}{y!}\lambda^y \mathrm{e}^{-\lambda}, \tag{22}$$

where λ denotes the mean number of events (also known as the shape parameter or rate parameter).

As mentioned in Eq. (6), the exponential family generalizes a wide variety of distributions by changing the likelihood parameters. For GGPR with Poisson likelihood, the parameter $\theta = \ln\lambda$, the dispersion $\phi = 1$, and the parameter functions in the exponential family form are

$$a(\phi) = 1,\ b(\theta) = \mathrm{e}^\theta,\ h(y,\phi) = 1/y!. \tag{23}$$

The canonical link function for GGPR with Poisson likelihood can be expressed as

$$\mathbb{E}[y] = g^{-1}(\eta) = \mathrm{e}^\eta = \lambda,\ g(\mu) = \ln\mu. \tag{24}$$

According to the Taylor approximation inference of GGPR with Poisson likelihood (Nickisch and Rasmussen, 2008), the target elements t_i and diagonal elements w_i can be calculated as

$$\tilde{t}_i = \ln(y_i + c) - \frac{c}{y_i + c}, \tag{25}$$

$$\tilde{w}_i = \frac{1}{y_i + c}, \tag{26}$$

where $c \geq 0$ (e.g., $c = 0.001$) is a constant to prevent from taking the logarithm of zero.

A step-by-step procedure for implementing the proposed SA-GGPR algorithm is summarized in Algorithm 1. The model hyperparameters (kernel parameters) in Algorithm 1 are optimized by maximizing the marginal likelihood using the GPML toolbox (Rasmussen and Nickisch, 2010). The SA-GGPR codes can be downloaded from https://github.com/IBD-CSE/SAGGPR.

Algorithm 1 SA-GGPR

Input: input data matrix $\boldsymbol{X} \in \mathbb{R}^{N\times M}$ and output data vector $\boldsymbol{y} \in \mathbb{R}^N$. The kernel (covariance) function is squared exponential kernel and the observation likelihood function is Poisson likelihood

Output: the importance score VI_m $(m = 1, 2, \ldots, M)$

1: Construct the GGPR model and obtain the objective function $\phi_\eta(\boldsymbol{x}) = \boldsymbol{k}_q(\boldsymbol{K}+\tilde{\boldsymbol{W}})^{-1}\tilde{\boldsymbol{t}} = \boldsymbol{k}_q\boldsymbol{\alpha}_\eta = \sum_{p=1}^{N}\alpha_{\eta,p}k(\boldsymbol{x}_p,\boldsymbol{x}_q)$

2: **for** $m = 1, 2, \ldots, M$ **do**

3: Calculate the measure of sensitivity of variable m: $\hat{s}_m = \frac{1}{N}\sum_{q=1}^{N}\left(\sum_{p=1}^{N}\frac{\alpha_{\eta,p}(x_{p,m}-x_{q,m})}{\lambda^2}k(\boldsymbol{x}_p,\boldsymbol{x}_q)\right)^2$

4: **end for**

5: Calculate the importance score of variable m: $\mathrm{VI}_m = \hat{s}_m / \sum_{j=1}^{M}\hat{s}_j$

4 Case study

In this section, we apply the proposed SA-GGPR method to a simulated nonlinear counting system and a real steel casting-rolling process. The application results are compared with those of the PLS-Beta, PLS-VIP, RF-PI, and SA-GPR methods in terms of identification accuracy. In SA-GPR, the standard GPR using Gaussian likelihood is adopted.

4.1 Numerical example

4.1.1 Data generation

Data is generated from the following nonlinear counting system:

$$\begin{cases} x_1 = t^2 - t + 1 + \varepsilon, \\ x_2 = \sin t + \varepsilon, \\ x_3 = t^3 + t + \varepsilon, \\ x_4 = 2\cos(0.08t)\sin(0.06t) + \varepsilon, \\ x_5 = \sin(0.4t) + 1.5\cos(0.2t) + \varepsilon, \\ \mu = \exp(0.7x_1 + 0.4x_4 + 0.5x_5), \\ y \sim \text{Poisson}(\mu), \end{cases} \tag{27}$$

where x_1–x_5 are input variables, μ denotes the mean of the output variable distribution, y is the output variable, t is uniformly distributed within $[-2,\ 2]$, and ε is a Gaussian measurement noise with zero mean and a standard deviation of 0.1.

