Gaussian process regression with multiple response variables

Bo Wang[a,*], Tao Chen[b]

[a] *Department of Mathematics, University of Leicester, Leicester LE1 7RH, UK*
[b] *Department of Chemical and Process Engineering, University of Surrey, Guildford GU2 7XH, UK*

Abstract

Gaussian process regression (GPR) is a Bayesian non-parametric technology that has gained extensive application in data-based modelling of various systems, including those of interest to chemometrics. However, most GPR implementations model only a single response variable, due to the difficulty in the formulation of covariance function for correlated multiple response variables, which describes not only the correlation between data points, but also the correlation between responses. In the paper we propose a direct formulation of the covariance function for multi-response GPR, based on the idea that its covariance function is assumed to be the "nominal" uni-output covariance multiplied by the covariances between different outputs. The effectiveness of the proposed multi-response GPR method is illustrated through numerical examples and response surface modelling of a catalytic reaction process.

Keywords: Block coordinate descent, Covariance function, Cholesky decomposition, Gaussian process regression, Multiple response

1. Introduction

In recent years, Gaussian process regression (GPR) has received significant attention as a powerful statistical tool for data-driven modelling. In chemometrics and related areas, GPR has been applied to calibration of spectroscopic analysers [6, 16, 26], response surface modelling [27], dynamic process modelling [11, 15], system identification [5] and ensemble learning [12, 26], among others. The popularity of GPR is

[*] Corresponding author. Tel.: +44 116 252 2162, Fax: +44 116 252 3915.
Email addresses: bo.wang@le.ac.uk (Bo Wang), t.chen@surrey.ac.uk (Tao Chen)

partly due to its theoretical link to Bayesian non-parametric statistics [9, 17], infinite neural networks [14], kernel methods in machine learning [4, 24], and spatial statistics (where it is more widely known as kriging) [8]. In addition, various empirical studies have demonstrated that GPR attains prediction accuracy that is at least comparable to (and in many cases better than) other models such as neural networks [15, 21, 26, 27].

Despite the high uptake of GPR for various modelling tasks, there still exists some outstanding issues with the GPR method. Of particular interest in this paper is the need to model multiple response variables. Traditionally, one response variable is treated as a Gaussian process, and multiple responses are modelled independently without considering their correlation. This pragmatic and straightforward approach was taken in many applications (e.g. [7, 26, 27]), though it is not ideal. A key to modelling multi-response Gaussian processes is the formulation of covariance function that describes not only the correlation between data points, but also the correlation between responses. Neal [14] suggested to share all covariance terms between different outputs but the noise variance; however this approach may not be adequate because it has a single hyper-parameter (the noise variance) to differentiate multiple outputs. An alternative method, termed dependent GPR, is to treat Gaussian processes as the outputs of stable linear filters, i.e. the covariance function is indirectly parameterised by using these linear filters [2]. A specific covariance function was proposed in [18], where the focus was on correlated periodic signals that are not always relevant to chemometrics.

Against this background, the present paper develops a direct formulation of the covariance function for multi-response GPR. To this end, the Bayesian regression framework will be followed, since it is usually used for deriving GPR. Furthermore, this scheme will be extended to the scenarios where different responses may be observed at different covariate values. Then, optimisation algorithms to estimate the hyper-parameters of this covariance function will be presented. The effectiveness of the proposed multi-response GPR method will be illustrated through numerical examples and response surface modelling of a catalytic reaction process.

2. Overview of GPR

This section follows the framework of Bayesian regression to provide an overview of GPR. This framework will facilitate the extension of GPR from single response to multiple response variables in the next section.

Linear regression expresses the response variable y as a function of p-dimensional covariates $\mathbf{x} = [x_1, \ldots, x_p]^T$ parameterised by $\mathbf{w} = [w_1, \ldots, w_p]^T$:

$$y_i = \sum_{d=1}^{p} w_d \, x_{id} + \epsilon_i = \mathbf{x}_i^T \mathbf{w} + \epsilon_i, \quad i = 1, \ldots, n$$

where n is the number of observations (data points), x_{id} is the d-th covariate of $\mathbf{x}_i$, and ϵ_i is additive Gaussian noise with zero mean and unknown variance σ_ϵ^2.

Under the Bayesian framework, a prior distribution needs to be specified for the regression parameter $\mathbf{w}$. An usual choice is a Gaussian distribution with zero mean and diagonal covariance matrix: $p(\mathbf{w}) = G(\mathbf{0}, \sigma_w^2 \mathbf{I}_p)$, independent of ϵ_i, where $\mathbf{I}_p$ is the $p \times p$ identity matrix. The response, $\mathbf{y} = [y_1, \ldots, y_n]^T$, which is a linear function of $\mathbf{w}$ and the Gaussian noise ϵ_i, is also Gaussian distributed with zero mean, and covariance matrix $\mathbf{C}$. This is a Gaussian process [13, 21], i.e., $p(\mathbf{y}) = G(\mathbf{0}, \mathbf{C})$. The entry of $\mathbf{C}$ is

$$\begin{aligned} C_{ij} &= C(\mathbf{x}_i, \mathbf{x}_j) = < y_i y_j > \\ &= < \mathbf{x}_i^T \mathbf{w} \mathbf{w}^T \mathbf{x}_j > + < \epsilon_i \epsilon_j > \\ &= \mathbf{x}_i^T < \mathbf{w} \mathbf{w}^T > \mathbf{x}_j + < \epsilon_i \epsilon_j > \\ &= \sigma_w^2 \mathbf{x}_i^T \mathbf{x}_j + \delta_{ij} \sigma_\epsilon^2 \end{aligned} \tag{1}$$

where $<>$ is the expectation operator; $\delta_{ij} = 1$ if $i = j$ and 0 otherwise.

This Bayesian approach gives rise to a new non-parametric view of the regression problem. Instead of inferring parameter $\mathbf{w}$, the regression model can be summarised by a covariance function, $C(\mathbf{x}_i, \mathbf{x}_j)$. Furthermore, the form of covariance function is not restricted to that in equation (1), the only constraint being that it must generate a non-negative definite covariance matrix for any set of data points [13]. The following covariance function is widely used in the literature [16, 26, 27] and also adopted in this paper:

$$C(\mathbf{x}_i, \mathbf{x}_j; \boldsymbol{\theta}) \quad = \quad a_0 + a_1 \sum_{d=1}^{p} x_{id} x_{jd} + v_0 \exp\left(-\sum_{d=1}^{p} \eta_d (x_{id} - x_{jd})^2\right) + \delta_{ij} \sigma_\epsilon^2$$

where $\boldsymbol{\theta} = (a_0, a_1, \eta_1, \ldots, \eta_p, v_0, \sigma_\epsilon^2)$ is the vector of hyper-parameters. The exponential term is similar to the form of radial basis function, recognising high correlation between outputs with nearby inputs. Other forms of covariance functions are discussed in [13, 21].

