Chemometric Calibration of Surface enhanced Raman Spectra for Quantitative Analysis

Xiaofei Shi
xashi@mix.wvu.edu

Follow this and additional works at: https://researchrepository.wvu.edu/etd

Part of the Operational Research Commons

Recommended Citation
https://researchrepository.wvu.edu/etd/7443

This Dissertation is protected by copyright and/or related rights. It has been brought to you by the The Research Repository @ WVU with permission from the rights-holder(s). You are free to use this Dissertation in any way that is permitted by the copyright and related rights legislation that applies to your use. For other uses you must obtain permission from the rights-holder(s) directly, unless additional rights are indicated by a Creative Commons license in the record and/or on the work itself. This Dissertation has been accepted for inclusion in WVU Graduate Theses, Dissertations, and Problem Reports collection by an authorized administrator of The Research Repository @ WVU. For more information, please contact researchrepository@mail.wvu.edu.
Chemometric Calibration of Surface enhanced Raman Spectra for Quantitative Analysis

Xiaofei Shi

Follow this and additional works at: https://researchrepository.wvu.edu/etd

Part of the Operational Research Commons
Chemometric Calibration of Surface-enhanced Raman Spectra for
Quantitative Analysis

Xiaofei Shi

A dissertation submitted to the
Benjamin M. Statler College of Engineering and Mineral Resources
at West Virginia University
in partial fulfillment of the requirements
for the degree of

Doctor of Philosophy
in
Industrial Engineering

Feng Yang, Ph.D., Chair
Nianqiang Wu, Ph.D.
Bhaskaran Gopalakrishnan, Ph.D.
Xinjian He, Ph.D.
Xi Chen, Ph.D.

Department of Industrial and Management Systems Engineering

Morgantown, West Virginia
2019

Keywords: Chemometrics, Kriging, Design of experiments, Inverse modeling, Uncertainty quantification

Copyright 2019 Xiaofei Shi
ABSTRACT

Chemometric Calibration of Surface-enhanced Raman Spectra for Quantitative Analysis

Xiaofei Shi

Raman spectroscopy, recognized as a powerful analytical technique, has been widely employed in many fields especially in the detection of hazard material containing in food or water. However, multiplexed analyte quantification based on Raman spectra remains a challenge due to the difficulties in accurate and precise modeling of the relationship between multiple analyte concentrations and dense spectral data with noise.

In this work, a statistical procedure was developed to efficiently generate high-quality calibration models quantifying the analyte concentrations versus spectra relationship. The resulting calibration models are able to provide estimated concentration ranges (which reflect both point and uncertainty estimates) for the analytes of interest in an unknown sample based on its observed Raman spectrum. The calibration procedure integrates three unique methodology components. (i) Stochastic kriging with time-series errors was adapted to model Raman spectra as a function on analyte concentrations. (ii) Built on the kriging modeling, bootstrap resampling methods were adapted to quantify the uncertainty of analyte concentration estimates. (iii) Based on the uncertainty quantification capability, a two-stage experimental design method was developed for efficient sampling: How to use a minimum amount of experimental effort to achieve calibration models with desired uncertainty of analyte estimates?

Simulation studies were derived from laboratory experimental data, and used to demonstrate the efficiency of the calibration procedure over the methods.
ACKNOWLEDGMENTS

I would like to thank my advisor, Dr. Feng Yang for her great guidance and continuous support throughout my study at West Virginia University. It has been a great honor to be her Ph.D. student. Without her inspiration, it wouldn’t be possible for me to finish my Ph.D. degree. I am also thankful to Dr. Nianqiang Wu, Dr. Bhaskaran Gopalakrishnan, Dr. Jianxin He and Dr. Xi Chen for serving on my committee, and for their insightful ideas and assistance in preparing this dissertation.
Contents

List of Figures vi

List of Tables vii

List of Acronyms viii

1 Introduction 1
  1.1 Background ................................................................. 1
  1.2 Literature Review .......................................................... 2
  1.3 Overview of Methodologies .............................................. 2
  1.4 Contribution ................................................................. 3

2 Data Description 6

3 Calibration Modeling 8
  3.1 Background Removal ...................................................... 8
  3.2 Forward and Inverse Directions ........................................ 8
  3.3 Statistical Modeling ........................................................ 9
    3.3.1 Forward Modeling .................................................... 10
    3.3.2 Inverse Modeling ..................................................... 17
  3.4 Quantifying Estimation Uncertainty ...................................... 18

4 Design of Experiments 23
  4.1 Design Space ............................................................... 24
## List of Figures

1.1 Overview of the calibration procedure. ........................................ 4  
1.2 The objective of the calibration procedure ................................. 5  
2.1 An example of Raman spectrum .................................................. 6  
2.2 Spectrum decomposition ........................................................... 7  
3.1 Polynomial curve fitting result: (a) the fitted background after the first iteration; (b) the fitted background after convergence. .................. 10  
5.1 Prediction model of peak-based method ...................................... 34  
5.2 ERMSE box plots in case 1 ....................................................... 35  
5.3 The location of design points: circles represent initial design points; diamonds represent added design points. .............................. 38  
5.4 VDS-based evaluation results for one macro-replication ................ 40  
5.5 VDS-based evaluation results for 100 macro-replications ............... 41  
A.1 The spectrum before and after Savitzky-Golay filtering ............... 49  
A.2 Noise v.s. Raman shift ............................................................ 50  
A.3 Simulating signal ...................................................................... 52
List of Tables

5.1 The adapted SK parameters of the forward model based on initial design . . 31
5.2 The DK parameters of the inverse model based on initial design . . . . . 31
5.3 The DK parameters of the inverse model based on final design . . . . . . 32
5.4 Comparison of point estimator and uncertainty . . . . . . . . . . . . . . . 36
5.5 PI coverage probabilities of two procedures . . . . . . . . . . . . . . . . 36
5.6 The DK parameters of the inverse model based on initial design for case 2 . 38
5.7 Coverage probability in case 2 . . . . . . . . . . . . . . . . . . . . . . . . . . 41

A.1 Concentration combinations . . . . . . . . . . . . . . . . . . . . . . . . . . 47
A.2 The DK parameters of the background simulation model . . . . . . . . . . 49
A.3 Lorentzian function parameters . . . . . . . . . . . . . . . . . . . . . . . . 52
List of Acronyms

**SERS** Surface Enhanced Raman Spectroscopy

**DOE** Design of Experiments

**SK** Stochastic Kriging

**DK** Deterministic Kriging

**VDS** Validation Data Set

**PI** Prediction Interval

**PCA** Principal Component Analysis

**PLS** Partial Least Squares

**IID** Independent and Identically Distributed
Chapter 1

Introduction

1.1 Background

Raman spectroscopy, recognized as a powerful analytical technique, has been widely employed in many fields such as physics, chemistry, biology, and biomedicine [1, 2, 3, 4, 5]. Raman signals are generated by the inelastic scattering of the incident light from a sample and are also known as Raman spectrum, which enables both qualitative and quantitative analysis [6, 7]. The characteristic fingerprinting pattern in a Raman spectrum renders the possibility to identify substances and quantify analyted concentrations [8, 9, 10, 11]. For example, in a most recent study, Raman spectrum was applied to measure the concentrations of cocaine, cannabis and alcohol from human saliva [12, 13].

The key to use Raman spectr for analyte quantifications lies in establishing the analyte concentrations versus spectra relationship, which is referred to as the calibration model. It remains a challenge to obtain such calibration models of desired quality (or controlled uncertainty) due to the following reasons. First, a Raman spectrum is a highly-dense data series involving noises that are not independent or identically distributed. Second, the co-existence of both forward and inverse directions in a typical calibration problem makes standard statistical methods invalid for quantifying the uncertainty of target estimates (i.e., analyte concentrations). Third, for simultaneous quantification of multiple analytes, it becomes critical to most efficiently sample calibration data in the high-dimensional concentration space of multiple analytes. These difficulties are far from being adequately addressed in the existing literature of calibrating Raman signals.
1.2 Literature Review

Substantial efforts have been made in the literature regarding the calibration of Raman signals, which can be roughly divided into two streams: peak-based methods [11, 14] and dimension reduction methods [15, 9, 16]. Peak-based methods employ only one peak in a spectrum (the characteristic peak) at a specified Raman shift (or wave number), and estimate the peak intensity vs. analyte concentration relationship for calibration. Such methods have obvious disadvantages: only a small piece of Raman spectrum is used, and it is often not clear which is the characteristic peak. Peak-based methods are not able to handle multiplexed analyte quantification. Dimension reduction methods, such as partial least squares (PLS) regression [17, 18], seek to utilize the entire spectrum to build the calibration model. These methods do not consider statistical inference (uncertainty quantification) issues, which serve as the basis of DOE.

In light of the discussions above, this work developed a statistical calibration procedure which uses a least amount of sampling effort to calibration Raman signals with desired precision (pre-specified uncertainty levels).

1.3 Overview of Methodologies

The calibration model is estimated from a set of experimental data denoted as \( \{(c_i, s_i); i = 1, 2, \ldots \} \), where \( c_i \) represents the concentration vector of analytes in the \( i^{th} \) mixture sample, and \( s_i \) represents the spectrum of the \( i^{th} \) sample. Figure 1.1 summarized the proposed calibration procedure.

For efficient sampling, a sequential DOE is adopted involving iterations as shown in Figure 1.1. Since the information regarding the underlying relationship is not known when designing experiments, preliminary experiments are carried out first following the initial design determined by a "naive" once-and-for-all design, leading to the initial data set. The spectra data are modeled to obtain the calibration model, and statistical inference (the
model quality reflected by estimation uncertainty) derived. If desired model quality has been achieved, then stop. Otherwise, more sampling will be performed following the augmented design.

1.4 Contribution

The developed calibration procedure represents a new addition to the chemometric literature in the following aspects. It enables the quantification of multiple analytes in an unknown sample based on its Raman spectrum. It quantifies the uncertainty of target estimates (estimated analyte concentrations). It guides the calibration experiments to minimize the experimental effort that it takes to meet the estimation quality (uncertainty) requirements.

The resulting calibration model from the procedure will be used for analyte quantification as illustrated in Figure 1.2. The Raman spectrum of an unknown sample will be pre-processed, and then fed to the calibration model for concentration estimates of the underlying analytes.
Initial Design (Section 4.2): using classic once-and-for-all methods to determine the locations of initial design points, then carry out experiments following the initial design.

Calibration Modeling (Chapter 3): fit the calibration models to quantify the relationship between the concentration and Raman Spectra.

Quantify Uncertainty of Model Estimation (Section 3.4): estimate the variability of model estimation by resampling the experimental data.

Design Augmentation (Section 4.3): select a new design point adding to the current design and collect data at this new design point.

Design Stop? (Section 4.4): determine whether the calibration models meet the desired precision.

Final calibration model which meet the desired precision.

Figure 1.1: Overview of the calibration procedure.
A new spectrum of sample with unknown concentration

Data Pre-processing

Calibration Models

Concentration Estimation within Desired Precision

Figure 1.2: The objective of the calibration procedure
Chapter 2

Data Description

The data resulted from chemometric experiments are denoted as \( \{(c_i, s_i); i = 1, 2, \ldots\} \), where \( c_i \) represents the concentration vector of \( P \) analytes in the \( i^{th} \) mixture sample, with
\[
c_i = (c_{1i}, c_{2i}, \ldots, c_{Pi})
\]
and \( s_i \) represents the spectrum of the \( i^{th} \) sample. It is worth to note that the spectrum \( s_i \) is a two-dimensional curve (Figure 2.1), with the x-axis being Raman Shift denoted as \( x = \{x_j; j = 1, 2, \ldots, J\} \) and y-axis being intensity values denoted as \( y = \{y_j; j = 1, 2, \ldots, J\} \). Herein, a spectrum \( s \) can be represented as a set of points \( \{(x_j, y_j); j = 1, 2, \ldots, J\} \) and \( J \) represents the length of the spectrum.