Note that the output variable y is discrete count data, and that the important factors or variables affecting the output variable are x_1, x_4, and x_5.

4.1.2 Performance measure

To evaluate the prediction performance of each method, the root mean squared error (RMSE) and correlation coefficient R are used. RMSE and R are calculated as

$$\text{RMSE} = \sqrt{\frac{1}{N_\text{t}}\sum_{i=1}^{N_\text{t}}(y_i - \hat{y}_i)^2}, \tag{28}$$

$$R = \frac{\sum_{i=1}^{N_\text{t}}(y_i - \bar{y})(\hat{y}_i - \bar{\hat{y}})}{\sqrt{\sum_{i=1}^{N_\text{t}}(y_i - \bar{y})^2}\sqrt{\sum_{i=1}^{N_\text{t}}(\hat{y}_i - \bar{\hat{y}})^2}}, \tag{29}$$

where y_i and $\hat{y}_i$ represent the actual observed value and the predicted value respectively, $\bar{y}$ and $\bar{\hat{y}}$ represent the mean values of y_i and $\hat{y}_i$ respectively, and N_t represents the size of testing samples.

To evaluate the identification performance of each method for the important variables, a confusion matrix (also known as an error matrix) is employed. The confusion matrix reports information about the predicted and actual classes. Each column of the matrix represents the instance in a predicted class, while each row represents the instance in an actual class. The first three variables in the order of variable importance predicted by each method are classified as important variables. The remaining two variables at the bottom are classified as unimportant variables. Table 1 shows a confusion matrix, in which c_1–c_4 denote the numbers of variables identified by each method in each group. Three metrics that are calculated from the confusion matrix are commonly employed to evaluate the identification performance of each method quantitatively. They are defined as

$$\text{Accuracy} = (c_1 + c_4)/(c_1 + c_2 + c_3 + c_4), \tag{30}$$

$$\text{Recall} = c_1/(c_1 + c_2), \tag{31}$$

$$\text{Selectivity} = c_4/(c_3 + c_4). \tag{32}$$

Table 1 Confusion matrix

		Predicted class	
		Important	Unimportant
Actual class	Important	c_1	c_2
	Unimportant	c_3	c_4

4.1.3 Results and discussion

Using the above data generation process (simulation system), 2000 samples are generated. The whole dataset is divided into the training dataset and the testing dataset according to the 10-fold cross-validation criterion. That is, the dataset is randomly divided into ten parts, nine of which are used for training and the remaining one for testing. This process can be repeated 10 times, and the testing data used is different each time. Table 2 shows the mean prediction accuracy of each method in terms of RMSEP (RMSE of prediction) and R criteria. In PLS, the number of latent variables used is set at 3, which is determined by cross-validation. In RF, the number of trees is set at 500, which is determined by the OOB error criterion. In GPR and GGPR, the model hyperparameters are optimized by maximizing the marginal likelihood using the GPML toolbox (Rasmussen and Nickisch, 2010). From Table 2, it can be seen that GGPR is the most accurate model

Table 2 Prediction results of different methods in the numerical example

Method	RMSEP	R
PLS	10.8894	0.9379
RF	5.4181	0.9846
GPR	7.5924	0.9693
GGPR	5.0024	0.9867

among all the methods. Thus, the implementation of SA-GGPR is feasible.

