Maximum correntropy criterion based regression for multivariate calibration

Jiantao Peng\textsuperscript{a,∗}, Lu Guo\textsuperscript{a}, Yong Hu\textsuperscript{b}, KaiFeng Rao\textsuperscript{c}, Qiwei Xie\textsuperscript{d,e}

\textsuperscript{a} Faculty of Mathematics and Statistics, HuBei Key Laboratory of Applied Mathematics, HuBei University, WuHan 430062, China
\textsuperscript{b} Beijing Research Institute of Uranium Geology, Beijing 100029, China
\textsuperscript{c} Research Center for Eco-Environment Sciences, Chinese Academy of Sciences, Beijing 100085, China
\textsuperscript{d} Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China
\textsuperscript{e} CAS Center for Excellence in Brain Science and Intelligence Technology, Shanghai Institutes for Biological Sciences, Chinese Academy of Sciences, China

\textbf{A R T I C L E  I N F O}

\textbf{Keywords:}
- Maximum correntropy criterion
- Least-squares
- Multivariate calibration
- Regularization

\textbf{A B S T R A C T}

The least-squares criterion is widely used in the multivariate calibration models. Rather than using the conventional linear least-squares metric, we employ a nonlinear correntropy-based metric to describe the spectra-concentrate relations and propose a maximum correntropy criterion based regression (MCCR) model. To solve the correntropy-based model, a half-quadatic optimization technique is developed to convert a non-convex and nonlinear optimization problem into an iteratively re-weighted least-squares problem. Finally, MCCR can provide an accurate estimation of the regression relation by alternatively updating an auxiliary vector represented as a nonlinear Gaussian function of fitted residuals and a weight computed by a regularized weighted least-squares model. The proposed method is compared to some modified PLS algorithms and robust regression methods on four real near-infrared (NIR) spectra data sets. Experimental results demonstrate the efficacy and effectiveness of the proposed method.

1. Introduction

Multivariate calibration (MVC) is an effective tool to reveal the intrinsic quantitative relations between the spectra and corresponding concentrations by means of a regression model [1]. The most commonly used MVC model is partial least squares (PLS) [2], which has shown excellent performance in high-dimensional small-sample and collinear problems [2]. A simple and effective PLS implementation method is nonlinear iterative partial least squares (NIPALS) [2], which extracts the PLS components by successively updating the weights, loadings, and scores. In NIPALS, the weight describing the covariance structure between the predictor variables X and response y is first calculated and then used to solve subsequent loadings and scores. When the expected PLS components are extracted, the regression coefficients can be represented in terms of the weights and loadings.

Due to the iterative nature of NIPALS, its first step of weight computation is crucial, which directly affects the subsequent loadings and scores. That is, the weight describing the y-X relations is a key factor. In the past few years, many methods have been proposed to improve the y-X weight relations [3–7]. In the orthogonal projections to latent structures (O-PLS) algorithm [3], the orthogonal weights are involved to remove the non-correlated systematic variation. In the PLS with orthogonal scatter correction (PLS2OSC) [4], the non-correlated systematic variation is removed, which improves the interpretation of PLS and reduces the model complexity. In the Power PLS (PPLS) [5], the weight is computed by taking powers of the y-X correlations and X standard deviations, which neutralizes the influence of dominance of irrelevant X-variance and spurious y-correlations [5]. In sparse matrix transform based PLS (SMTPLS) [6], a sparse matrix transform technique is first used to decorrelate the observation data, then the weight is computed in the decorrelated data space by the least-squares regression. The SMT decorrelation operation can alleviate the effect of correlated variables on the least-squares computation of the PLS weight. For sparse PLS [7], an 1-norm penalization is imposed on the weight vectors, which leads to a sparse solution for variable selection.

The aforementioned methods have shown good performance in some specified applications. However, these algorithms compute the y-X relations based on the least-squares criterion, which is very sensitive to outliers [8]. The quadratic loss function in the least-squares framework measures data and noise on the same norm scale and does not differentiate noise from data [9]. Therefore, the obtained weight vector may also reflect the effect of noisy samples, in the case of a contaminated data. For spectroscopic data, the noise or outlier is usually encountered during the data measurement and acquisition,
such as instrument noise or abnormality, environmental noise, and so on [10]. These noise or outliers will make the least-squares-based weight vector distorted. Recently, a correntropy-based metric is proposed for robust classification and regression, and has shown excellent performance for handling noisy data [9,11]. Compared with the widely used quadratic function, correntropy is a local metric, which measures the similarity of two random variables within a small neighborhood determined by the kernel width. Depending upon the kernel width, the correntropic function can winnow noise from the data [9]. The localization provided by the kernel width proves to be very useful in reducing the detrimental effects of outliers and noise [9].