The observed spectrum from experiments consists of desired signal as well as undesired elements. Basically, a spectrum can be decomposed into 3 parts as shown in Figure 2.2:

![Raman Spectrum](image)

Figure 2.1: An example of Raman spectrum
Background Raman signals are usually obscured by the background due to fluorescence from organic molecules and contamination [19]. The background may lead to serious problems if the data recorder reaches the detection limits during the practical operation (Blue curve in Figure 2.2).

Noise Noise is another significant source of variation, usually caused by the inherent characteristic of sensor or detector, varying from sample to sample [20]. In general, noise was observed in the form of high frequency part of a spectrum (Black curve in Figure 2.2).

Signal This is the valid part of the spectrum which is wanted by users. The useful information of analytes is included in the signal (Red curve in Figure 2.2).

Given the decomposition of spectrum, the intensity values at each Raman Shift $x_j$ also consist of three parts: the intensity of background denoted as $Y_j^{(b)}$, the intensity of signal denoted as $Y_j^{(s)}$ and noise denoted as $\varepsilon_j$. That is, $y_j = Y_j^{(b)} + Y_j^{(s)} + \varepsilon_j$. 

Figure 2.2: Spectrum decomposition
Chapter 3

Calibration Modeling

3.1 Background Removal

The background noise of a spectrum needs to be removed from the raw Raman spectrum. In this work, the background removal is performed by the Modified Polyfit method developed by Lieber et al [21]. The basic idea behind this method is to fit a least-squares polynomial function. However, the Raman bands should not be selected as the background data. Only the points lain on the bottom of the spectrum can be considered as the background. In order to remove the Raman bands from the fit, an automatically point reassignment method (Algorithm 1) is developed. First, all points on the spectrum are used to fit a polynomial function denoted as $F^{\text{ply}}(x_j)$ (Figure 3.1 A). Then, the points in the fitted curve that have intensity values higher than their respective values on spectrum are reassigned to the original value (Figure 3.1 B). Repeat this process until there is no point on the fitted curve needing reassignment. However, affected by noise factors and other inherent properties, it is inefficient, even impossible to obtain the optimal fitted curve. Thus, a termination criterion is defined to cease the process when there is convergence in the number of points reassigned by each iteration. The background curve denoted as $s^{(b)}$ is then subtracted from the raw spectrum.

3.2 Forward and Inverse Directions

In the chemometric analysis, there are two directions involved: forward and inverse directions:
### Algorithm 1: Background removal algorithm

**Input:** The Raman spectrum \( s = \{(x_j, y_j); j = 1, 2, ..., J\} \).

**Initialize** Initialize \( \text{Stop} = false, \text{Count} = 0. \)

**while** \( \text{Stop}=false \) do

- Fit a \( P \)-order polynomial function \( F^{ploy}(x_j) \) and calculate the \( \hat{y}_j = F^{ploy}(x_j) \).

  **for** \( j=1,2,...,J \) do
  
  | If, \( y_j \geq \hat{y}_j \), then \( \text{Count} = \text{Count} + 1 \), set \( y_j = \hat{y}_j \). |

end

- If, \( \text{Count} \leq 0.02J \), \( \text{Stop} = true \) and **EXIT LOOP**, ELSE CONTINUE.

end

- Exposing a sample with known analyte concentrations \( c \) to observe its Raman spectrum \( s \) is represented as \( c \rightarrow s \), which is considered as the forward direction. Typically, such samples are prepared under the laboratory environment and the concentrations are measured with extremely high accuracy, and thus the concentrations \( c \) can be treated as free of errors. The observed spectrum, on the other hand, is subject to random errors caused by the instrument noise, impurity, fluorescence, etc.

- Estimating the analyte concentrations of an unknown sample based on its observed spectrum, \( s \rightarrow c \), is considered as the inverse direction.

The ultimate objective of this work is to estimate the concentrations of analytes of an unknown sample given its Raman spectrum, which is described by the inverse direction. To obtain the concentration estimates with higher accuracy, DOE is necessary, which provides a set of design points \( \{c_1, c_2, ...\} \), at which the corresponding Raman spectra could be observed by the instrument. This process follows the forward direction.

### 3.3 Statistical Modeling

Since there are two directions involved, two corresponding calibration models need be developed in this work: forward and inverse calibration models.
3.3.1 Forward Modeling

To model the relationship in forward direction, the input factors of forward model are analyte concentrations and the Raman Shift values of the spectrum denoted as $\mathbf{w} = (x, c)$. The output response is the intensity values of spectrum after removing the background denoted as $y^{(bf)}$, which is calculated by Equation (3.1),

$$y_j^{(bf)} = y_j - Y_j^{(b)}, \quad (3.1)$$

where $y_j$ is the $jth$ intensity value of the observed spectrum at a setting $\mathbf{w}_i$ and $Y_j^{(b)}$ is obtained by the background removal method (Algorithm 1).
Recall \( J \) represents the length of the spectrum. Define \( I \) as the number of different concentration combinations. The data set to do the forward modeling is represented as:

\[
\{(w_i, y^{(bf)}_j); i = 1, 2, \ldots, I; j = 1, 2, \ldots, J\},
\]

(3.2)

where \( L \) denotes the number of distinct design points (i.e., factor setting at which experiments are performed) and \( L = I \times J \), \( w_i \) the \( i^{th} \) design point with \( w_i = (x_i, c_i) \), \( y^{(bf)}_j \) the intensity value after background removal (response) of the \( j^{th} \) replicated spectrum at a setting \( w_i \) (Equation (3.1)), and \( n \) the number of replications performed at each design point \( w_i \).

The forward model is written as

\[
y^{(bf)} = F(w; \Gamma) + \varepsilon(w),
\]

(3.3)

where \( w = (x, c) \), \( \Gamma \) is a vector of parameters, \( \varepsilon(w) \) is the random error. Since the data set to fit the forward model in this work contains non iid random errors which form a time series structure, stochastic kriging (SK)[22] is adapted to estimate the forward model considering the powerfulness and flexibility of Kriging models [23].

**Stochastic Kriging for Data with Time Series Errors**

In this part, the Stochastic Kriging (SK) modeling and inference method is adapted to model the data with special data structure containing time series errors.

The response at a setting \( w \) for the \( j^{th} \) replication is modeled by the adapted SK as

\[
y_j^{(bf)}(w) = Y(w) + \varepsilon_j(w) = f(w)^\top \beta + M(w) + \varepsilon_j(w).
\]

(3.4)

The expectation \( Y(w) \) consists of two parts: \( f(w)^\top \beta \) and \( M(w) \). \( f(w) \) is a vector of known functions of \( w \), and \( \beta \) a vector of unknown parameters of compatible dimension. In this work, we set \( f(w)^\top \beta = \beta_0 \), which has been widely accepted as sufficient for most
applications. The term $M(w)$ represents a realization of stationary Gaussian process of a mean $0$. We refer to the stochastic nature of $M(w)$ as extrinsic uncertainty. The intrinsic noise $\{\varepsilon_j(w); j = 1, 2, \ldots , n\}$ at a design point $w$ is naturally independent and identically distributed across replications. The error variance $\text{Var}[\varepsilon(w)]$ is allowed not to be constant, depending on $w$. For a given concentration combinations $c_i$, the intrinsic variability of responses across two Raman Shift settings $x_h$ and $x_k$ are not independent with the intrinsic dependence $\text{Corr}[\varepsilon(x_h, c_i), \varepsilon(x_k, c_i)] > 0(i = 1, 2, ..., L)$; $\varepsilon(x_i, c_h)$ and $\varepsilon(x_j, c_k)$ are independent for any two settings with different concentrations $h \neq k$.

The adapted SK requires the following assumption:

**Assumption 1** The random field $M$ is a stationary Gaussian random field; $\varepsilon_1(w), \varepsilon_2(w), \ldots , \varepsilon_n(w)$ are iid $N(0, \text{Var}[\varepsilon(w)])$; $\varepsilon_j(x_h, c_i)$ is not independent of $\varepsilon_j(x_k, c_i)$ for all $i = 1, 2, ..., L$ ($h \neq k$), but independent of $M$; $\varepsilon(x_i, c_h)$ and $\varepsilon(x_j, c_k)$ are independent for any $h \neq k$; $\varepsilon(x_1, c_i), \varepsilon(x_2, c_i), \ldots , \varepsilon(x_I, c_i)$, form a time series structure for all $i = 1, 2, ..., L$.

With the sample data (Equation (3.2)), let the sample average at $w_i$ across the $n$ replications be:

$$\bar{y}(w_i) = \frac{1}{n} \sum_{j=1}^{n} y_{ij}^{(bf)} = \beta_0 + M(w_i) + \frac{1}{n} \sum_{j=1}^{n} \varepsilon_j(w_i) = \beta_0 + M(w_i) + \bar{\varepsilon}(w_i), \quad (3.5)$$

where $\bar{\varepsilon}(w_i)$ denotes the sample average error at $w_i$ across the $n$ replications.

Then let define the column vector of sample averages for $L$ distinct design points as:

$$\mathbf{y} = (\bar{y}(w_1), \bar{y}(w_2), \ldots , \bar{y}(w_L))^\top. \quad (3.6)$$

Similarly, let define the column vector of sample average errors as:

$$\mathbf{e} = (\bar{\varepsilon}(w_1), \bar{\varepsilon}(w_2), \ldots , \bar{\varepsilon}(w_L))^\top. \quad (3.7)$$
Extrinsic Uncertainty The extrinsic uncertainty modeled in adapted SK is specified by the covariance of the stationary Gaussian process $M(\cdot)$:

$$\text{Cov}[M(w_h), M(w_k)] = \delta^2 \cdot \text{Corr}[M(w_h), M(w_k)] = \delta^2 \cdot r_{hk}, \quad (3.8)$$

where $\delta^2$ is the variance of the Gaussian process for all $w$. And $r_{hk}$ denotes the extrinsic correlation across the factor settings $w_h$ and $w_k$. Choices of the functional forms to calculate $r_{hk}$ were discussed in details in Wang et al. [23]. In this work, the exponential correlation function is adopted as follows:

$$r_{hk} = \exp \left\{ \sum_{d=1}^{D} -\theta_d |w_{hd} - w_{kd}|^p \right\}, \quad (3.9)$$

where $\theta = (\theta_1, \theta_2, \ldots, \theta_D)$ is a vector of unknown parameters with $\theta_d > 0 (d = 1, 2, \ldots, D)$ and $D$ being the dimensions of input factor $w$. The parameter $p \in (0, 2]$ also needs to be estimated unless the quadratic correlation function [24] is taken by pre-specifying $p$ as 2.