Table 3 shows the results of SA-GGPR in identifying the important factors for the above nonlinear counting system in terms of confusion matrix criterion. For comparison, the identification results of PLS-Beta, PLS-VIP, RF-PI, and SA-GPR are also provided. In Table 3, the identification result is the average of 50 repeated experiments. From Table 3, it can be seen that PLS-Beta, PLS-VIP, RF-PI, and SA-GPR yield poor identification performance with low accuracy, recall, and selectivity. In comparison, the proposed SA-GGPR achieves the best identification performance with the highest accuracy, recall, and selectivity. The detailed identification results are given in Fig. 1, where the results are shown visually in boxplots. In Fig. 1, the green boxes represent important variable IDs and the white boxes represent unimportant variable IDs. The importance of variables is normalized so that the sum is one. As shown

Table 3 Identification results of important factors by different methods in the numerical example

Method	Accuracy	Recall	Selectivity
PLS-Beta	0.60	0.67	0.50
PLS-VIP	0.20	0.33	0
RF-PI	0.47	0.56	0.34
SA-GPR	0.61	0.67	0.51
SA-GGPR	0.98	0.98	0.97

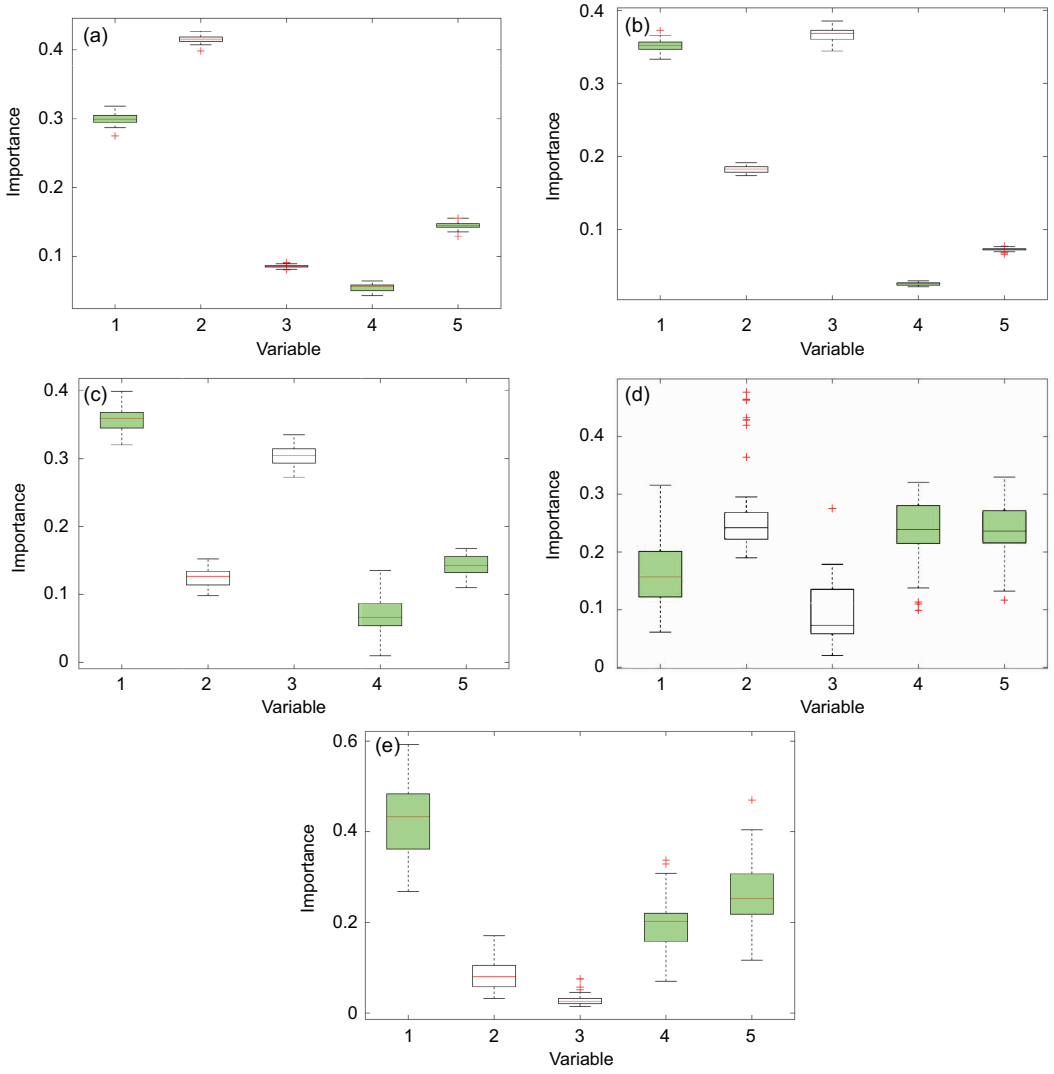

Fig. 1 Identification results of important factors by different methods in the numerical example: (a) PLS-Beta; (b) PLS-VIP; (c) RF-PI; (d) SA-GPR; (e) SA-GGPR. References to color refer to the online version of this figure

in Fig. 1, PLS-Beta, PLS-VIP, RF-PI, and SA-GPR cannot fully identify all important variables (x_1, x_4, and x_5). In contrast, the proposed SA-GGPR successfully identified that x_1, x_4, and x_5 are important variables, which is consistent with the experimental design.

4.2 Steelmaking process

In this subsection, we apply the proposed SA-GGPR method to solve a practical engineering problem and identify the most influential operating process variables that affect the number of defects in the steel plate.