With the covariance function, the predictive distribution of the output variable y_*, given its input $\mathbf{x}_*$, is Gaussian with mean and variance

$$\begin{aligned} \hat{y}_* &= \mathbf{k}_*^T \mathbf{C}^{-1} \mathbf{y} \\ \sigma_*^2 &= C(\mathbf{x}_*, \mathbf{x}_*) - \mathbf{k}_*^T \mathbf{C}^{-1} \mathbf{k}_* \end{aligned}$$

where $\mathbf{k}_* = [C(\mathbf{x}_*, \mathbf{x}_1), \ldots, C(\mathbf{x}_*, \mathbf{x}_n)]^T$. To obtain the hyper-parameters using maximum likelihood estimation, the log-likelihood of the training data is given by

$$L = -\frac{1}{2} \log \det \mathbf{C} - \frac{1}{2} \mathbf{y}^T \mathbf{C}^{-1} \mathbf{y} - \frac{n}{2} \log 2\pi$$

The derivative of the log-likelihood with respect to each hyper-parameter (denoted by a generic notation θ) is:

$$\frac{\partial L}{\partial \theta} = -\frac{1}{2} \mathrm{tr}\left(\mathbf{C}^{-1} \frac{\partial \mathbf{C}}{\partial \theta}\right) + \frac{1}{2} \mathbf{y}^T \mathbf{C}^{-1} \frac{\partial \mathbf{C}}{\partial \theta} \mathbf{C}^{-1} \mathbf{y}$$

3. Multiple correlated responses

3.1. Model formulation

Let $\mathbf{y}_i$, a column vector, be the q-dimensional response with sample index i, and $\mathbf{x}_i$ be the corresponding p-dimensional covariate vector. The multi-response linear regression model is

$$\mathbf{y}_i = \mathbf{K}_i\mathbf{W} + \boldsymbol{\epsilon}_i, \quad i = 1, \dots, n$$

where $\mathbf{K}_i$ is the Kronecker product of the $q \times q$ identity matrix $\mathbf{I}_q$ and $\mathbf{x}_i^T$: $\mathbf{K}_i = \mathbf{I}_q \otimes \mathbf{x}_i^T$, and $\mathbf{W}$ is the concatenation of the regression vectors for the q response variables: $\mathbf{W} = [\mathbf{w}_1^T, \dots, \mathbf{w}_q^T]^T$. The q-dimensional noise term $\boldsymbol{\epsilon}_i$ is given by a zero mean Gaussian distribution: $\boldsymbol{\epsilon}_i \sim G(\mathbf{0}, \mathbf{S})$, and the correlation between $\boldsymbol{\epsilon}_i$ and $\boldsymbol{\epsilon}_j$, $i \neq j$ is ignored (i.e. the usual assumption of independently and identically distributed noise). The prior for the regression vectors, $\mathbf{W}$, needs to be extended from zero-mean multivariate Gaussian to a zero-mean matrix Gaussian distribution [3, 22]. To simplify the notation, we first define the prior for each regression vector, $\mathbf{w}_g$, as follows:

$$\mathbf{w}_g \sim G(\mathbf{0}, \sigma^2\mathbf{I}_p), \quad g = 1, \dots, q \tag{2}$$

Furthermore, the covariance matrix between regression vectors, is specified as

$$\mathrm{cov}(\mathbf{w}_g, \mathbf{w}_h) = \boldsymbol{\Sigma}^{gh}, \quad g, h = 1, \dots, q$$

which implies $\boldsymbol{\Sigma}^{gg} = \sigma^2\mathbf{I}_p$. We also assume that $\mathbf{W}$ is independent of $\boldsymbol{\epsilon}_i$ $(i = 1, \dots, n)$.

The response vector, $\mathbf{y} = [\mathbf{y}_1^T, \dots, \mathbf{y}_n^T]^T$, a linear function of Gaussian random vector $\mathbf{w}_g$ and Gaussian noise $\boldsymbol{\epsilon}_i$, is itself a Gaussian with zero mean. For ease of derivation the outputs are re-organised hereafter in the following way:

$$\mathbf{y} = [y_{11}, \dots, y_{1n}, \dots, y_{q1}, \dots, y_{qn}]^T \tag{3}$$

which is also Gaussian distributed with zero mean and covariance matrix denoted by $\mathbf{C}$. The entries of $\mathbf{C}$ can be calculated as follows:

$$\begin{aligned}C_{ij}^{gh} &= \mathrm{cov}(y_{gi}, y_{hj}) =< y_{gi}y_{hj}^T > \\ &= < \mathbf{x}_i^T \mathbf{w}_g \mathbf{w}_h^T \mathbf{x}_j > + < \epsilon_{gi}\epsilon_{hj} > \\ &= \mathbf{x}_i^T < \mathbf{w}_g \mathbf{w}_h^T > \mathbf{x}_j + < \epsilon_{gi}\epsilon_{hj} > \\ &= \mathbf{x}_i^T \mathbf{\Sigma}^{gh} \mathbf{x}_j + \delta_{ij} S_{gh} \end{aligned} \tag{4}$$

where $<>$ is the expectation operator, and S_{gh} is the entry of $\mathbf{S}$ at the g-th row and h-th column.

Eq. (4) suggests that the covariance between the i-th sample on the g-th output and the j-th sample on the h-th output may be formulated in two components: the first due to covariance at the inputs, and the second due to that of residual. However, in comparison with the uni-output case, if this covariance function is directly implemented, significantly more hyper-parameters need to be estimated: $\mathbf{\Sigma}^{gh}$ is a $p \times p$ matrix for *each* pair (g, h), and $\mathbf{S}$ is a $q \times q$ matrix. In practice, a more parsimonious model is desired.