Given the data set (Data 3.2) collected at $L$ distinct design points, the $L \times L$ variance-covariance matrix $\Sigma_M$ is defined as

$$\Sigma_M = \delta^2 \mathbf{R}_M(\theta) = \delta^2 \begin{pmatrix} 1 & r_{12} & \ldots & r_{1L} \\ r_{21} & 1 & \ldots & r_{2L} \\ \vdots & \vdots & \ddots & \vdots \\ r_{L1} & r_{L2} & \ldots & 1 \end{pmatrix}, \quad (3.10)$$

where $\mathbf{R}_M(\theta)$ denotes the correlation matrix with each element representing a correlation as explained above and involves the unknown parameters $\theta$. For an arbitrary point $w_0$, the
\( L \times 1 \) vector \( \Sigma_M(w_0, \cdot) \) is defined as

\[
\Sigma_M(w_0, \cdot) = \delta^2 \mathbf{v}(w_0, \theta) = \delta^2 \begin{pmatrix}
    r_{01} \\
    r_{02} \\
    \vdots \\
    r_{0L}
\end{pmatrix},
\]

where \( \mathbf{v}(w_0, \theta) \) is a correlation vector involving \( w_0 \) and \( \theta \).

**Intrinsic Uncertainty**  
For the intrinsic uncertainty due to sampling at a design point \( w_i = (x_i, c_i); (i = 1, 2, \ldots, L) \), the \( L \times L \) variance-covariance matrix of vector \( \bar{\varepsilon} \) defined in (3.7) is denoted as \( \Sigma_{\bar{\varepsilon}} \):

\[
\Sigma_{\bar{\varepsilon}} = \begin{pmatrix}
    \text{Cov}[\bar{\varepsilon}(x_1, c_1), \bar{\varepsilon}(x_1, c_1)] & \text{Cov}[\bar{\varepsilon}(x_1, c_1), \bar{\varepsilon}(x_2, c_1)] & \ldots & \text{Cov}[\bar{\varepsilon}(x_1, c_1), \bar{\varepsilon}(x_J, c_I)] \\
    \text{Cov}[\bar{\varepsilon}(x_2, c_1), \bar{\varepsilon}(x_1, c_1)] & \text{Cov}[\bar{\varepsilon}(x_2, c_1), \bar{\varepsilon}(x_2, c_1)] & \ldots & \text{Cov}[\bar{\varepsilon}(x_2, c_1), \bar{\varepsilon}(x_J, c_I)] \\
    \vdots & \vdots & \ddots & \vdots \\
    \text{Cov}[\bar{\varepsilon}(x_J, c_I), \bar{\varepsilon}(x_1, c_1)] & \text{Cov}[\bar{\varepsilon}(x_J, c_I), \bar{\varepsilon}(x_2, c_1)] & \ldots & \text{Cov}[\bar{\varepsilon}(x_J, c_I), \bar{\varepsilon}(x_J, c_I)]
\end{pmatrix}.
\]

(3.12)

From Assumption 1, \( \varepsilon(x_1, c_i), \varepsilon(x_2, c_i), \ldots, \varepsilon(x_J, c_i) \) form a time series structure for all \( i = 1, 2, \ldots, L \). Based on the propriety of time series structure, \( \text{Cov}[\bar{\varepsilon}(x_i, c_k), \bar{\varepsilon}(x_j, c_k)] = \text{Cov}[\bar{\varepsilon}(x_{i+h}, c_k), \bar{\varepsilon}(x_{j+h}, c_k)] \). Define

\[
\rho_{i,j}^{(k)} = \text{Cov}[\bar{\varepsilon}(x_h, c_k), \bar{\varepsilon}(x_{h+i}, c_{k+j})].
\]

(3.13)
Since \( \varepsilon(x_i, c_h) \) and \( \varepsilon(x_j, c_k) \) are independent when \( h \neq k \), \( \rho_{i,j}^{(k)} = 0 \) for all \( j \neq 0 \). Therefore, the matrix \( \Sigma_\varepsilon \) could be rewritten as:

\[
\Sigma_\varepsilon = U(\sigma, a, b) = \begin{pmatrix}
K_1 & 0 & 0 & \ldots & 0 \\
0 & K_2 & 0 & \ldots & 0 \\
0 & 0 & K_3 & \ldots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & 0 & \ldots & K_I
\end{pmatrix},
\] (3.14)

where

\[
K_k = \begin{pmatrix}
\rho_{0,0}^{(k)} & \rho_{1,0}^{(k)} & \rho_{2,0}^{(k)} & \ldots & \rho_{J-1,0}^{(k)} \\
\rho_{1,0}^{(k)} & \rho_{0,0}^{(k)} & \rho_{1,0}^{(k)} & \ldots & \rho_{J-2,0}^{(k)} \\
\rho_{2,0}^{(k)} & \rho_{1,0}^{(k)} & \rho_{0,0}^{(k)} & \ldots & \rho_{J-3,0}^{(k)} \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
\rho_{J-1,0}^{(k)} & \rho_{J-2,0}^{(k)} & \rho_{J-3,0}^{(k)} & \ldots & \rho_{0,0}^{(k)}
\end{pmatrix},
\] (3.15)

with the property that \( \rho_{-i,j}^{(k)} = \rho_{i,j}^{(k)} \).

\( \rho_{i,0}^{(k)} \) is calculated by Equation (3.16),

\[
\rho_{i,0}^{(k)} = \sigma_k^2 \sum_{j=0}^{\infty} \psi_j^{(k)} \psi_{j+i}^{(k)}, \quad i \geq 0,
\] (3.16)

\[
\psi_j^{(k)} = \begin{cases} 
1 & j = 0 \\
b_j^{(k)} + \sum_{i=1}^{p^{(k)}} a_i^{(k)} \psi_{j-i} & j = 1, 2, \ldots
\end{cases}
\] (3.17)

with the assumption that \( b_j^{(k)} = 0 \) if \( j \geq q^{(k)} \) and \( \psi_j^{(k)} = 0 \) if \( j < 0 \).
Estimation and Inference  The log-likelihood function of $\bar{y}$ in terms of the unknown parameters ($\beta_0, \delta^2, \theta, \sigma, a, b$) can be written as

$$
\ln L(\beta_0, \delta^2, \theta, \sigma, a, b) = - \ln[(2\pi)^{L/2}] - \frac{1}{2} \ln[\delta^2 R(\theta) + U(\sigma, a, b)]
- \frac{1}{2} (\bar{y} - \beta_0 1_L)^\top [\delta^2 R(\theta) + U(\sigma, a, b)]^{-1} (\bar{y} - \beta_0 1_L),
$$

(3.18)

where $1_L$ is a $(L \times 1)$ vector of ones. To estimate the parameters, the procedure which maximizes the Equation (3.18) is developed as follows: (i) Given $\delta^2, \theta, \sigma, a,$ and $b$, the maximum likelihood estimate of $\beta_0$ is

$$
\hat{\beta}_0(\delta^2, \theta, \sigma, a, b) = (1_1^\top [\delta^2 R(\theta) + U(\sigma, a, b)]^{-1} 1_1)^{-1} 1_1^\top [\delta^2 R(\theta) + U(\sigma, a, b)]^{-1} \bar{y}.
$$

(3.19)

(ii) substituting $\beta_0(\delta^2, \theta, \sigma, a, b)$ into Equation (3.18), the problem reduces to maximizing

$$
\ln L(\delta^2, \theta, \sigma, a, b) = - \ln[(2\pi)^{L/2}] - \frac{1}{2} \ln[\delta^2 R(\theta) + U(\sigma, a, b)]
- \frac{1}{2} (\bar{y} - \hat{\beta}_0(\delta^2, \theta, \sigma, a, b) 1_L)^\top [\delta^2 R(\theta, \Phi) + U(\sigma, a, b)]^{-1} (\bar{y} - \hat{\beta}_0(\delta^2, \theta, \sigma, a, b) 1_L),
$$

(3.20)

w.r.t. $(\delta^2, \theta, \sigma, a, b)$. However, solving Equation (3.20) is challenging due to high dimension of the decision variables involved. Therefore, an iterative procedure is developed as follows:

**Step 0:** Fit the SK model and calculate the residual $\hat{\varepsilon} = y^{(bf)} - \hat{y}^{(bf)}$.

**Step 1:** Fit an ARMA model based on the residual $\hat{\varepsilon}$, estimate the ARMA parameters $(\hat{\sigma}, \hat{a}, \hat{b})$.

**Step 2:** Substitute $(\hat{\sigma}, \hat{a}, \hat{b})$ into Equation (3.18) and (3.20), and using nonlinear optimization algorithm such as the Matlab fmincon function to estimate the $(\hat{\delta}^2, \hat{\theta})$.

**Step 3:** Use the Equation (3.21) to estimate $\hat{y}^{(bf)}$ and update $\hat{\varepsilon}$. 
Step 4: Repeat Step 1 - Step 3 until one of the following stop criteria is achieved. The stop criteria are: (1) the desired precision is achieved; (2) the estimated \( \tilde{y}^{(bf)} \) is converged.

For an arbitrary setting \( w_0 \), estimate the expected response \( Y(w_0) \) by

\[
\hat{Y}(w_0) = \hat{\beta}_0 + v(w_0, \hat{\theta})^\top [\hat{\delta}^2 R(\hat{\theta}) + U(\hat{\sigma}, \hat{a}, \hat{b})]^{-1} (\bar{y} - \hat{\beta}_0 1_L),
\]

where \((\hat{\beta}_0, \hat{\delta}^2, \hat{\theta}, \hat{a}, \hat{b})\) are the maximum likelihood estimates obtained from the previous steps. The mean squared error (MSE) of \( \hat{Y}(w_0) \) is obtained as

\[
\hat{\text{MSE}}[\hat{Y}(w_0)] = \hat{\delta}^2 - \hat{\delta}^4 v(w_0, \hat{\theta})^\top [\hat{\delta}^2 R(\hat{\theta}) + U(\hat{\sigma}, \hat{a}, \hat{b})]^{-1} v(w_0, \hat{\theta}) + \eta^2 (1_L^\top [\hat{\delta}^2 R(\hat{\theta}) + U(\hat{\sigma}, \hat{a}, \hat{b})]^{-1} 1_L)^{-1},
\]

where \( \eta = 1 - 1_L^\top [\hat{\delta}^2 R(\hat{\theta}) + U(\hat{\sigma}, \hat{a}, \hat{b})]^{-1} v(x_0, \hat{\theta}) \hat{\delta}^2 \).

The two-sided 100(1-\( \alpha \))% confidence interval for \( Y(w_0) \) can be constructed as:

\[
\hat{Y}(w_0) \pm z_{1-\alpha/2} \sqrt{\hat{\text{MSE}}[\hat{Y}(w_0)]},
\]

where \( z_{1-\alpha/2} \) is the 100(1 - \( \alpha \)/2)\(^{th} \) percentile of standard normal distribution.

3.3.2 Inverse Modeling

The inverse model models the analyte concentration vector \( c \) as a function of the background-free spectrum \( s^{(bf)} = \{(x_j, y_j^{(bf)}); j = 1, 2, ..., J\} \). However, it is very difficult, even infeasible to directly consider the vector \( y^{(bf)} \) as input due to its extremely high dimensions and limited observations.

To solve this problem, dimension reduction technique is employed to reduce the dimensions of input \( y^{(bf)} \) firstly. Then the inverse model is fitted on the transferred data set denoted as \((\gamma, c)\), where \( \gamma \) is the transferred input w.r.t. \( y^{(bf)} \) after dimension reduction.
The inverse model is written as:

$$c = G(\gamma; \Theta), \quad (3.24)$$

where, $G = (G^{(1)}, G^{(2)}, \ldots, G^{(P)})$ is a vector of function of dimension $P$, and $\Theta$ represents the unknown parameters. Each element function $\{G^{(p)}; (p = 1, 2, \ldots, P)\}$ approximates the $p$th analyte $c_p$ as a function of the vector $\gamma$.