The defect data contains 5000 samples and 71 process variables, and is collected from an industrial casting-rolling process. The input variables include the casting speed, rolling temperature, cooling temperature, and so on. The output variable is the number of surface defects in the steel plate, which is a count-type output. According to the knowledge and experience of experts, the important and unimportant variables are listed in Table 4. Similar to the numerical example, the confusion matrix criterion is employed to evaluate the identification performance of each method quantitatively. From Table 4, it can be seen that the actual classes include 28 important variables and 43 unimportant variables. For the predicted classes, the first 28 variables in the order of variable importance predicted by each method are classified as important variables, and the remaining 43 variables at the bottom are classified as unimportant variables. Based on Eqs. (30)–(32), three metrics (accuracy, recall, and selectivity) can be calculated.

Before implementing SA-GGPR, the accuracy of the GGPR model first needs to be evaluated. We randomly split the whole dataset into two parts. The first is the training dataset with 4500 samples and the second is the testing dataset with 500 samples. The training dataset was used to train the model, and the built model was then evaluated using the testing dataset. The above procedure was repeated 20 times. Table 5 summarizes the average prediction error of each model. In PLS, the number of latent variables used was set at 35. In RF, the number of trees was set at 500. In GPR and GGPR, the model hyperparameters were optimized by maximizing the marginal likelihood using the GPML toolbox (Rasmussen and Nickisch, 2010). As shown in Table 5, GGPR is the most accurate model with the smallest RMSEP and the largest R among all the methods. Thus, the implementation of SA-GGPR is feasible.

Table 6 shows the identification results of important factors by different methods in terms of the confusion matrix criterion. PLS-Beta, PLS-VIP, RF-PI, and SA-GPR exhibited low accuracy, recall, and selectivity. As a consequence, the proposed SA-GGPR had the best performance with the highest accuracy, recall, and selectivity. More detailed identification results of each method are shown in Fig. 2. The green boxes represent important variable IDs and the white boxes represent unimportant variable IDs. It can be seen that the proposed SA-GGPR distinguished the important variables from the other variables more accurately and clearly than the other methods.