Motivated by the superiority performance of correntropy in robust estimation, we employ a correntropy-based metric to replace the traditional least-squares metric in measuring the spectra-concentrate y-X weight relations and propose a maximum correntropy criterion based regression (MCCR) model for multivariate calibration. To solve the correntropy-based model, a half-quadratic optimization technique is developed to convert the original non-convex and nonlinear optimization problem into an iteratively re-weighted least squares problem.

2. The algorithm

2.1. Partial least squares (PLS)

Let \( X = [x_1, \ldots, x_n]^T \) be the \( n \times p \) data matrix with each sample \( x_i \in \mathbb{R}^p \), and \( y = [y_1, \ldots, y_n]^T \) be the \( n \times 1 \) response vector. The PLS method, which in its classical form is based on the NIPALS algorithm [2], estimates the weight vector \( w \in \mathbb{R}^p \) such that:

\[
[\text{cov}(t, y)]^2 = [\text{cov}(Xw, y)]^2 = \max_{\|w\|=1} [\text{cov}(Xr, y)]^2.
\]

When the PLS weight vector is computed, the NIPALS method can resolve the PLS loadings and scores successively, and finally outputs the regression coefficient vector [2]. In detail, the NIPALS algorithm repeats a sequence of the following steps until convergence:

\[
\begin{align*}
(1) & \quad w = X^Ty/\|X^Ty\| \\
(2) & \quad t = Xw \\
(3) & \quad p = X^Tu(t^Tt) \\
(4) & \quad q = y^T(u^Tt) \\
(5) & \quad X = X - tp^T
\end{align*}
\]

From the whole process of NIPALS, it can be seen that the PLS algorithm contains four regression steps, step (1), (2), (3) and (4). Each of these regression equations is actually an ordinary least-squares regression [12], which is sensitive to outliers [8]. In the presence of noise or outliers, the estimated weights, loadings, and scores may be distorted. In order to eliminate these effects, a robust correntropy-based regression model is proposed.

2.2. Maximum correntropy criterion (MCC)

From the information theoretic learning (ITL) point of view [9], the concept of correntropy was proposed to process non-Gaussian noise. It is a generalized similarity measure between two arbitrary random variables \( A \) and \( B \), defined as

\[
\mathcal{V}_\sigma(A, B) = E[k_\sigma(A - B)],
\]

where \( k_\sigma(\cdot) \) is a kernel function and \( E \) is the expectation operator. Usually, \( k_\sigma \) is chosen as the Gaussian kernel \( k_\sigma(x) = \exp(-\|x\|^2/(2\sigma^2)) \) with a kernel width parameter \( \sigma \).

In practice, the joint probability density function of \( A \) and \( B \) is often unknown and only a finite number of empirical data \( \{(a_i, b_i)\}_{i=1}^N \) are available. In this case, the correntropy defined in (2) is approximated by the following empirical correntropy:

\[
\tilde{\mathcal{V}}_\sigma(A, B) = \frac{1}{N} \sum_{i=1}^N k_\sigma(a_i - b_i).
\]

The correntropy can induce a metric for any two vectors \( A = (a_1, \ldots, a_n)^T \) and \( B = (b_1, \ldots, b_n)^T \) as follows:

\[
\text{CIM}(A, B) = \sqrt{\sum_{i=1}^N k_\sigma(a_i - b_i)}.
\]

If we denote the error \( e_i = a_i - b_i \), the maximum correntropy criterion (MCC) aims to maximize the below correntropy of error [9]:

\[
\max \frac{1}{n} \sum_{i=1}^n k_\sigma(e_i) = \frac{1}{n} \sum_{i=1}^n k_\sigma(a_i - b_i).
\]

Compared with the global mean squared error (MSE) metric, correntropy is a local metric, which measures the similarity of two random variables within a small neighborhood determined by the kernel width \( \sigma \). The value of correntropy is mainly decided by the kernel function along the line \( A = B \) [9]. The localization provided by the kernel width proves to be very useful in reducing the detrimental effects of outliers and impulsive noise [9].