**Model Selection of Inverse Modeling** There are two major problems needed to be solved when doing the inverse modeling: the dimension of transferred input $\gamma$, and the modeling method used to estimate the inverse model. There are a wide range of dimension reduction methods. In the empirical study later in this paper (Chapter 5), PCA is adopted as an example to calculate the transferred input $\gamma$. The dimension of $\gamma$ is determined by the percentage of information carried out by $\gamma$. This information percentage typically takes values between the range $[0.9, 0.99]$. In addition, the dimension of $\gamma$ needs to be not less than the dimension of $c$ which is $P$. The modeling method is determined by the properties of data set $\{(\gamma_i, c_i); i = 1, 2, \ldots\}$. In the case of the data $(\gamma, c)$ with nonlinearity and high dimensions, the nonlinear modeling method, Deterministic Kriging (DK), is employed to estimate the inverse model.

**3.4 Quantifying Estimation Uncertainty**

The ultimate goal of the calibration procedure is estimating the concentration vector for a given spectrum. On a sample with true but unknown concentration $c^{(f)}$, a random spectrum $s^{(f)} = (x, y^{(f)})$ can be observed. After removing the background, the inverse model is able to provide a concentration estimate:

$$\hat{c}^{(f)} = G(\gamma^{(f)}; \hat{\Theta}). \quad (3.25)$$
Here, we are interested in the uncertainty quantification for the inverse model which involves two types:

- \( \text{Var}[\hat{\Theta}] \), the variance of the estimated model parameters \( \hat{\Theta} \) of the inverse model, which would affect the quality of the estimated analyte concentrations.

- \( \text{Var}[\hat{c}^{(f)}] \), the variance of the estimated analyte concentrations \( \hat{c}^{(f)} \) for a sample.

With the increasing of sample size, \( \text{Var}[\hat{\Theta}] \) is decreasing and is associated with D-optimum design criterion in the DOE literature [25]. In this paper, the \( \text{Var}[\hat{\Theta}] \)-based D-optimum criterion is applied to guide the experiments (Section 4.3). In such a way, \( \text{Var}[\hat{\Theta}] \) can be minimized within a limited size of experimental efforts.

However, in the practical use of chemometric analysis, \( \text{Var}[\hat{c}^{(f)}] \) is usually what directly concerns the end users. Thus, \( \text{Var}[\hat{c}^{(f)}] \) is utilized in this work to derive the stopping criterion for the calibration experiments (Section 4.4). As discussed in Equation (3.25), \( \hat{c}^{(f)} \) is a function of the fitted parameters \( \hat{\Theta} \) and the transferred responses \( \gamma^{(f)} \). Thus, two sources are considered as contributing to the variability of \( \hat{c}^{(f)} \):

- \( \text{Var}[\gamma^{(f)}] \), the variance of the transferred responses which are subject to random errors. \( \text{Var}[\gamma^{(f)}] \) represents the inherent variability caused by the instrument when observing the spectrum, and cannot be reduced through experiments-based modeling effort.

- \( \text{Var}[\hat{\Theta}] \) as explained above, follows the decreasing trend with the increasing of experimental effort and converges to zero if the sample size is large enough.

To provide valid variability estimates for the inverse model, bootstrap resampling method [26] is adapted to evaluate \( \text{Var}[\hat{c}^{(f)}] \) and \( \text{Var}[\hat{\Theta}] \). The bootstrap is a data-based method for statistical inference. By resampling, the variability of estimates of interest is quantified.

The resampling follows the forward direction (Section 3.2) as the real sampling does. Given the real data set \( \{(w_i, y_{ij}^{(bf)}); i = 1, 2, \ldots, L; j = 1, 2, \ldots, n\} \), the forward model is
fitted as $F(w; \hat{\Gamma})$, which is used to resample the spectra mimicking the real ones by step (1) of the Algorithm 2: At a certain design point $w_i$, $B$ spectra are regenerated by adding a random error derived from the time series generator (Section A.2.1) to the estimated response. Based on $B$ inverse models output from Algorithm 2, the uncertainty of the fitted model parameters $\text{Var}[\hat{\Theta}]$ can be evaluated based on the collection of $B$ model parameters $\{\hat{\Theta}_b^*; b = 1, 2, ..., B\}$ by Formula 3.26. Empirical studies suggest that $B = 1000$ usually suffice as a bootstrap sample size for the construction of confidence intervals (CIs) [27].

\[
\text{Var}[\hat{\Theta}] = B^{-1} \sum_{b=1}^{B} (\hat{\Theta}_b^* - \hat{\Theta})(\hat{\Theta}_b^* - \hat{\Theta})'.
\] (3.26)

**Algorithm 2: Bootstrap.1**

**Input:** (a) The Raman Shift $x$; (b) the design points $\{c_i; i = 1, 2, ..., I\}$, at which real data have been collected; (c) the forward model $F(w; \hat{\Gamma})$ and the inverse model $G(\gamma; \hat{\Theta})$, both estimated from the real data collected at the design points $\{c_i; i = 1, 2, ..., I\}$; (d) the resample size $B$.

for $b=1,2,...,B$ do

1. for $i=1,2,...,I$ do

   - Set $c_{i,b}^* = c_i$ and $w_{i,b}^* = (x, c_{i,b})$.
   - Randomly sample $e_{i,b}^*$ from a time series generator ARMA(1,1,$\sigma(c)$) (Section A.2.1)
   - Set $y_{i,b}^* = F(w_{i,b}^*; \hat{\Gamma}) + e_{i,b}^*$.

end

2. Perform the dimension reduction procedure to get $\gamma_{i,b}^*$.

3. Based on the bootstrap data $\{(c_{i,b}^*, \gamma_{i,b}^*); i = 1, 2, ..., I\}$, fit the inverse model $G(\gamma; \hat{\Theta}_b^*)$.

end

**Output:** $B$ fitted model $\{G(\gamma; \hat{\Theta}_b^*); b = 1, 2, ...B\}$

Then how to estimate $\text{Var}[\hat{c}]$? As shown before, the inverse model $G(\gamma; \hat{\Theta})$, which is fitted from real data $\{(c_i, \gamma_i); i = 1, 2, ..., I\}$, is able to provide the point estimate of concentration $\hat{c}(f)$ for a given spectrum $s(f)$, where the true concentration vector of the
sample is denoted as $c^{(f)}$. Algorithm 3 is developed to quantify the uncertainty of the estimated concentrations $\text{Var}[\hat{c}^{(f)}]$.

**Algorithm 3: Bootstrap.2**

**Input:** (a) The target concentration vector $c^{(f)}$; (b) the Raman shift $x$; (c) the design points $\{c_i; i = 1,2,...,I\}$, at which the real data have been collected; (d) the forward model $F(w; \hat{\Gamma})$ and inverse model $G(\gamma; \hat{\Theta})$, both estimated from the real data collected at the design points $\{c_i; i = 1,2,...,I\}$; (e) the sample size $B$

**Call** Algorithm Bootstrap.1 with the inputs given above: (b), (c), (d) and (e).

**Return** $\{G(\gamma; \hat{\Theta}^*_b); b = 1,2,...,B\}$

**for** $b=1,2,...,B$ **do**

1. Generate a response vector $\gamma^*_b(f)$ at a setting $c^{(f)}$:
   - Randomly sample $e^*_b$ from a time series generator ARMA($1,1,\sigma(c)$)
   - Set $y^*_b(f) = F(w^{(f)}; \hat{\Gamma}) + e^*_b$ where $w^{(f)} = (x, c^{(f)})$
   - Reduce the dimension of $y^*_b(f)$ to get $\gamma^*_b(f)$.

2. Estimate the analyte concentrations based on the resampled responses $\gamma^*_b(f)$:
   $\hat{c}^*_b(f) = G(\gamma^*_b(f); \hat{\Theta}^*_b)$.

**end**

**Output:** $\{\hat{c}^*_b(f); b = 1,2,...,B\}$, the $B$ bootstrap estimates of $c^{(f)}$.

As discussed earlier, there are two sources contributing to $\text{Var}[\hat{c}^{(f)}]$: $\text{Var}[\gamma^{(f)}]$ and $\text{Var}[\hat{\Theta}]$. The resampling step (1) in Algorithm 3 accounts for one source $\text{Var}[\gamma^{(f)}]$ by resampling the spectra mimicking real ones subject to non iid random errors. And Algorithm 2 is called in Algorithm 3 to account for the variability of $\hat{\Theta}$.

The variance-covariance matrix $\text{Var}[\hat{c}^{(f)}]$ can then be estimated as follows:

$$\text{Var}[\hat{c}^{(f)}] = B^{-1} \sum_{b=1}^{B} (\hat{c}_b^{*}(f) - \hat{\bar{c}}^{*}(f))(\hat{c}_b^{*}(f) - \hat{\bar{c}}^{*}(f))', \tag{3.27}$$

where $\{\hat{c}_b^{*}(f); b = 1,2,...,B\}$ are outputs of Algorithm 3 and $\hat{\bar{c}}^{*}(f)$ is the average of $\{\hat{c}_b^{*}(f); b = 1,2,...,B\}$.
$\text{Var}[\hat{c}(f)]$ is a $P \times P$ matrix and the diagonal element of $\text{Var}[\hat{c}(f)]$ is represented as $(\text{Var}[\hat{c}(f)])_{p,p}(p = 1, 2, ..., P)$.

The standard error (SE) (Equation (3.28)) measures the uncertainty of the estimated concentration of $p$th component in the mixture $\hat{c}_p(f)$.

$$\text{SE}[\hat{c}_p(f)] = \sqrt{(\text{Var}[\hat{c}(f)])_{p,p}}.$$ (3.28)

Then, the approximate uncertainty of the concentration estimate $\hat{c}_p(f)$ could be estimated in the form of $\text{SE}[\hat{c}_p(f)]$:

$$[\hat{c}(f) - z_{1-\alpha/2} \cdot \text{SE}[\hat{c}(f)], \hat{c}(f) + z_{1-\alpha/2} \cdot \text{SE}[\hat{c}(f)]],$$  (3.29)

where $\alpha$ is a small percentage which typically takes values between the range $[0.05, 0.1]$, and $z_{1-\alpha/2}$ is the $100(1 - \alpha/2)$th percentile of standard normal distribution. This interval is interpreted as the unknown true concentration of analytes $c_p(f)$ would falls within this interval with a chance of $1 - \alpha$. 

22
Chapter 4

Design of Experiments

To achieve high-quality calibration models with the least experimental effort, DOE needs to be performed. Compared to standard once-and-for-all experiments, sequential DOE, adopted in this paper, belongs to model-based DOE, which aims at optimizing the quality of the resulting estimated models with respect to the design points. It typically leads to savings in sampling size, time, and cost [28]. Herein, the sequential design procedure is adapted to efficiently collect the experimental data for the calibration modeling with the prespecified inputs given as follows:

- $\mathcal{C}$: the feasible region of the concentration vector, as defined the constraints: $c_{L,p} \leq c_p \leq c_{U,p}$; $p = 1, 2, ..., P$
- $\{\delta_p; p = 1, 2, ..., P\}$: the desired SE bound on the estimated concentration of $p$th component where $\text{SE}[\hat{c}_p(f)] \leq \delta_p$.
- $N_{\text{max}}$: the total number of samples allowed, which is depended on the experimental budget available.
- $N_0$: the number of samples assigned to the initial stage.
- $n$: the number of replications at each design point. It is recommended to set $n \geq 3$, which is the common practice with chemometric calibration.