In this study, we adopt a concept that when extending from uni- to multi-output linear regression, the covariance function could be parameterised as a "nominal" uni-output covariance as in eq. (2), multiplied by an additional term, b_{gh}, reflecting the covariance between outputs g and h: $\mathbf{\Sigma}^{gh} = b_{gh}\sigma^2\mathbf{I}_p$, and these b_{gh}'s form a $q \times q$ symmetric matrix $\mathbf{B}$. The model is further simplified by setting the noise covariance $\mathbf{S}$ to be diagonal, i.e. $\mathbf{S} = \mathrm{diag}(S_{11}, \ldots, S_{qq})$, since the between-output covariance has already been captured in the first component. The simplified covariance function becomes:

$$C_{ij}^{gh} = b_{gh}\sigma^2 \mathbf{x}_i^T \mathbf{x}_j + \delta_{ij}\delta_{gh}S_{gh}$$

where

$$\delta_{gh} = \begin{cases} 1 & \text{if } g = h \\ 0 & \text{if } g \neq h \end{cases}$$

Furthermore, to mimic the move from a simple linear correlation to more complex covariance function in uni-output case, the following covariance function is adopted by combining bias, linear correlation and exponential terms:

$$C_{ij}^{gh} = \left[a_0 + a_1 \sum_{d=1}^{p} x_{id} x_{jd} + v_0 \exp\left(- \sum_{d=1}^{p} \eta_d (x_{id} - x_{jd})^2 \right) \right] b_{gh} + \delta_{ij} \delta_{gh} S_{gh}$$

If the first part, except b_{gh}'s, in the above equation is organised into a matrix, $\mathbf{Q}$ with entries

$$Q_{ij} \triangleq Q(\mathbf{x}_i, \mathbf{x}_j) = a_0 + a_1 \sum_{d=1}^{p} x_{id} x_{jd} + v_0 \exp\left(- \sum_{d=1}^{p} \eta_d (x_{id} - x_{jd})^2 \right) \tag{5}$$

then the covariance matrix of $\mathbf{y}$ given in (3) becomes the following form:

$$\mathbf{C} = \begin{pmatrix} b_{11}\mathbf{Q} + S_{11}\mathbf{I}_n & b_{12}\mathbf{Q} & \cdots & b_{1q}\mathbf{Q} \\ b_{21}\mathbf{Q} & b_{22}\mathbf{Q} + S_{22}\mathbf{I}_n & \cdots & b_{2q}\mathbf{Q} \\ \vdots & \vdots & \ddots & \vdots \\ b_{q1}\mathbf{Q} & b_{q2}\mathbf{Q} & \cdots & b_{qq}\mathbf{Q} + S_{qq}\mathbf{I}_n \end{pmatrix} \tag{6}$$

or $\mathbf{C} = \mathbf{B} \otimes \mathbf{Q} + \mathbf{S} \otimes \mathbf{I}_n$, where $\otimes$ denotes the Kronecker product. It is known that if $\mathbf{S}$, $\mathbf{B}$ and $\mathbf{Q}$ are all positive definite, so is $\mathbf{C}$.

Similar to the case of a single response, the prediction at a new data point $\mathbf{x}_*$ is a multivariate Gaussian with mean and covariance given by

$$\begin{aligned} \hat{\mathbf{y}}_* &= \mathbf{K}_*^T \mathbf{C}^{-1} \mathbf{y} \\ \mathbf{\Sigma}_* &= \mathbf{B} \otimes Q(\mathbf{x}_*, \mathbf{x}_*) + \mathbf{S} - \mathbf{K}_*^T \mathbf{C}^{-1} \mathbf{K}_* \end{aligned}$$

where $\mathbf{K}_* = \mathbf{B} \otimes \mathbf{Q}_*$ and $\mathbf{Q}_* = [Q(\mathbf{x}_1, \mathbf{x}_*), \ldots, Q(\mathbf{x}_n, \mathbf{x}_*)]^T$ consists of the covariances between the test data point $\mathbf{x}_*$ and the training set as calculated in eq. (5).

The prediction formulae above involve the calculation of the inverse of the covariance matrix $\mathbf{C}$, and the computation is in order of $O(n^3 q^3)$. In many practical applications, the number of response variables is often small, and thus the computation may not pose a serious issue. In addition, if needed, fast approximate methods are available for this calculation; see e.g. [20].

3.2. An alternative derivation of the covariance function

The covariance matrix (6) can alternatively be derived as follows. Let $\mathbf{y} = [y_1, \dots, y_q]$ be a row vector of q-dimensional response, satisfying

$$\mathbf{y} = \mathbf{f}(\mathbf{x}) + \boldsymbol{\epsilon}$$

where $\mathbf{x} = [x_1, \dots, x_p]$ is a p-dimensional covariate vector, $\mathbf{f} = [f_1, \dots, f_q]$ is a vector valued random function, and $\boldsymbol{\epsilon} = [\epsilon_1, \dots, \epsilon_q] \sim G(\mathbf{0}, \mathbf{S})$ is independent random noise, i.e. $\mathbf{S}$ is diagonal. Assume $\mathbf{f} \sim G_q(\mathbf{0}, \mathbf{B})$: q-variate Gaussian distribution with mean $\mathbf{0}$ and covariance matrix $\mathbf{B}$.

Suppose we have n samples $(\mathbf{x}_1, \mathbf{y}_1), \dots, (\mathbf{x}_n, \mathbf{y}_n)$, and assume that, for any output y_g $(g = 1, \dots, q)$, the corresponding random function $f_g(\cdot)$ satisfies a GP with covariance function

$$\begin{aligned}\operatorname{cov}(f_g(\mathbf{x}_i), f_g(\mathbf{x}_j)) &= Q(\mathbf{x}_i, \mathbf{x}_j) \\ &= a_0 + a_1 \sum_{d=1}^{p} x_{id}x_{jd} + v_0 \exp\left(-\sum_{d=1}^{p} \eta_d (x_{id} - x_{jd})^2\right) \triangleq Q_{ij}\end{aligned}$$

Let $\mathbf{Q} = (Q_{ij})_{n\times n}$. Then the $n \times q$ matrix $[\mathbf{f}^T(\mathbf{x}_1), \dots, \mathbf{f}^T(\mathbf{x}_n)]^T$ has a matrix normal distribution $MN(\mathbf{0}, \mathbf{Q}, \mathbf{B})$, where $\mathbf{Q}$ is the covariance among rows and $\mathbf{B}$ is the one among columns. Equivalently, $\operatorname{vec}[\mathbf{f}^T(\mathbf{x}_1), \dots, \mathbf{f}^T(\mathbf{x}_n)]^T \sim G_{nq}(\mathbf{0}, \mathbf{B} \otimes \mathbf{Q})$: nq-variate Gaussian distribution. Hence $\operatorname{vec}[\mathbf{y}_1^T, \dots, \mathbf{y}_n^T]^T \sim G_{nq}(\mathbf{0}, \mathbf{B} \otimes \mathbf{Q} + \mathbf{S} \otimes \mathbf{I}_n)$, the same as (6). Note that the above derivation implies that the matrix $\mathbf{B}$ in (6) must be positive definite.