To investigate the computational cost of each method, Table 7 presents the comparison results in

Table 4 Importance based on the knowledge and experience of experts

Importance	Variable ID	Amount
Important	1, 3, 4, 5, 22, 24, 25, 41, 42, 48, 54–71	28
Unimportant	The others	43

Table 5 Prediction results by different methods in the casting-rolling process

Method	RMSEP	R
PLS	6.2552	0.5215
RF	3.7334	0.8764
GPR	5.1568	0.7128
GGPR	0.8396	0.9966

Table 6 Important factors identified by different methods in the casting-rolling process

Method	Accuracy	Recall	Selectivity
PLS-Beta	0.47	0.37	0.52
PLS-VIP	0.54	0.43	0.61
RF-PI	0.51	0.43	0.55
SA-GPR	0.70	0.69	0.70
SA-GGPR	0.80	0.75	0.84

Table 7 Computational time comparison of different methods

Method	Time (s)
PLS-Beta	0.11
PLS-VIP	0.18
RF-PI	32.85
SA-GPR	583.68
SA-GGPR	273.20

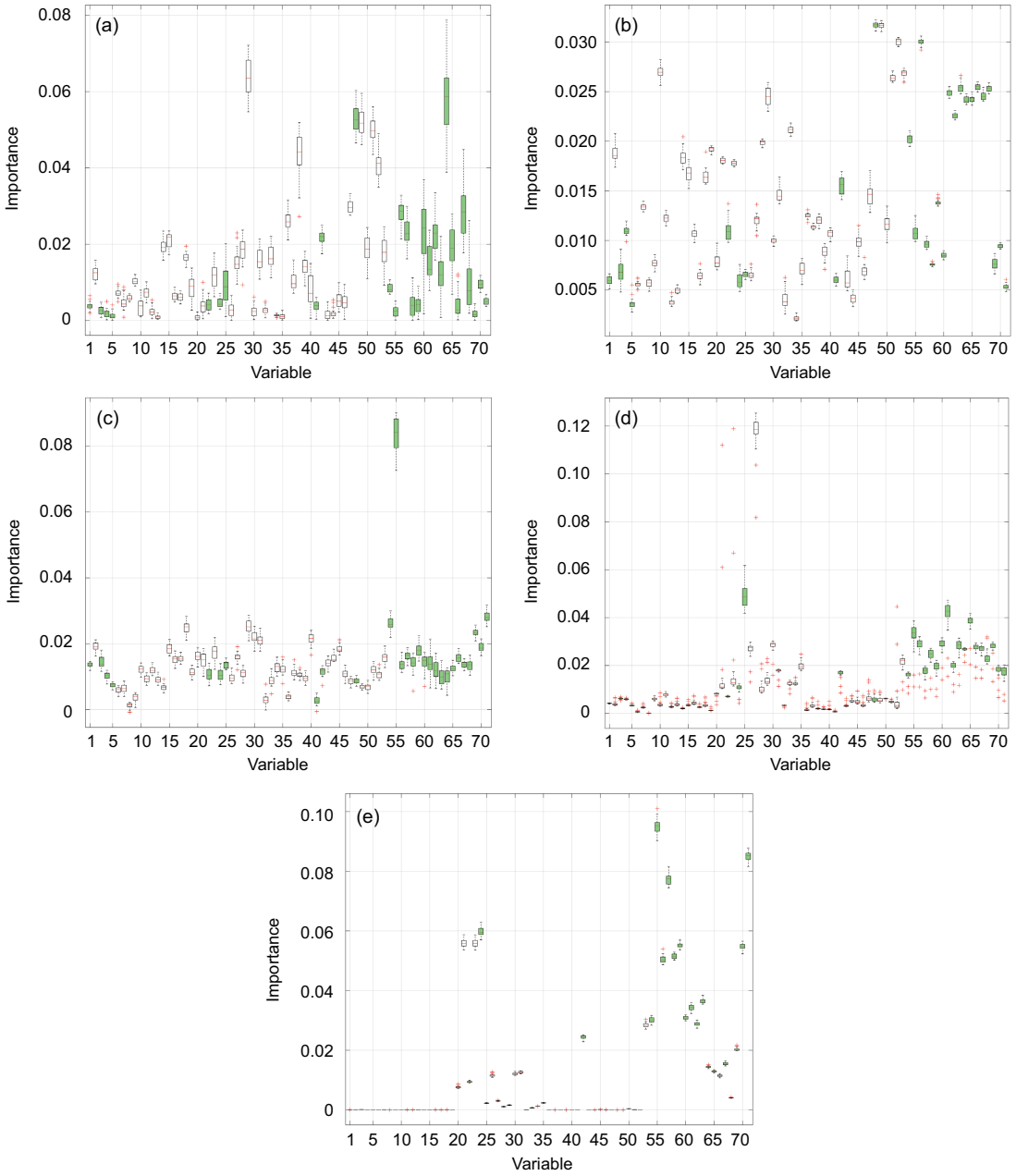

Fig. 2 Identification results of important factors by different methods in the casting-rolling process: (a) PLS-Beta; (b) PLS-VIP; (c) RF-PI; (d) SA-GPR; (e) SA-GGPR. References to color refer to the online version of this figure

the same computing environment, which is a desktop computer with Windows 10 (64 bit), Intel (R) Core (TM) i7-9700 CPU, 16 GB RAM, and MATLAB R2019b. From Table 7, it can be seen that PLS-Beta and PLS-VIP require shorter computing time than the other methods. The computing time used in RF-PI is longer than those in PLS-Beta and PLS-VIP, but it is shorter than those in SA-GPR and SA-GGPR. The computing time used in SA-GGPR is shorter than that in SA-GPR. As a result, the proposed SA-GGPR obtains the highest identification accuracy without incurring the highest computational cost. It should be emphasized that in many cases, accuracy is more important than speed. Therefore, the proposed SA-GGPR can be widely used in many important factor identification cases.

5 Conclusions