An overview of the sequential design procedure (Algorithm 4) is briefed as follows: First, $N_0$ preliminary experiments are performed following the initial design $\mathcal{D}_0$, which is discussed in Section 4.2. Based on the preliminary data collected following the initial design,
the forward model $F(c; \hat{\Gamma})$ and inverse model $G(\gamma; \hat{\Theta})$ are fitted by applying the modeling procedure in Chapter 3. Second, the information contained in the two fitted models is utilized to guide how to select the remaining design points one by one: At a certain stage $s$, perform Algorithm 5 to determine the location of the next design point, leading to the updated design $D_s$. Then the forward and inverse models are also updated based on the data collected from the design in the current stage. Last, evaluate the quality of the inverse model by calling Algorithm 6. If the stop criteria is achieved, then stop. Otherwise, repeat the whole process to add more design points until it stops.

**Algorithm 4: Sequential design procedure**

1. **Step (1):** Set the stage index $s = 0$, perform initial design in Section 4.2.
2. **Step (2):** Carry out experiments following current design $D_s$ and based on the data collected, perform the calibration modeling described in Chapter 3 to fit $F(c; \hat{\Gamma})$ and $G(\gamma; \hat{\Theta})$.
3. **Step (3):** Call Algorithm 6: Design.Stop to evaluate the quality of inverse model and return the Boolean variable STOP.
   - **If** STOP = false, set $s = s + 1$ and perform Algorithm 5: Design.Augment to add another design point and go to Step (2).
   - Otherwise, quit the sequential design procedure.

**Output:** The design $D_s$

## 4.1 Design Space

The design space is a $P$-dimensional hypercube $C$, and the design point is a concentration vector inside of the hypercube. In this work, the hypercube $C$ is divided into a fine grid of equally-sized hyper-rectangles, and the grid points are considered as the candidate design points denoted as $\{c_g^{(\text{Grid})}; g = 1, 2, ..., G\}$, where $G$ is total number of grid points.

## 4.2 Initial Design

At the initial stage $s = 0$, $N_0$ samples are to be assigned. The number of $N_0$ is set as $1/4$ to $1/2$ of $N_{max}$ based on the general guideline found in [29]. Given the replications of each
design \( n \), the number of design points \( I_0 \) is then calculated as \( I_0 = N_0/n \). The purpose of the initial design is to provide a fair coverage of the design space to gain some information on the target relationship. All the vertex points of the hypercube have to be included in \( I_0 \) in order to avoid the extrapolation problem. The remaining non-vertex design points are evenly dispersed within the hypercube. There are a range of designs can be adopted as initial design. In this work, the Latin Hypercube Design (LHD) [30] was adopted as an example to determine the locations of \( I_0 \) design points. Denote the initial design as

\[
D_0 = \begin{pmatrix} c_1 & c_2 & \cdots & c_{I_0} \\ n & n & \cdots & n \end{pmatrix}, \tag{4.1}
\]

Based on the initial design \( D_0 \), \( N_0 \) pilot experiments are carried out resulting in the initial data denoted as \{\((c_i, s_{ij}); i = 1, 2, ..., I_0; j = 1, 2, ..., n\}\}. From the initial data, two preliminary models are fitted: forward model \( F(w; \hat{\Gamma}) \) and inverse models \( G(\gamma; \hat{\Theta}) \) (Section 3.3). These two fitted models which contain the information of the target relationship will be utilized to guide the remaining designs.

### 4.3 Design Augmentation

After the initial design, update \( s \) by \( s = s+1 \) whenever the design augmentation is performed.

At the beginning of Stage \( s \), a total of \( I_{s-1} = I_0 + s - 1 \) design points have been determined by the design \( D_{s-1} \) of the last stage \( s-1 \). Based on the data collected from the last Stage \( s-1 \), the forward and inverse models have been fitted.

Then the task of this Stage \( s \) is to select the next design point among the rest of the candidate design points. As discussed in Section 3.4, \( \text{Var}[\hat{\Theta}] \)-based D-optimum design criterion is utilized here to select this design point by minimizing the uncertainty of the estimated parameters, denoted as \( \text{Var}[\hat{\Theta}] \), of the resulting inverse model \( G(\gamma; \hat{\Theta}) \). The data used to estimate \( G(\gamma; \hat{\Theta}) \) are obtained on the basis of the updated design \( D_s \), which is
a combination of design in the last stage $\mathcal{D}_{s-1}$ and this new design point. This metric is widely used to evaluate the quality of the fitted inverse model $G(\gamma; \hat{\Theta})$. Recall that the bootstrapping-based method (Algorithm 2) is applied to numerically estimate $\text{Var}[\hat{\Theta}]$, and embedded in the design augmentation algorithm (Algorithm 5). In this algorithm, the remaining candidate design points are all tested by calculating the $\text{Var}[\hat{\Theta}]$. The one the minimum value of $\text{Var}[\hat{\Theta}]$ is selected to be added at this stage. Denote the design at the end of stage $s$ as:

$$
\mathcal{D}_s = \begin{pmatrix}
c_1 & c_2 & \cdots & c_{I_0+s-1} \\
n & n & \cdots & n
\end{pmatrix},
$$

(4.2)

4.4 Stopping Criterion

At the end of Stage $s$, the design is updated to be $\mathcal{D}_s$ which is the union of the design of last stage $\mathcal{D}_{s-1}$ and the new design point selected at this stage. The total number of samples used so far is $N_s = N_0 + s \times n$. Design the $N_s$ experiments following $\mathcal{D}_s$ to collect data and update the forward and inverse models based on the newest data set.

Then the inverse model is evaluated to see if the desired quality of model is achieved by Equation (4.4). If the standard error of the estimated concentration of each analyte $\text{SE}[\hat{c}_p]$ for candidate design points are all not greater than a threshold $\delta_p$, the goodness of the resulting inverse model is considered as qualified.

$$
\text{SE}[\hat{c}_p^{(f)}] \leq \delta_p, \forall p = 1, 2, \ldots, P.
$$

(4.4)

The sequential design procedure stops by meeting either of two stopping criteria (Algorithm 6): (1) the quality of the inverse model is eligible; (2) the number of samples taken has already reached the maximum $N_{max}$.
Algorithm 5: Design.Augmentation

Input: (a) the design point \( \{c_i; i = 1, 2, ..., I_0 + s - 1\} \) in the design \( \mathcal{D}_{s-1} \); (b) the forward model \( F(\mathbf{w}; \hat{\Gamma}) \) and inverse model \( G(\gamma; \hat{\Theta}) \), both estimated from the data obtained by performing the design \( \mathcal{D}_{s-1} \); (c) the resampling size \( B \); (d) the remaining candidate design points at stage \( s \) (the grid points) \( \{c^{(\text{Grid})}_g; g = 1, 2, ..., G(s)\} \);

1. for \( g = 1, 2, ..., G(s) \) do
   
   if \( c^{(\text{Grid})}_g \cap \mathcal{D}_{s-1} = \emptyset \) then
     
     • Call Algorithm 2 with the inputs: the design points \( c^{(\text{Grid})}_g \cup \mathcal{D}_{s-1} \), input (b), (c) above.
     
     Return fitted inverse model \( \{G(\gamma; \hat{\Theta}^*_b(g)); b = 1, 2, ..., B\} \).
     
     • Calculate \( \hat{\text{Var}}[\hat{\Theta}^{(g)}] \) based on \( \{\hat{\Theta}^*_b(g); b = 1, 2, ..., B\} \) by using Formula (3.26).
   
end

2. The design point which gives the minimum value of \( \{\hat{\text{Var}}[\hat{\Theta}^{(g)}]; g = 1, 2, ..., G(s)\} \)
   
   is selected by Equation (4.3).

   \[
   c^{(\text{opt})} = \min_{c^{(\text{Grid})}_g} \hat{\text{Var}}[\hat{\Theta}^{(g)}]; \forall g = 1, 2, ..., G(s). \tag{4.3}
   \]

3. Update \( \mathcal{D}_s = \mathcal{D}_{s-1} \cup c^{(\text{opt})} \)

Output: The updated design \( \mathcal{D}_s \).
Algorithm 6: Design.STOP

Input: (a) the design point \( \{c_i; i = 1, 2, ..., I_0 + s\} \) in current design \( \mathcal{D}_s \), at which the real data will be collected; (b) the forward model \( F(w; \hat{\Gamma}) \) and inverse model \( G(\gamma; \hat{\Theta}) \), both estimated from the data collected based on the design \( \mathcal{D}_s \); (c) the resampling size \( B \); (d) the candidate design points \( \{c_{g}^{(\text{Grid})}; g = 1, 2, ..., G\} \); (e) the desired precision \( \{\delta_p; p = 1, 2, ..., P\} \); (f) the maximum number of samples allowed \( N_{\text{max}} \).

Initialize \( \text{STOP} = \text{false} \)

if \( N \geq N_{\text{max}} \) then
  \( \text{STOP} = \text{true} \)
else
  for \( g = 1, 2, ..., G \) do
    1. Set \( c^{(f)} = c_{g}^{(\text{Grid})} \)
    2. Call Algorithm 3: Bootstrap 2, with the following inputs: \( c^{(f)} \), input (a), (b), and (c) given above.
       Return \( \{\hat{c}_b^{*}(f); b = 1, 2, ..., B\} \), the \( B \) bootstrap estimates of \( c^{(f)} \)
    3. Calculate \( \{\text{SE}[\hat{c}_p^{(f)}]; p = 1, 2, ..., P\} \) based on \( \{\hat{c}_b^{*}(f); b = 1, 2, ..., B\} \).
    4. if \( \text{SE}[\hat{c}_p^{(f)}] > \delta_p \) for any \( p \), then set \( \text{STOP} = \text{false} \) and \( \text{EXIT LOOP} \).
       ELSE set \( \text{STOP} = \text{true} \)
  end
end

Output: The boolean variable \( \text{STOP} \), indicating the termination of the sequential design procedure if true.
Chapter 5

Case Study

Two empirical cases were designed and performed to demonstrate the advantages of the proposed calibration procedure. In those cases, Raman spectrum is used to measure the concentrations of Dithiocarbamate (DTC) pesticides, which are a threat to human health. Both cases are derived from the spectrum-concentration study of DTC pesticides performed by Zhu et al [10].

**Case 1:** A single-material case is developed to show the efficiency of the proposed calibration procedure. The estimated accuracy of the resulting model from the calibration procedure is compared to those provided by the peak-based method which is utilized in current chemometric study. There is only one material in case 1 due to the restriction of the peak-based method.

**Case 2:** A two-material case is developed to demonstrate how the proposed calibration procedure is applied to the mixture to generate the calibration model of desired precision.

These two empirical studies are based on the simulation model developed in Chapter A. Only in simulation-based experiments, it allows for performing the efficient DOE which is critical to achieve a high-quality calibration models with the least experimental effort and to evaluate the accuracy of the resulting model by providing the true benchmark.

5.1 Case 1: Single Material Case

The concentration of ferbam, a DTC pesticide, would be quantified by both the benchmark method and the calibration procedure. The simulation model in Appendix A is used to
generate the simulated spectra that mimic real experimental outcomes w.r.t the samples of ferbam.

- Simulate Background: the intensity of background curve $s^{(b)}$ is simulated by Formula (A.2) with the parameters given in Table A.2.