This formulation also explains the proposed multiple response model. In fact, it is assumed that the noise free q-dimensional response variable $\mathbf{f} = [f_1, \dots, f_q]$ follows a

q-variate Gaussian distribution $G_q(\mathbf{0}, \mathbf{B})$. If the random samples of $\mathbf{f}$ at $\mathbf{x}_1, \ldots, \mathbf{x}_n$, namely $\mathbf{f}(\mathbf{x}_1), \ldots, \mathbf{f}(\mathbf{x}_n)$, were independent, the matrix $[\mathbf{f}^T(\mathbf{x}_1), \ldots, \mathbf{f}^T(\mathbf{x}_n)]^T$ would have a matrix normal distribution $MN(\mathbf{0}, \mathbf{\Sigma}, \mathbf{B})$ where $\mathbf{\Sigma}$ is a diagonal matrix depending on $\mathbf{x}_1, \ldots, \mathbf{x}_n$. The proposed model effectively addresses the problem of correlated samples $\mathbf{f}(\mathbf{x}_1), \ldots, \mathbf{f}(\mathbf{x}_n)$ in the similar way as the univariate Gaussian process regression, by replacing the diagonal matrix $\mathbf{\Sigma}$ by $\mathbf{Q}$ which is calculated from the covariance function.

3.3. Parameter estimation

Estimation of hyper-parameters becomes to maximize the log likelihood whilst observing the constraint that $\mathbf{C}$ must be positive definite, or equivalently $\mathbf{S}$, $\mathbf{B}$ and $\mathbf{Q}$ are all positive definite. This can be done by the *block coordinate descent*, also termed *non-linear Gaussian-Seidel* method [1]. That is, alternate between the two steps: (i) estimate the usual GPR hyper-parameters $(a_0, a_1, \eta_1, \ldots, \eta_p, v_0)$ and the elements of $\mathbf{S}$ $(S_{11}, \ldots, S_{qq})$ by fixing $\mathbf{B}$, and (ii) estimate $\mathbf{B}$ by fixing the hyper-parameters and $\mathbf{S}$, where in both steps the objective function is the following log-likelihood:

$$L = -\frac{1}{2}\log\det\mathbf{C} - \frac{1}{2}\mathbf{y}^T\mathbf{C}^{-1}\mathbf{y} - \frac{nq}{2}\log 2\pi$$

The estimation of the hyper-parameters in $\mathbf{Q}$ and $\mathbf{S}$ follows the same strategy as in the case of a single response discussed at the end of the previous section. The only difference is that related to the derivatives of the covariance matrix with respect to a hyper-parameter (denoted by a generic notation θ) and S_{gg} $(g = 1, \ldots, q)$, which are

$$\begin{aligned}
\frac{\partial\mathbf{C}}{\partial\theta} &= \frac{\partial\left(\mathbf{S}\otimes\mathbf{I}_n + \mathbf{B}\otimes\mathbf{Q}\right)}{\partial\theta} = \frac{\partial\left(\mathbf{B}\otimes\mathbf{Q}\right)}{\partial\theta} = \mathbf{B}\otimes\frac{\partial\mathbf{Q}}{\partial\theta}\\
\frac{\partial\mathbf{C}}{\partial S_{gg}} &= \frac{\partial\left(\mathbf{S}\otimes\mathbf{I}_n + \mathbf{B}\otimes\mathbf{Q}\right)}{\partial S_{gg}} = \frac{\partial\left(\mathbf{S}\otimes\mathbf{I}_n\right)}{\partial S_{gg}} = \mathbf{E}_{gg}\otimes\mathbf{I}_n
\end{aligned}$$

where $\mathbf{E}_{gg}$ is the $q \times q$ elementary matrix having unity in the (g, g)-th element and zeros elsewhere. $\partial\mathbf{Q}/\partial\theta$ depends on the form of the covariance function and can be derived accordingly [21]. In practice, since these hyper-parameters must be positive to ensure the positive definiteness of $\mathbf{Q}$ and $\mathbf{S}$, the hyper-parameters are log-transformed before estimation, a common strategy to convert the constrained optimisation problem into an unconstrained one.

The estimation of the positive definite matrix $\mathbf{B}$, again, is a constrained optimisation problem. In fact, this can be cast into a semidefinite programming (SDP) problem, for which efficient solution methods are available especially for linear cases [25]. Unfortunately, the objective function (the log-likelihood) is a non-linear function of $\mathbf{B}$, and thus to be able to use SDP algorithms, the log-likelihood must be linearised iteratively during the solution process. Such a strategy is complex to implement, and may converge slowly due to the approximate linearisation used.

We instead adopt a more straightforward approach to transforming the constrained problem to an unconstrained one. A common method is to utilize the Cholesky decomposition [19]. Let $\mathbf{B} = \mathbf{\Phi}\mathbf{\Phi}^T$ where $\mathbf{\Phi}$ is a lower triangular matrix such that

$$\mathbf{\Phi} = \begin{pmatrix} \phi_{11} & 0 & \cdots & 0 \\ \phi_{21} & \phi_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \phi_{q1} & \phi_{q2} & \cdots & \phi_{qq} \end{pmatrix}$$