In this research, the sensitivity analysis of the generalized Gaussian process regression (SA-GGPR) model is proposed to identify important factors of the nonlinear counting system. On one hand, the GGPR model with Poisson likelihood is adopted to describe the nonlinear counting system. The GGPR model with Poisson likelihood inherits the merits of nonparametric kernel learning and Poisson distribution, and can handle complex nonlinear counting systems. On the other hand, the identification of important factors for the nonlinear counting system is introduced using SA-GGPR. SA-GGPR implements a quantitative assessment of how different inputs affect the system output based on the sensitivity measure. The usefulness and advantages of SA-GGPR are verified by its application to a simulated nonlinear counting system and a real steel casting-rolling process. The application results show that the proposed SA-GGPR method is feasible and more accurate in identifying important factors of the nonlinear counting system compared with several state-of-the-art methods.

Contributors

Xinmin ZHANG designed the research, processed the data, and drafted the manuscript. Jingbo WANG, Chihang WEI, and Zhihuan SONG revised and finalized the paper.

Compliance with ethics guidelines

Xinmin ZHANG, Jingbo WANG, Chihang WEI, and Zhihuan SONG declare that they have no conflict of interest.

References

Abdi H, 2010. Partial least squares regression and projection on latent structure regression (PLS regression). *WIREs Comput Stat*, 2(1):97-106. https://doi.org/10.1002/wics.51

Biau G, 2012. Analysis of a random forests model. *J Mach Learn Res*, 13(1):1063-1095.

Blix K, Camps-Valls G, Jenssen R, 2017. Gaussian process sensitivity analysis for oceanic chlorophyll estimation. *IEEE J Sel Top Appl Earth Obs Remote Sens*, 10(4):1265-1277. https://doi.org/10.1109/JSTARS.2016.2641583

Bühlmann P, 2012. Bagging, boosting and ensemble methods. In: Gentle JE, Härdle WK, Mori Y (Eds.), Handbook of Computational Statistics. Springer, Berlin, Germany, p.985-1022. https://doi.org/10.1007/978-3-642-21551-3_33

Chan AB, Dong DX, 2011. Generalized Gaussian process models. Proc 24th IEEE Conf on Computer Vision and Pattern Recognition, p.2681-2688. https://doi.org/10.1109/CVPR.2011.5995688