- Simulate Noise: the intensity of noise curve $s^{(e)}$ is simulated by Formula A.4 and $\sigma(c)$ is calculated by the Formula (A.5).

- Simulate Signal: the intensity of signal curve $s^{(s)}$ is simulated by Formula A.6 with the value of parameter $u, \omega$ given in Table A.3. The value of $A$ is calculated by the Formula (A.6).

Then the simulation model in case 1 to generate a spectrum mimicking the real one is derived by adding the three portions above up.

5.1.1 Applying the Calibration Procedure

The sequential procedure in Section 4 is applied to design the experiments. The inputs of DOE are given as follows for this case: $N_{\text{max}} = 18; N_0 = 9; n = 3; \mathbf{C} = [0, 2000]; \delta_1 = 100$. The candidate design points $\{c^{(\text{Grid})}_g; g = 1, 2, ..., G\}$ in $\mathbf{C}$ are 100 evenly-spaced points with a step size of 20. The application of the calibration procedure is as follows.

**Initial Design:** A total of $N_0 = 9$ samples are assigned to the initial stage. The number of replications at each design point is given as $n = 3$, and thus, the number of design points is $N_0 \div n = 3$. Since the design space of this case has only one dimension, the locations of these 3 design points are evenly-spaced on the concentration range $[0, 2000]$ by LHD. 3 experiments are carried out based on the simulation model in Chapter A at each design point, leading to the 9 initial data denoted as $\{(c_i, s_{ij}); i = 1, 2, 3; j = 1, 2, 3\}$. The initial
design of case 1 is as follows:

\[ \mathcal{D}_0 = \begin{pmatrix} 0 & 1000 & 2000 \\ 3 & 3 & 3 \end{pmatrix}. \] (5.1)

Based on this initial data, two models are fitted: the forward model \( F(w; \hat{\Gamma}) \) and inverse model \( G(\gamma; \hat{\Theta}) \). The estimated parameters in forward and inverse models are displayed in Table 5.1 and 5.2.

**Table 5.1: The adapted SK parameters of the forward model based on initial design**

<table>
<thead>
<tr>
<th>( \hat{\beta}_0 )</th>
<th>( \hat{\delta}^2 )</th>
<th>( \hat{\theta}_1 )</th>
<th>( \hat{\theta}_2 )</th>
<th>( \hat{\theta}_3 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0408</td>
<td>0.4112</td>
<td>0.2912</td>
<td>0.3730</td>
<td>0.3885</td>
</tr>
<tr>
<td>( \hat{\sigma}_1 )</td>
<td>( \hat{\sigma}_2 )</td>
<td>( \hat{\sigma}_3 )</td>
<td>( \hat{\alpha}_1 )</td>
<td>( \hat{\alpha}_2 )</td>
</tr>
<tr>
<td>31.17</td>
<td>50.71</td>
<td>64.45</td>
<td>0.59</td>
<td>0.63</td>
</tr>
<tr>
<td>( \hat{\alpha}_3 )</td>
<td>( \hat{b}_1 )</td>
<td>( \hat{b}_2 )</td>
<td>( \hat{b}_3 )</td>
<td></td>
</tr>
<tr>
<td>0.84</td>
<td>-0.33</td>
<td>-0.27</td>
<td>-0.38</td>
<td></td>
</tr>
</tbody>
</table>

**Table 5.2: The DK parameters of the inverse model based on initial design**

<table>
<thead>
<tr>
<th>( \hat{\beta}_0 )</th>
<th>( \hat{\theta}_1 )</th>
<th>( \hat{\theta}_2 )</th>
<th>( \hat{\theta}_3 )</th>
<th>( \hat{\theta}_4 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0820</td>
<td>0.8721</td>
<td>0.5888</td>
<td>0.5005</td>
<td>0.4254</td>
</tr>
<tr>
<td>( \hat{\theta}_5 )</td>
<td>( \hat{\theta}_6 )</td>
<td>( \hat{\theta}_7 )</td>
<td>( \hat{\theta}_8 )</td>
<td></td>
</tr>
<tr>
<td>0.3617</td>
<td>0.3102</td>
<td>0.3656</td>
<td>0.3998</td>
<td></td>
</tr>
</tbody>
</table>

**Design Augmentation:** Here, the total number of samples allowed to be assigned is \( N_{max} - N_0 = 9 \). With the replication \( n = 3 \), there are additional 3 design points available for design augmentation, which means the maximum number of stage being \( s = 3 \). To determine the locations of added design points, the iterative procedure in Section 4.3 is utilized. The inputs needed for this iterative procedure have been obtained from the initial
Stage: initial design $\mathcal{D}_0$, the fitted forward model $F(w; \hat{\Gamma})$ and fitted inverse model $G(\gamma; \hat{\Theta})$. The locations of the resulting design points added from the design augmentation step are [205, 550, 685].

The whole design after applying the sequential DOE is as follows:

$$\mathcal{D}_3 = \begin{pmatrix} 0 & 205 & 550 & 685 & 1000 & 2000 \\ 3 & 3 & 3 & 3 & 3 & 3 \end{pmatrix}.$$ 

(5.2)

Calibration Modeling: Following the whole design $\mathcal{D}_3$ derived above, a total of $N = 18$ spectra are generated by simulation experiments in the same manner as the initial stage, which gives birth to 18 experimental data denoted as $\{(c_i, s_{ij}), i = 1, 2, \ldots, 6; j = 1, 2, 3\}$. Perform Algorithm 1 to remove the background from spectra, then PCA is employed to reduce the dimensions of spectra to ten, represented by $(\gamma_p; p = 1, 2, \ldots, 8)$. The final inverse model is fitted based on the estimation data $\{(\gamma_i, c_i); i = 1, 2, \ldots, 18\}$ by DK (Section 3.3.2). Table 5.3 shows the estimated parameters of the inverse calibration model.

It is worth noting that the outcomes (resulting design, sample data and fitted models) of performing the calibration procedure each time are random: $\mathcal{D}_3$ represents one possible realization of the sequential DOE and the models are estimated from the randomly sampled data following the sequential design.

Table 5.3: The DK parameters of the inverse model based on final design

<table>
<thead>
<tr>
<th>$\hat{\beta}_0$</th>
<th>$\hat{\theta}_1$</th>
<th>$\hat{\theta}_2$</th>
<th>$\hat{\theta}_3$</th>
<th>$\hat{\theta}_4$</th>
<th>$\hat{\theta}_5$</th>
<th>$\hat{\theta}_6$</th>
<th>$\hat{\theta}_7$</th>
<th>$\hat{\theta}_8$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.2779</td>
<td>0.8903</td>
<td>0.5739</td>
<td>0.4854</td>
<td>0.4106</td>
<td>0.3473</td>
<td>0.3473</td>
<td>0.5728</td>
<td>0.2716</td>
</tr>
</tbody>
</table>
5.1.2 Applying the Peak-Based Method

As described by Zhu et al [10], the peak at Raman Shift 1382 cm$^{-1}$, is selected as the characteristic peak to perform the peak-based method. That means the intensity values at $x = 1382$ on spectra are used to fit the prediction model.

The total number of samples here is the same as the sample size taken in Section 5.1.1, which is $N_{max} = 18$. The number of replication is also set to be $n = 3$. Naive design which is used in current chemometric study is performed: 3 replications are carried out at 6 evenly-spaced concentration levels $[0,400,800,1200,1600,2000]$, leading to 18 spectra. Background remove procedure (Algorithm 1) is performed, which results in the background-free intensity values $y^{(p)}$ of the characteristics peaks of 18 spectra. Since the modeling data, denoted as $\{(c_i, y_{ij}^{(p)}); i = 1,2,\ldots,6; j = 1,2,3\}$, is very simple, Simple Linear Regression (SLR) is employed to model the relationship in forward direction $c \rightarrow y^{(p)}$, represented as $y^{(p)} = \text{SLR}(c)$. Figure 5.1 shows an example of the sampling data depicted as blue stars and the fitted SLR model depicted as the red line.

To estimate the concentration of a sample from its $y^{(p)}$, the inverse calculation given in (5.3) is performed by numerical interpolation based on the fitted SLR model.

$$c = \text{SLR}^{-1}(y^{(p)}).$$ (5.3)

Here, $\text{SLR}^{-1}$ represents the functional dependence of $c$ upon $y^{(p)}$, assuming that the inverse mapping exists. In this work, the cubic spline interpolation recommended by [31] is used to perform the inverse computation for concentration estimation. The variability of concentration estimation is estimated by applying the Algorithm 2 and Algorithm 3 with the inputs of forward model replaced by $\text{SLR}(c)$ and inverse model replaced by (5.3).

5.1.3 Comparison of Prediction Results
Figure 5.1: Prediction model of peak-based method

**Compare the Accuracy of Point Estimate** To evaluate the prediction accuracy of the resulting calibration model, the estimated root mean squared error (ERMSE) is defined as follows:

$$\text{ERMSE}(\mathcal{C}(v)) = \sqrt{\frac{1}{\#[\mathcal{C}(v)]} \sum_{c \in \mathcal{C}(v)} (\hat{c} - c)^2}. \quad (5.4)$$

In Equation (5.4), $\mathcal{C}(v)$ represents the collection of the check points, which are evenly spaced in the design space; $\#[\mathcal{C}(v)]$ represents the total number of check points in the set $\mathcal{C}(v)$ and is equal to 100.

In light of the stochastic nature of the outcome of experiments, the quality of the proposed calibration procedure is statistically evaluated and compared to the peak-based method on a large of macro-replications (that is, independent applications of a certain method). Here, set the number of macro-replications as $M = 100$. For each method, apply two procedures independently 100 times, leading to 100 models and 100 estimated root mean squared errors denoted as $\{\text{ERMSE}^m; m = 1, 2, ..., 100\}$. Figure 5.2 shows box plots of 100 ERMSEs of both procedures respectively.
From Figure 5.2, the box of the proposed calibration procedure is lower, illustrating its better performance in terms of ERMSE compared to the peak-based method. And the box is relatively narrow, showing that the performance of the proposed method is stable. Except for the more accurate point estimate, the calibration procedure is able to provide the valid inferences (that is, quantify the uncertainty of the estimation) which is very important in practice.

![Figure 5.2: ERMSE box plots in case 1](image)

**Compare the Prediction Variability** To illustrate the comparison of estimated variability of concentration prediction, five points (Table 5.4) are picked to show their point estimates and prediction intervals (PI) (Section 3.4). As proven above, the point estimator given by the calibration procedure is more closer to the true value and PI is generally narrower than those given by peak-based method. This pattern of the result is generated by one macro-replication, holds consistently for all of the $M=100$ macro-replications carried out in this study.