To guarantee the uniqueness of $\mathbf{\Phi}$, we require its diagonal elements to be positive and denote $\psi_{ii} = \log(\phi_{ii})$ for $i = 1, \ldots, q$. Consequently the matrix $\mathbf{B}$ is reparameterized by $[\psi_{11}, \phi_{21}, \psi_{22}, \ldots, \psi_{qq}]^T$. Hence the derivatives of the covariance matrix $\mathbf{C}$ with respect to these parameters are as follows: for $i, j = 1, \ldots, q$ and $i > j$

$$\frac{\partial \mathbf{C}}{\partial \phi_{ij}} = \frac{\partial \left(\mathbf{B} \otimes \mathbf{Q}\right)}{\partial \phi_{ij}} = \frac{\partial \left(\mathbf{\Phi}\mathbf{\Phi}^T\right)}{\partial \phi_{ij}} \otimes \mathbf{Q} = \left(\mathbf{E}_{ij}\mathbf{\Phi}^T + \mathbf{\Phi}\mathbf{E}_{ji}\right) \otimes \mathbf{Q}$$
$$\frac{\partial \mathbf{C}}{\partial \psi_{ii}} = \frac{\partial \left(\mathbf{B} \otimes \mathbf{Q}\right)}{\partial \psi_{ii}} = \frac{\partial \left(\mathbf{\Phi}\mathbf{\Phi}^T\right)}{\partial \psi_{ii}} \otimes \mathbf{Q} = \left(\mathbf{J}_{ii}\mathbf{\Phi}^T + \mathbf{\Phi}\mathbf{J}_{ii}\right) \otimes \mathbf{Q}$$

where $\mathbf{E}_{ij}$ is the $q \times q$ elementary matrix having unity in the (i, j)-th element and zeros elsewhere, and $\mathbf{J}_{ii}$ is the same as $\mathbf{E}_{ii}$ but with the unity being replaced by $e^{\psi_{ii}}$.

4. Extension

The multi-response model discussed in the previous section assumes that all the outputs are observed at the same covariate values. We now extend the model to the case where different outputs may be observed at different covariate values.

Suppose that the gth output has n_g observations $y_{g1}, \ldots, y_{gn_g}$ at the corresponding covariate values $\mathbf{x}_{g1}, \ldots, \mathbf{x}_{gn_g}$. We can define the covariance between y_{gi} and y_{hj} as

$$C_{ij}^{gh} = \left[a_0 + a_1 \sum_{d=1}^{p} x_{gid} x_{hjd} + v_0 \exp\left(- \sum_{d=1}^{p} \eta_d (x_{gid} - x_{hjd})^2 \right) \right] b_{gh} + \delta_{ij} \delta_{gh} S_{gh}$$

for $g, h = 1, \ldots, q$, $i = 1, \ldots, n_g$ and $j = 1, \ldots, n_h$.

Letting

$$Q_{ghij} \triangleq Q(\mathbf{x}_{gi}, \mathbf{x}_{hj}) = a_0 + a_1 \sum_{d=1}^{p} x_{gid} x_{hjd} + v_0 \exp\left(- \sum_{d=1}^{p} \eta_d (x_{gid} - x_{hjd})^2 \right)$$

and $\mathbf{Q}_{gh} = (Q_{ghij})_{n_g \times n_h}$, then the covariance matrix of $\mathbf{y} = [y_{11}, \ldots, y_{1n_1}, \ldots, y_{q1}, \ldots, y_{qn_q}]^T$ becomes the following form:

$$\mathbf{C} = \begin{pmatrix} b_{11}\mathbf{Q}_{11} + S_{11}\mathbf{I}_{n_1} & b_{12}\mathbf{Q}_{12} & \ldots & b_{1q}\mathbf{Q}_{1q} \\ b_{21}\mathbf{Q}_{21} & b_{22}\mathbf{Q}_{22} + S_{22}\mathbf{I}_{n_2} & \ldots & b_{2q}\mathbf{Q}_{2q} \\ \vdots & \vdots & \ddots & \vdots \\ b_{q1}\mathbf{Q}_{q1} & b_{q2}\mathbf{Q}_{q2} & \ldots & b_{qq}\mathbf{Q}_{qq} + S_{qq}\mathbf{I}_{n_q} \end{pmatrix}$$

where $\mathbf{Q}_{gh} = \mathbf{Q}_{hg}^T$. Let $\mathbb{Q}$ be a $q \times q$ block matrix with the block elements $\mathbf{Q}_{gh}$ and $\mathbb{I}$ a $q \times q$ block identity matrix with the block elements $\mathbf{I}_{n_g}$, then $\mathbf{C} = \mathbf{B} \circ \mathbb{Q} + \mathbf{S} \circ \mathbb{I}$, where $\circ$ denotes the block Hadamard product. Apparently $\mathbb{Q}$ is a positive definite matrix, and since $\mathbf{B}$ is also positive definite, so is $\mathbf{B} \circ \mathbb{Q}$ by the Schur Product Theorem [10]. Hence $\mathbf{C}$ is positive definite.

The parameter estimation can be done along the same line as discussed in the previous section. The derivatives needed are:

$$
\begin{aligned}
\frac{\partial \mathbf{C}}{\partial \theta} &= \mathbf{B} \circ \frac{\partial \mathbb{Q}}{\partial \theta} \\
\frac{\partial \mathbf{C}}{\partial S_{gg}} &= \mathbf{E}_{gg} \circ \mathbb{I} \\
\frac{\partial \mathbf{C}}{\partial \phi_{ij}} &= \left(\mathbf{E}_{ij}\boldsymbol{\Phi}^T + \boldsymbol{\Phi}\mathbf{E}_{ji}\right) \circ \mathbb{Q} \\
\frac{\partial \mathbf{C}}{\partial \psi_{ii}} &= \left(\mathbf{J}_{ii}\boldsymbol{\Phi}^T + \boldsymbol{\Phi}\mathbf{J}_{ii}\right) \circ \mathbb{Q}
\end{aligned}
$$

The prediction at new data points $[\mathbf{x}_{1*}, \ldots, \mathbf{x}_{q*}]$ is a multivariate Gaussian with mean and covariance given by

$$
\begin{aligned}
\hat{\mathbf{y}}_* &= \mathbf{K}_*^T \mathbf{C}^{-1} \mathbf{y} \\
\boldsymbol{\Sigma}_* &= \mathbf{B} \circ \mathbf{Q}_{**} + \mathbf{S} - \mathbf{K}_*^T \mathbf{C}^{-1} \mathbf{K}_*
\end{aligned}
$$

where $\mathbf{Q}_{**}$ is a $q \times q$ matrix representing the covariances between the test points and themselves with elements $Q_{**}^{gh} = Q(\mathbf{x}_{g*}, \mathbf{x}_{h*})$, and $\mathbf{K}_* = \mathbf{B} \circ \mathbb{Q}_*$ where the block matrix $\mathbb{Q}_*$ has block elements $\mathbf{Q}_*^{gh}$ which consists of the covariances between the training data points of the gth output and the test data point of the hth output.

5. Numerical examples

In this section we demonstrate the effectiveness of the proposed model using some numerical examples, including simulated data and real data.