Coxe S, West SG, Aiken LS, 2009. The analysis of count data: a gentle introduction to Poisson regression and its alternatives. *J Pers Assess*, 91(2):121-136. https://doi.org/10.1080/00223890802634175

Cutler A, Cutler DR, Stevens JR, 2012. Random forests. In: Zhang C, Ma YQ (Eds.), Ensemble Machine Learning: Methods and Applications. Springer, Boston, USA, p.157-175. https://doi.org/10.1007/978-1-4419-9326-7

Ge ZQ, 2018. Process data analytics via probabilistic latent variable models: a tutorial review. *Ind Eng Chem Res*, 57(38):12646-12661. https://doi.org/10.1021/acs.iecr.8b02913

Ge ZQ, Song ZH, Ding SX, et al., 2017. Data mining and analytics in the process industry: the role of machine learning. *IEEE Access*, 5:20590-20616. https://doi.org/10.1109/ACCESS.2017.2756872

Hutchinson MK, Holtman MC, 2005. Analysis of count data using Poisson regression. *Res Nurs Health*, 28(5):408-418. https://doi.org/10.1002/nur.20093

Kano M, Ogawa M, 2010. The state of the art in chemical process control in Japan: good practice and questionnaire survey. *J Process Contr*, 20(9):969-982. https://doi.org/10.1016/j.jprocont.2010.06.013

Mohri M, Rostamizadeh A, Talwalkar A, 2018. Foundations of Machine Learning. MIT Press, Cambridge, UK.

Nickisch H, Rasmussen CE, 2008. Approximations for binary Gaussian process classification. *J Mach Learn Res*, 9:2035-2078.

Rasmussen CE, Williams CKI, 2006. Gaussian Processes for Machine Learning. MIT Press, Cambridge, UK.

Rasmussen CE, Nickisch H, 2010. Gaussian processes for machine learning (GPML) toolbox. *J Mach Learn Res*, 11:3011-3015.

Shao WM, Tian XM, 2015. Adaptive soft sensor for quality prediction of chemical processes based on selective ensemble of local partial least squares models. *Chem Eng Res Des*, 95:113-132. https://doi.org/10.1016/j.cherd.2015.01.006

Sugiyama M, 2015. Introduction to Statistical Machine Learning. Morgan Kaufmann Publishers, Waltham, MA, USA.

Talabis M, McPherson R, Miyamoto I, et al., 2014. Information Security Analytics: Finding Security Insights, Patterns, and Anomalies in Big Data. Syngress, Waltham, MA, USA.

Wang ZX, He QP, Wang J, 2015. Comparison of variable selection methods for PLS-based soft sensor modeling. *J Process Contr*, 26:56-72. https://doi.org/10.1016/j.jprocont.2015.01.003

Wold S, Sjöström M, Eriksson L, 2001. PLS-regression: a basic tool of chemometrics. *Chemom Intell Lab Syst*, 58(2):109-130. https://doi.org/10.1016/S0169-7439(01)00155-1

Zhang XM, Kano M, Li Y, 2017. Locally weighted kernel partial least squares regression based on sparse nonlinear features for virtual sensing of nonlinear time-varying

processes. *Comput Chem Eng*, 104:164-171. https://doi.org/10.1016/j.compchemeng.2017.04.014

Zhang XM, Kano M, Matsuzaki S, 2019. A comparative study of deep and shallow predictive techniques for hot metal temperature prediction in blast furnace ironmaking. *Comput Chem Eng*, 130:106575. https://doi.org/10.1016/j.compchemeng.2019.106575

Zhang XM, Kano M, Song ZH, 2020a. Optimal weighting distance-based similarity for locally weighted PLS modeling. *Ind Eng Chem Res*, 59(25):11552-11558. https://doi.org/10.1021/acs.iecr.9b06847

Zhang XM, Wada T, Fujiwara K, et al., 2020b. Regression and independence based variable importance measure. *Comput Chem Eng*, 135:106757. https://doi.org/10.1016/j.compchemeng.2020.106757