One macro-replication refers to the process of applying each of the methods one time. To evaluate the performance of two procedures regarding the quality of PIs, $M = 100$ macro-replications are run leading to 100 PIs of the target concentrations $c^{(f)}$. Therefore,
Table 5.4: Comparison of point estimator and uncertainty

<table>
<thead>
<tr>
<th>$c(f)$</th>
<th>$\hat{c}(f)$</th>
<th>LB of PI</th>
<th>UB of PI</th>
<th>$\hat{c}(f)$</th>
<th>LB of PI</th>
<th>UB of PI</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>-107.57</td>
<td>187.11</td>
<td>63.89</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>200</td>
<td>433.48</td>
<td>728.06</td>
<td>202.51</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>600</td>
<td>662.03</td>
<td>949.73</td>
<td>603.94</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>900</td>
<td>922.94</td>
<td>1255.21</td>
<td>905.57</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1000</td>
<td>944.91</td>
<td>1202.50</td>
<td>1001.14</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1400</td>
<td>1306.22</td>
<td>1580.63</td>
<td>1405.74</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1800</td>
<td>1645.56</td>
<td>1970.32</td>
<td>1805.33</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

the coverage probability of the PIs can be estimated as the percentage of 100 PIs that include the true target concentrations $c(f)$. In our case, $c(f)$ is available from the simulation model on the purpose of evaluating the PIs. Ideally, the percentage of the PIs that actually contain $c(f)$ should be close to 95%, the nominal coverage level. Table 5.5 displays the coverage probabilities of the PIs given by peak-based method and our calibration procedure respectively, based on $M = 100$ macro-replications at 5 check points as an illustration. It shows that the PI provided by calibration procedure is more reliable than it provided by peak-based method.

Table 5.5: PI coverage probabilities of two procedures

<table>
<thead>
<tr>
<th>$c(f)$</th>
<th>50</th>
<th>200</th>
<th>600</th>
<th>900</th>
<th>1000</th>
<th>1400</th>
<th>1800</th>
</tr>
</thead>
<tbody>
<tr>
<td>Peak-based Method</td>
<td>1.00</td>
<td>0.99</td>
<td>0.90</td>
<td>0.93</td>
<td>0.89</td>
<td>0.94</td>
<td>0.85</td>
</tr>
<tr>
<td>Calibration Procedure</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
</tr>
</tbody>
</table>

5.2 Case 2: Two Material Case

In this case, the Raman spectrum is obtained from the sample containing both ferbam and thiram. The concentrations of those two analytes would be measured based on one Raman spectrum. In the same manner as case 1 (Section 5.1), simulation experiment based on the
model in Chapter A is employed to generate spectra under a concentration combination setting $c = (c_1, c_2)$.

### 5.2.1 Applying the Sequential DOE

The inputs of DOE are given as follows for this case: $N_{max} = 48$; $N_0 = 24$; $n = 3$; $\mathcal{C} = [0, 2000] \times [0, 2000]$; $\delta = [100, 100]$. The candidate design points $\{c^{(\text{Grid})}_g, g = 1, 2, ..., G\}$ in $\mathcal{C}$ (depicted as dots in Figure 5.3) are $20 \times 20$ evenly-spaced points with a step size of 100 at each dimension.

**Initial Design:** To assign $N_0 = 24$ samples in the initial stage $s = 0$ with the number of replications at each design point being $n = 3$, LHD (Section 4.2) is performed to determine the locations of the $I_0 = N_0 \div n = 8$ distinct design points. A realization of LHD is plotted as circles in Figure 5.3. Thus the initial design $\mathcal{D}_0$ consists of 8 distinct design points with 3 replications at each point. Following $\mathcal{D}_0$, a batch of experiments are implemented to generate 3 random spectra at each design point. On the basis of the initial data, denoted as $\{(c_i, s_{ij}), i = 1, 2, ..., 8; j = 1, 2, 3\}$, forward model $F(w; \hat{\Gamma})$ and inverse model $G(\gamma; \hat{\Theta})$ are fitted.

**Design Augmentation:** The total samples available to be assigned in this step is $N_{max} - N_0 = 24$. Given $n = 3$, the maximum number of distinct design point available is 8. The locations of additional points could be determined by applying Algorithm 5 with the input obtained from the initial stage: $\mathcal{D}_0$, $F(w; \hat{\Gamma})$ and $G(\gamma; \hat{\Theta})$. The diamonds in Figure 5.3 are the 7 design points added to the initial design at the end of design augmentation step. As is mentioned before, the design results in Figure 5.3 is a random outcome.

### 5.2.2 Calibration Modeling

At each of the 15 design points obtained above (Figure 5.3), 3 replicated spectra are produced. The experimental data is denoted as $\{(c_i, s_{ij}); i = 1, 2, ..., 15; j = 1, 2, 3\}$. After
removing the background from spectra, PCA reduces the dimensions of background-free spectra to ten. Then the final inverse model is estimated upon the estimation data set \( \{(\gamma_i, c_i); i = 1, 2, ..., 45\} \) (Section 3.3.2). Table 5.6 shows the estimated parameters of the inverse calibration model.

Table 5.6: The DK parameters of the inverse model based on initial design for case 2

<table>
<thead>
<tr>
<th>Inverse Model for ( c_1 )</th>
<th>( \hat{\beta}_0 )</th>
<th>( \hat{\theta}_1 )</th>
<th>( \hat{\theta}_2 )</th>
<th>( \hat{\theta}_3 )</th>
<th>( \hat{\theta}_4 )</th>
<th>( \hat{\theta}_5 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.2677</td>
<td>2.1095</td>
<td>1.3797</td>
<td>1.9474</td>
<td>1.4142</td>
<td>2.3469</td>
<td>( \hat{\theta}_6 )</td>
</tr>
<tr>
<td>0.0838</td>
<td>0.0593</td>
<td>0.0419</td>
<td>0.0397</td>
<td>0.0562</td>
<td>( \hat{\theta}_6 )</td>
<td>( \hat{\theta}_7 )</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Inverse Model for ( c_2 )</th>
<th>( \hat{\beta}_0 )</th>
<th>( \hat{\theta}_1 )</th>
<th>( \hat{\theta}_2 )</th>
<th>( \hat{\theta}_3 )</th>
<th>( \hat{\theta}_4 )</th>
<th>( \hat{\theta}_5 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.9465</td>
<td>2.0000</td>
<td>2.6816</td>
<td>0.2781</td>
<td>0.6026</td>
<td>0.5563</td>
<td>( \hat{\theta}_6 )</td>
</tr>
<tr>
<td>0.1154</td>
<td>0.0861</td>
<td>0.0642</td>
<td>0.0642</td>
<td>0.1547</td>
<td>( \hat{\theta}_6 )</td>
<td>( \hat{\theta}_7 )</td>
</tr>
</tbody>
</table>
5.2.3 Sampling-based Evaluation

The efficiency of the calibration procedure is evaluated in two ways: (1) For one macro-replication, the estimated variability of concentration estimates satisfy the desired calibration precision or not; (2) For a large number of macro-replications \( M = 100 \), the reliability of obtained calibration model is good or not.

To avoid bias, we define a new type of data called validation data set (VDS), which distinct from the estimation data obtained through the sequential DOE. These two types of data sets both generate from experimental sampling (simulation experiments in this paper), but VDS is independent of estimation data which is not used to fit forward and inverse models. We create a large VDS, denoted as \( \{(c_{g}^{(Grid)}, s_{g}); g = 1, 2, ..., 400\} \), with \( 20 \times 20 \) grid points ranging over the feasible region \( C \) and \( s_{g} \) being a simulated spectrum at the grid point \( c_{g}^{(Grid)} \).

Figure 5.4 shows the evaluation results in the first way (one macro-replication). Four example grid points, marked as ”x” in the figure, are selected for illustration; each of them represents a true concentration combination for an unknown sample, on which the ”observed” simulated spectrum could be obtained. Feeding this spectrum to the calibration model (Table 5.6), the point estimate of this unknown concentration vector is generated. By applying Algorithm 3 and Formula (3.28), the variability of this concentration estimation is obtained. The dashed rectangular around ”x” represents the estimated uncertainty of concentrations estimation for two materials. It shows that the estimated variability region (dashed rectangular) cover the true value. For any other grid point, the evaluation procedure has been performed in the same manner as that for those four grid points in Figure 5.4, and similar results occurred.

It was proved by Figure 5.4 that the calibration procedure provides precise estimated variability of model estimates by means of the first way (one macro-replication), and Figure 5.5 displays the evaluation results based on \( M = 100 \) macro-replications. The dot clouds in Figure 5.5 are the \( M = 100 \) estimated concentration vectors for the four example points. To
Figure 5.4: VDS-based evaluation results for one macro-replication

judge if the desired calibration precision is achieved consistently every time this procedure is used, compare these concentration estimates (dot cloud) to the required region specified by 
\[
[c_p^{(f)} - z_{1-\alpha/2} \times \delta_p, c_p^{(f)} + z_{1-\alpha/2} \times \delta_p] (p = 1, 2),
\]
which are depicted as the solid squares in Figure 5.5. Each square represents a region within which an estimated concentration vector is required to fall. All four groups of macro-replicated concentration estimates are inside of the corresponding solid squares, which shows that the desired precision has been met stably.

In order to test the quality of estimated variability of fitted concentration vector (PI) over \( M = 100 \) macro-replications, we calculate the coverage probability of the PIs at 5 check point as an example (Table 5.7). The coverage probability is the same as what is defined in case 1 (Section 5.1.3): the number of PIs including the true concentration vector divided by the toal number of PIs. Although the coverage probabilities are all 1.00 instead of numbers around 0.95, it is still acceptable.
Figure 5.5: VDS-based evaluation results for 100 macro-replications

Table 5.7: Coverage probability in case 2

<table>
<thead>
<tr>
<th>Concentration</th>
<th>PI</th>
</tr>
</thead>
<tbody>
<tr>
<td>[500 500]</td>
<td>100%</td>
</tr>
<tr>
<td>[1000 500]</td>
<td>100%</td>
</tr>
<tr>
<td>[1000 1000]</td>
<td>100%</td>
</tr>
<tr>
<td>[1500 1000]</td>
<td>100%</td>
</tr>
<tr>
<td>[1900 1900]</td>
<td>100%</td>
</tr>
</tbody>
</table>
Chapter 6

Summary

To enable multiplexed analyte quantification based on Raman spectra, calibration models need to be developed quantifying the relationship between analyte concentrations and Raman spectra. The accuracy and precision of the target estimates (estimated analyte concentrations) directly depend on the quality of the calibration models. In this work, a statistical procedure was developed to use the least experimental effort to generate calibration models of the highest quality. There are two important components in this procedure: sequential design of experiments (DOE) and calibration modeling. With the limited sample size, chemometric experiments need to be designed in such a way that sampling data collected are the most informative leading to the calibration model of the highest quality. To assess model quality/uncertainty, which is the basis for DOE, stochastic kriging is adapted accommodating non-i.i.d errors to model the relationship of interest as well as to quantify the estimation uncertainty (precision).

Through the two substantial simulation-based studies, the efficiency of this calibration procedure is demonstrated to be able to provide calibration models of pre-specified accuracy and precision. Compared to the currently used peak-based method with the same experimental budget, our procedure generates calibration model of higher and stable quality.
References


Appendix A

Simulation Model

The real data involved in this paper is from Zhu et al [10]. It includes the study of the mixture of two materials: ferbam and thiram. As shown in Table A.1, there are 18 different concentration combinations. At each concentration combination, 10 Raman spectra are observed at the range of Raman shift [530, 1600] cm$^{-1}$.