5.1. Simulated examples

We first consider a simulated data from bivariate analytical functions. The true model used to generate data is given by

$$y_1 = f_1(x_1, x_2) + \epsilon_1, \text{ with } f_1(x_1, x_2) = 3\cos(x_1) + 4\cos(2x_2)$$

$$y_2 = f_2(x_1, x_2) + \epsilon_2, \text{ with } f_2(x_1, x_2) = 2\cos(x_1 + 1.0) + 3\cos(2x_2 + 1.0)$$

where $\epsilon_1 \sim G(0, 0.25)$ and $\epsilon_2 \sim G(0, 0.16)$. The covariates x_1 and x_2 both have 20 equally spaced values in $[-5, 5]$ so that a sample of 20 observations for y_1 and y_2 are generated, which gives the sample correlation between the two response variables 0.527.

Table 1: The RMSE for data from bivariate analytical functions

	Output 1 (y_1)			Output 2 (y_2)		
	Multi-GP	Ind-GP	PLS	Multi-GP	Ind-GP	PLS
With observed	0.384	0.438	3.984	0.627	0.708	2.603
With true	0.312	0.350	3.817	0.482	0.515	2.730

To test the performance of the model, leave-one-out cross validation is performed, that is, each of the 20 data points is left as test data whilst the remaining data are used for model training. The predicted values are then compared with the observed as well as the true ones calculated from $f_1(x_1, x_2)$ and $f_2(x_1, x_2)$, and the root mean square errors (RMSE) are presented in Table 1. For comparison, the conventional uni-output Gaussian process regressions and the widely used partial least squares regression for multi-inputs and multi-outputs (PLS) are also performed, where the former is conducted for the two outputs independently, without considering their correlation. The table shows that the proposed model (multi-GP) which takes the correlation between the responses into account significantly improves the prediction accuracy compared with the method of modelling each output independently (ind-GP). PLS is essentially a linear model so it is not surprising that it fails to make sensible predictions for these highly nonlinear functions.

If there is little or no correlation among the responses the multi-GP can not borrow information from other outputs. In this case it is understandable that the multi-GP may not improve the prediction accuracy or even perform worse than the ind-GP, due to the fact that the former actually imposes more constraints on the covariance functions than the latter. This feature is also demonstrated by the following example. The same experimental scheme as above is conducted, with $f_1(x_1, x_2)$ and $f_2(x_1, x_2)$ being defined as

$$f_1(x_1, x_2) = 2\cos(x_1 + 0.5) + 3\cos(2x_2 + 0.5), \quad f_2(x_1, x_2) = 0.5x_1 + x_2$$

Table 2: The RMSE for data from bivariate analytical functions with no correlation

	Output 1 (y_1)		Output 2 (y_2)	
	Multi-GP	Ind-GP	Multi-GP	Ind-GP
With observed	1.119	1.016	0.448	0.334
With true	0.660	0.590	0.236	0.275

The sample correlation between the two response variables is -0.012. The results are reported in Table 2 which shows that the multi-GP does not perform as well as the ind-GP in terms of prediction accuracy, as expected.

The second simulated example is to test the model for the scenarios where different responses may be observed at different covariate values. The data is generated by the following true model:

$$y_1 = f_1(x) + \epsilon_1, \text{ with } f_1(x) = 3\cos(x)$$

$$y_2 = f_2(x) + \epsilon_2, \text{ with } f_2(x) = 2\cos(x + 0.3)$$

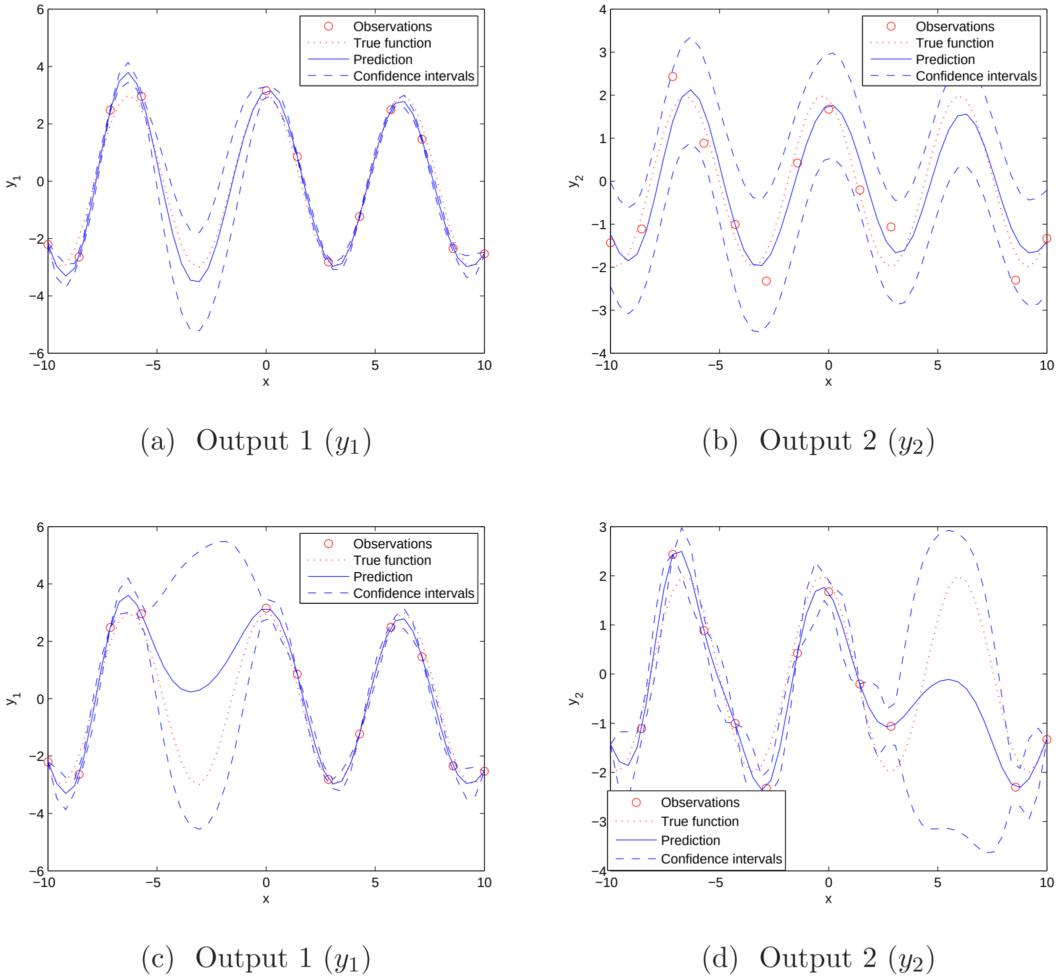
where $\epsilon_1 \sim G(0, 0.25)$ and $\epsilon_2 \sim G(0, 0.25)$. The covariate x has 15 equally spaced values in $[-10, 10]$ so that a sample of 15 observations for y_1 and y_2 are obtained with the sample correlation 0.921.

For model training, the data points for x in $[-5, -1]$ are removed from the first output y_1 and those in $[4, 8]$ removed from y_2. The prediction is then performed at all 50 covariate values equally spaced in $[-10, 10]$. The RMSEs between the predicted values and the true ones obtained from $f_1(x)$ and $f_2(x)$ are calculated. The uni-output GPR models are also applied to the same data for y_1 and y_2 independently. The results are presented in Table 3 and Figure 1.

From the table and the figure it can be seen that, at the intervals with relatively dense data points, multi-GP and ind-GP give comparable results; however, in the regions where some data points are missing, multi-GP is able to learn the patterns of

Table 3: The RMSE for outputs observed at different covariate values

Output 1 (y_1)		Output 2 (y_2)	
Multi-GP	Ind-GP	Multi-GP	Ind-GP
0.376	1.101	0.447	0.833



(a) Output 1 (y_1)

(b) Output 2 (y_2)

(c) Output 1 (y_1)

(d) Output 2 (y_2)

Figure 1: Predictions for outputs observed at different covariate values. Top panel: predictions by multi-GP; bottom panel: predictions by ind-GP. The solid lines are the predictions, the dotted lines are the true functions and the circles are the observations. The dashed lines represent the 95% confidence intervals.

the true functions from each other so can fill in the gap whilst ind-GP fails to make good predictions since it does not get any information from the other output. These results clearly illustrate the advantage of the proposed model which accommodates the correlation between outputs. It is also notable that the prediction variances of the multi-GP are much smaller than those of the ind-GP in the regions with missing data. This is another advantage of the proposed model. The prediction uncertainty of GPR can be used in constructing the prediction model by ensemble learning [12].

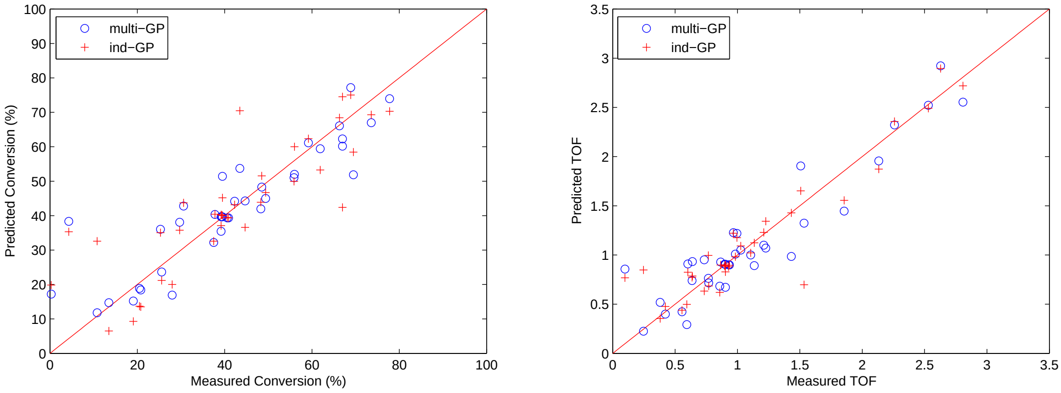
5.2. Modelling the response surface of a catalytic oxidation process

Oxidation of alcohols into the corresponding aldehydes or ketones, in particular benzyl alcohol to benzaldehyde, is one of the most important functional group transformations in organic synthesis. The selected catalyst, K-Mn/C, was prepared by co-impregnating aqueous solutions of potassium and manganese nitrates onto commercially available activated carbon. The catalytic oxidation process was conducted in a bath-type lab-scale reactor. More experimental details can be found in [23]. Experiments were conducted to study the impact of five process factors (reaction temperature, partial pressure of oxygen, concentration of benzyl alcohol in terms of mmol diluted within 10 ml of toluene, percentage of Mn, and K:Mn ratio) on the conversion of benzyl alcohol, and the turn over frequency (TOF). The conversion and TOF are regarded as the process response variable. It should be noted that the conversion and TOF are highly correlated (correlation coefficient 0.65), suggesting that a multi-response GPR model could be useful.

The original purpose of the experiments was to develop a quadratic regression model to relate the conversion to the five process factors. Hence, the central composite design, which is especially appropriate for quadratic regression, was adopted to give 32 experimental runs. In a later stage, an additional six experiments were conducted to further confirm the effect of increasing K:Mn ratio. Therefore, the data set is not the result of rigorously designed experiments, which is not uncommon in practical experimentation. The data for the 38 experimental runs have been published in [23].

Table 4: The RMSE for catalytic oxidation process data

Conversion		TOF	
Multi-GP	Ind-GP	Multi-GP	Ind-GP
8.66	10.75	0.31	0.33



(a) Conversion

(b) TOF

Figure 2: Prediction by leave-one-out cross validation. 'o': by multi-GP; '+': by ind-GP.

To test the effectiveness of the proposed model on this dataset, leave-one-out cross validation is performed using the multiple response model (multi-GP) and the independent GPR (ind-GP). The RMSEs between the predicted and the measured values are given in Table 4, and the predictions by both methods are illustrated in Figure 2. It is obvious that the multiple response model indeed improves the accuracy of prediction, particularly for the conversion.

Furthermore, the dataset is also used to test the proposed model for the case where different outputs may be observed at different covariate values using the following scheme. For model training we randomly select N out of 38 data points for the two responses independently, which means that at some covariate values only the conversion or TOF may be measured. The trained model is then used to make predictions on the

Table 5: The RMSE for catalytic oxidation process data

N	Conversion		TOF	
	Multi-GP	Ind-GP	Multi-GP	Ind-GP
10	14.93	16.06	0.42	0.43
15	11.76	12.99	0.33	0.34
20	9.80	12.21	0.31	0.35
30	6.62	8.07	0.18	0.26

remaining data points. The independent GPR model is also applied to the same data for comparison and the RMSEs between the predictions and the measurements are calculated. Table 5 presents the average RMSEs based on ten replications using the above scheme for $N = 10$, 15, 20 and 30.