Table A.1: Concentration combinations

<table>
<thead>
<tr>
<th>(c$_1$,c$_2$) nM</th>
<th>(c$_1$,c$_2$) nM</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1000, 0)</td>
<td>(500, 0)</td>
</tr>
<tr>
<td>(0, 1000)</td>
<td>(0, 500)</td>
</tr>
<tr>
<td>(100, 0)</td>
<td>(20, 100)</td>
</tr>
<tr>
<td>(0, 100)</td>
<td>(220, 110)</td>
</tr>
<tr>
<td>(2000, 0)</td>
<td>(192, 176)</td>
</tr>
<tr>
<td>(0, 2000)</td>
<td>(150, 550)</td>
</tr>
<tr>
<td>(200, 0)</td>
<td>(400, 700)</td>
</tr>
<tr>
<td>(0, 200)</td>
<td>(800, 1400)</td>
</tr>
<tr>
<td>(0, 50)</td>
<td>(50, 0)</td>
</tr>
</tbody>
</table>

The real data used in this paper is far from well designed to provide a fair coverage of the design space spanned by the concentrations of mixture. Therefore, a simulation model was built to demonstrate and assess the proposed calibration procedure. The simulation model was derived from real data (Section 2) and designed to generate spectra possessing the characteristics of real ones. The simulation model serves two purposes in this dissertation: generating the data via computer experiments for modeling and providing the true benchmark to compare the quality of the resulting calibration model by a certain method.

As mentioned in Chapter 2, the intensity of spectrum consists of three components: background $Y^{(b)}$, signal $Y^{(s)}$, and noise $\varepsilon$. Therefore, the simulation model also include three parts to simulate each component of the spectrum: $Y^{(b)} = BF(x, c)$, $Y^{(s)} = RS(x, c)$, and
ε = NE(x, c). Given the concentration vector c of a mixture, the intensity values of the spectrum over Raman Shifts x can be simulated by Equation (A.1).

\[ y = SS(x, c) = BG(x, c) + NE(x, c) + RS(x, c). \]  

(A.1)

The estimation of the three models is illustrated in detail in Section A.1, A.2.1, and A.3 respectively. Based on three portions of the simulation model, a spectrum can be simulated at any concentration vector of the mixture, leading to the data \( \{(c_i, s_i); i = 1, 2, \ldots\} \) mimicking the real experimental data. To build the simulation model, raw spectral data needs to be preprocessed to be decomposed into the background, noise and signal data.

A.1 Simulate the Background

By applying Algorithm 1, the background data to build the background portion of simulation model is obtained as \( \{(c_i, s_i^{(b)}); i = 1, 2, \ldots\} \), where \( s_i^{(b)} = (x, Y^{(b)}) \). Then deterministic Kriging (DK)[32] is employed to build the background portion of simulation model. The intensity of the background curve at a setting \( w^{(b)} = (x, c) \) is modeled as

\[ Y^{(b)} = BG(w^{(b)}) = f(w^{(b)})^\top \beta + M(w^{(b)}), \]  

(A.2)

where \( f(w^{(b)}) \) is a vector of known functions of \( w^{(b)} \), \( \beta \) is a vector of unknown parameters of compatible dimension, and \( M(w^{(b)}) \) is a mean-zero stationary Gaussian process. This work adopts \( f(w^{(b)})^\top \beta = \beta_0 \), which has been widely accepted.

When applying DK to the background data set, the Gaussian correlation function (Equation (A.3)) is used to model the correlations between input variables:

\[ \text{Corr}[M(w^{(b)}), M(w^{(b)'})] = \exp\left\{ \sum_{d=1}^{D} -\theta_d |w_d - w_d'|^2 \right\}, \]  

(A.3)

where \( D \) is the dimension of factor setting \( w \), and \( \theta = (\theta_1, \theta_2, ..., \theta_D) \) is a vector of unknown parameters.

In this case, there are two analytes in the mixture, which means \( D = 3 \). The estimated parameters of the DK model are displayed in Table A.2.
Table A.2: The DK parameters of the background simulation model

<table>
<thead>
<tr>
<th>$\hat{\beta}_0$</th>
<th>$\hat{\theta}_1$</th>
<th>$\hat{\theta}_2$</th>
<th>$\hat{\theta}_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.724</td>
<td>1.3127</td>
<td>1.7102</td>
<td>0.6375</td>
</tr>
</tbody>
</table>

A.2 Extracting Noise and Signal Data

After separate the background curve, Savitzky-Golay(SG) filter was applied to smooth out the noise of spectrum. This filter could reduce the noise while maintaining the shape and height of peaks. The basic idea of SG filter is fitting a least-squares polynomial function for the selected point and some of nearby neighbors. For example, when obtaining the signal value $Y_j^{(s)}$ of $j$th point, a data set including $Y_{j-M}^{(s)}, Y_{j-M+1}^{(s)}, ..., Y_j^{(s)}, Y_{j+1}^{(s)}, ..., Y_{j+M}^{(s)}$ were selected to fit a $P$ order polynomial function. Repeat this process $J$ times to calculate $Y_j^{(s)}$ for every points on the background free spectrum. Then the signal curve denoted as $s^{(s)}$ and noise denoted as $s^{(e)}$ are separated from the rest of the spectrum (Figure A.1).

The derived signal data to build the signal portion of simulation model is $\{(c_i, s_i^{(s)}); i = 1, 2, ...\}$, where $s^{(e)} = (x, Y^{(s)})$. And the noise data is $\{(c_i, s_i^{(e)}); i = 1, 2, ...\}$, where $s^{(e)} = (x, \varepsilon)$.

![Figure A.1: The spectrum before and after Savitzky-Golay filtering](image-url)
A.2.1 Simulate the Noise

By plotting the noise data obtained from Section A.2, it is shown that the noise data are not identically independently distributed (Figure A.2). To retain this characteristic of noise data, the Auto-Regressive Moving Average (ARMA) model is employed to build the noise portion of simulation model (Model A.4):

\[ \varepsilon_x = \text{NE}(x, c) = 0.7\varepsilon_{x-1} - 0.25\varepsilon_{x-1} + \varepsilon_x, \]

where \( \varepsilon_x \in N(0, \sigma(c)^2) \). Herein, \( \sigma(c) \) is related to the concentration vector \( c \). Then single-hidden layer neural network is utilized to model the relationship between concentration vector \( c \) and parameter \( \sigma(c) \) (Model A.5):

\[
\hat{\sigma}(c) = 72.0367 + \frac{71.6875}{1 + \exp(-4.0446 * c_1 + 0.1790 * c_2 - 4.9289)} - \frac{3.6615}{1 + \exp(12.9456 * c_1 - 4.9159 * c_2 - 11.3179)}.
\]

Figure A.2: Noise v.s. Raman shift
A.3 Simulate the Signal

Lorentzian peaks simulation [33], which is a popular method to characterize the spectral curve shape observed in spectroscopy, is employed to simulate the signal curve $s^{(s)}$ on null background and null noise. The expression is as follows:

$$Y^{(s)} = RS(x,c) = \sum_{j=1}^{J^*} \frac{2A_j(c)}{\pi} \frac{\omega_j}{4(x - u_j)^2 + \omega_j^2},$$  \hspace{1cm} (A.6)

where $J^*$ is total number of peaks, $u_j$ is the position of peaks, $\omega_j$ is the bandwidth of peaks at the full width half-maximum (FWHM), and $A_j(c)$ is the area under peaks. The $J^*$, $u_j$, $\omega_j$, and $A_j(c)$ are determined by the peak detection method on the paper [20].

According to the property of spectrum, $J^*$, $u_j$, $\omega_j$ are fixed for the same analytes in the mixture no matter the concentration values. While $A_j(c)$ is varying with the different concentration vector. The single-hidden layer neural network is applied to relate $A_j(c)$ to $c$. Given a concentration vector $c$, the parameter $A_j(c)$ is estimated by Equation (A.7):

$$A_j = 391720 + \frac{-1568400}{1 + exp(-0.0262 * u_j - 0.0002 * c_1 - 0.0002 * c_2 - 7.6761)} \frac{1089600}{1 + exp(-0.0066 * u_j - 0.0009 * c_1 - 0.0006 * c_2 - 13.9308)}$$

$$+ \frac{853300}{1 + exp(-0.0099 * u_j - 0.0001 * c_1 - 0 * c_2)} \frac{-72200}{1 + exp(-0.0272 * u_j + 0.0018 * c_1 + 0.0007 * c_2)}$$

$$+ \frac{289500}{1 + exp(0.0012 * u_j - 0.0011 * c_1 - 0.0007 * c_2)}.$$

The second and third columns in Table A.3 show the values of $u_j$ and $\omega_j$ for $J^* = 10$ peaks. As an example, the fourth column gives the values of $A_j$ of the sample with the concentration vector [1000,0].

Given the Table A.3, an example of the simulated pure signal curve $s^{(s)}$ at a setting of concentration vector [1000,0] is shown in Figure A.3

A.4 Simulate a Spectrum

Given the three portions of simulation model above, we can simulate a spectrum with a certain concentration vector $c_0$ by the procedure described in Algorithm 7. When generating
Table A.3: Lorentzian function parameters

<table>
<thead>
<tr>
<th>j</th>
<th>( u_j )</th>
<th>( \omega_j )</th>
<th>( A_j )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1512.4</td>
<td>30.375</td>
<td>4880.3</td>
</tr>
<tr>
<td>2</td>
<td>1443.1</td>
<td>19.708</td>
<td>792.7</td>
</tr>
<tr>
<td>3</td>
<td>1382.4</td>
<td>17.663</td>
<td>11388</td>
</tr>
<tr>
<td>4</td>
<td>1238.2</td>
<td>19.799</td>
<td>585.6</td>
</tr>
<tr>
<td>5</td>
<td>1145.4</td>
<td>26.099</td>
<td>4135.8</td>
</tr>
<tr>
<td>6</td>
<td>1044.1</td>
<td>13.679</td>
<td>222.0</td>
</tr>
<tr>
<td>7</td>
<td>933.7</td>
<td>28.389</td>
<td>2589.4</td>
</tr>
<tr>
<td>8</td>
<td>870.1</td>
<td>20.116</td>
<td>482.0</td>
</tr>
<tr>
<td>9</td>
<td>690.0</td>
<td>61.659</td>
<td>1471.0</td>
</tr>
<tr>
<td>10</td>
<td>561.8</td>
<td>16.148</td>
<td>5278.0</td>
</tr>
</tbody>
</table>

Figure A.3: Simulating signal
the noise in Step (2) in Algorithm 7, the common random number(CRN) technique [34] is utilized. To generate \( n \) replication of concentration \( c_0 \), simply repeat the step (2) \( n \) times by applying different random seed to generate \( \varepsilon \).

**Algorithm 7: Simulate a spectrum**

**Input:** (a) the concentration vector \( c_0 = (c_1, c_2, ..., c_P) \) and Raman Shift \( x = (x_1, x_2, ..., x_J) \); (b) the simulation model including three portions: background \( \text{BG}(x, c) \), noise \( \text{NE}(x, c) \) and signal \( \text{RS}(x, c) \);

**Step (1):** Simulate the intensity of background curve \( Y^{(b)} \) at each Raman Shift by the formula A.2

**Step (2):** Simulate the intensity of noise curve \( \varepsilon \):

- Calculate the variance of random fluctuation \( \sigma \) by formula A.5.
- Set \( \varepsilon_0 = 0 \), calculate the rest of noises \( \varepsilon_x \) for \( x = 1, 2, ..., J \) by formula A.4.

**Step (3):** Simulate the intensity of signal curve \( Y^{(s)} \):

- Calculate the parameter \( A \) by formula A.7
- Calculate the \( Y^{(s)} \) by formula A.6

**Step (4):** Obtain \( y \) by adding three parts together: \( y = Y^{(b)} + Y^{(s)} + \varepsilon \).

**Output:** The simulating spectrum \( \{(x_j, y_j); j = 1, 2, ..., J\} \) of the mixture with the concentration \( c_0 \)