It is obvious that with the increase of the number of training points, the predictions by both methods become more accurate. However, the multi-GP consistently outperforms the independent GP in all cases. It can also be observed that the improvement by multi-GP appears to be more significant as the number of training points increases. A possible explanation is that the proposed model can learn better on the correlations among different outputs with more data points.

6. Concluding remarks

We proposed a direct formulation of the covariance function for multiple response GPR, based on the idea that its covariance function was assumed to be the "nominal" uni-output covariance multiplied by the covariances between different outputs. The parameters were estimated by the *block coordinate descent* in which the estimation of the between-output covariance matrix $\mathbf{B}$ was implemented using Cholesky decomposition. The superiority of the proposed multi-response GPR method over the independent GPR was demonstrated through the simulated examples and the response surface modelling of a catalytic reaction process. The model was also extended to the scenarios where different responses may be observed at different covariate values.

Unlike the linear filtering method [2] where the covariance structure among different outputs is assumed to follow some specific forms, the proposed model is able to learn these dependencies from data, represented by the matrix $\mathbf{B}$. In this paper we assumed that the covariance functions for GP within each output were the same. In practice cases exist where it may be better to use different covariance functions for different response variables. It is however difficult to define the covariances between data points coming from different outputs in this setting, and this problem will be further investigated.

Acknowledgment

BW thanks the University of Leicester for the support in granting him Academic Study Leave during which this work was completed. The authors thank the reviewers for their constructive suggestions and helpful comments.

References

[1] D. Bertsekas, Nonlinear Programming, Athena Scientific, 1999.

[2] P. K. Boyle, M. R. Frean, Dependent Gaussian processes, in: Advances in Neural Information Processing Systems 17, MIT Press, 2004.

[3] P. J. Brown, Measurement, Regression, and Calibration, Oxford University Press, 1993.

[4] C. J. C. Burges, A tutorial on support vector machines for pattern recognition, Data Mining and Knowledge Discovery 2 (1998) 121–167.

[5] L. L. T. Chan, Y. Liu, J. Chen, Nonlinear system identification with selective recursive Gaussian process models, Industrial & Engineering Chemistry Research 52 (2013) 18276–18286.

[6] T. Chen, J. Morris, E. Martin, Gaussian process regression for multivariate spectroscopic calibration, Chemometrics and Intelligent Laboratory Systems 87 (2007) 59–67.

[7] T. Chen, K. Hadinoto, W.J. Yan, Y.F. Ma, Efficient meta-modelling of complex process simulations with time-space-dependent outputs, Computers and Chemical Engineering 35 (2011) 502–509.

[8] J. P. Chiles, P. Delfiner, Geostatistics, Modeling Spatial Uncertainty, Wiley Series in Probability and Statistics, 1999.

[9] T. Choi, J.Q. Shi, B. Wang, A Gaussian process regression approach to single index model, Journal of Nonparametric Statistics 23 (2011) 21–36.

[10] M. Güther, L. Klotz, Schur's theorem for a block Hadamard product, Linear Algebra and its Applications 437 (2012) 948–956.

[11] B. Likar, J. Kocijan, Predictive control of a gas-liquid separation plant based on a Gaussian process model, Computers and Chemical Engineering 31 (2007) 142–152.

[12] Y. Liu, Z. Gao, Real-time property prediction for an industrial rubber-mixing process with probabilistic ensemble GPR models, Journal of Applied Polymer Science 132 (2015) 1905–1913.

[13] D. J. C. MacKay, Introduction to Gaussian processes, in: C. M. Bishop (Ed.), Neural Networks and Machine Learning, volume 168 of F: Computer and Systems Sciences, NATO Advanced Study Institute, Springer, Berlin, Heidelberg, 1998, pp. 133–165.

[14] R. M. Neal, Bayesian Learning for Neural Networks, Springer-Verlag, New York, 1996.

[15] W. Ni, S. Tan, W. Ng, Recursive GPR for nonlinear dynamic process modeling, Chemical Engineering Journal 173 (2011) 636–643.

[16] W. Ni, L. Norgaard, M. Morup, Non-linear calibration models for near infrared spectroscopy, Analytica Chimica Acta 813 (2014) 1–14.

[17] A. O'Hagan, Curve fitting and optimal design for prediction, Journal of the Royal Statistical Society Series B-Methodological 40 (1) (1978) 1–42.

[18] M. Osborne, S. Roberts, A. Rogers, S. Ramchurn, N. Jennings, Towards real-time information processing of sensor network data using computationally efficient multi-output Gaussian processes, in: The 7th International Conference on Information Processing in Sensor Networks, 2008, pp. 109–120.

[19] J. C. Pinheiro, D. M. Bates, Unconstrained parameterizations for variance-covariance matrices, Statistics and Computing 6 (1996) 289–296.

[20] J. Quinonero-Candela, C.E. Rasmussen, A unifying view of sparse approximate Gaussian process regression, Journal of Machine Learning Research 6 (2005) 1939–1959.

[21] C. E. Rasmussen, C. K. I. Williams, Gaussian Processes for Machine Learning, MIT Press, 2006.

[22] N. Sha, Discussion of "Gaussian process regression for multivariate spectroscopic calibration", Chemometrics and Intelligent Laboratory Systems 87 (2007) 93–94.

[23] Q. Tang, Y. Chen, C. Zhou, T. Chen, Y. Yang, Statistical modelling and analysis of the aerobic oxidation of benzyl alcohol over K–Mn/C catalysts, Catalysis Letters 128 (2009) 210–220.

[24] M. E. Tipping, Sparse Bayesian learning and the relevance vector machine, Journal of Machine Learning Research 1 (2001) 211–244.

[25] K. Toh, M. Todd, R. Tütüncü, SDPT3 - a Matlab software package for semidefinite programming, Optimization Methods and Software 11/12 (1999) 545–581.

[26] K. Wang, T. Chen, R. Lau, Bagging for robust non-linear multivariate calibration of spectroscopy, Chemometrics and Intelligent Laboratory Systems 105 (2011) 1–6.

[27] J. Yuan, K. S. Wang, T. Yu, M. L. Fang, Reliable multi-objective optimization of high-speed WEDM process based on Gaussian process regression, International Journal of Machine Tools & Manufacture 48 (2008) 47–60.

List of Figures

List of Tables