2.3. Maximum correntropy criterion based regression (MCCR)

According to the Beer–Lambert law, there is a linear relationship between spectra \( X \) and corresponding concentration response \( y \) when the pathlength keeps constant, which motivates the development of linear multivariate calibration (MVC) techniques. The essential linear relation is:

\[
y = Xw,
\]

where \( w \) can be regarded as regression coefficient (or weight) and can be solved by the following multiple least-squares regression:

\[
\min_w \| y - Xw \|^2.
\]

Considering that the least-squares criterion in (6) is sensitive to noise and outliers, we propose a robust maximum correntropy criterion (MCCR) to replace the least-squares criterion as follows:

\[
J(w) = \max_w \sum_{i=1}^n k_\sigma(y_i - X_i^Tw) = \sum_{i=1}^n \exp\left(-\frac{(y_i - X_i^Tw)^2}{2\sigma^2}\right).
\]

The optimization problem (7) can be changed to its equivalent form:

\[
J(w) = \min_w \sum_{i=1}^n -\exp\left(-\frac{(y_i - X_i^Tw)^2}{2\sigma^2}\right).
\]

It can be seen that the objective function in (8) is nonlinear and nonconvex. So, it is difficult to be directly optimized. Here, we use the half-quadratic technique [13] to solve the optimization problem (8).

According to the property of conjugate functions [14], we have:

\[
\text{Proposition 1}. \quad \text{There exists a convex conjugate function } \varphi \text{ of } k_\sigma(x) = \exp(-\frac{\|x\|^2}{2\sigma^2}), \text{ such that,}
\]

\[
-k_\sigma(x) = \min_a \left( \frac{\|z\|^2}{2\sigma^2} + \varphi(-a) \right)
\]

and for a fixed \( z \), the minimum is reached at \( a = k_\sigma(z) \).

Substituting (9) into (8), we have the augmented objective function in an enlarged parameter space as follows:

\[
J(w, \alpha) = \min_{w, \alpha} \sum_{i=1}^n \left( \frac{(y_i - X_i^Tw)^2}{2\sigma^2} + \varphi(-\alpha_i) \right).
\]

where \( \alpha = (\alpha_1, \alpha_2, \ldots, \alpha_n)^T \) is the auxiliary vector in the half-quadratic optimization technique.

According to the Proposition 1, for a fixed \( w \), we can obtain the solution of auxiliary variables as
\[ \hat{a}_i = \exp \left( \frac{(y_i - x_i^T w)^2}{2\sigma^2} \right) \]  \hspace{1cm} (11)

For a fixed \( \alpha \), the vector \( w \) can be solved by a weighted least-squares model as follows:

\[ \min_w \sum_{i=1}^n \alpha_i \left( y_i - x_i^T w \right)^2 = (y - Xw)^T A(y - Xw), \]  \hspace{1cm} (12)

where \( A \) is a diagonal matrix whose diagonal elements are \( A_{ii} = \alpha_i/(2\sigma^2) \), \( i = 1, \ldots, n \). It can get \( w \) easily from (12):

\[ \hat{w} = (X^T A X + \lambda I)^{-1} X^T A y, \]  \hspace{1cm} (13)

In the practice, when the number of variables is larger than the number of samples, the least-squares solution in (13) is usually unstable. To improve the stability of the solution, a regularized least-squares solution can be used:

\[ \hat{w} = (X^T A X + \lambda I)^{-1} X^T A y, \]  \hspace{1cm} (14)

where \( \lambda \) is a regularization parameter and \( I \) is an identity matrix.

In fact, the above optimization procedure is an iteratively re-weighted least-squares problem. By alternatively updating the auxiliary vector \( \alpha \) and the weight vector \( w \), the algorithm finally converges and can output the desired weight vector. It should be mentioned here that the objective function \( J(w, \alpha) \) in (10) is non-increasing and the MCC algorithm converges after a certain number of iterations, which can be summarized as follows:

**Proposition 2.** Let \( w(t) \) and \( a(t) \) denote the solutions at \( t \)-th iteration, the sequence \( J(w(t), a(t)) \) converges.

The proof of Proposition 2 is provided in the Appendix.

When the vector \( \hat{w} \) is obtained, we can get the prediction for the measure data \( X \) as: \( \hat{y} = X \hat{w} \). However, in practice, there usually has a deviation between the true response \( y \) and the predicted response \( \hat{y} \). Inspired from the PLS, we consider \( \hat{w} \) as a weight describing the relation between the spectra and the response, and denote \( t = X \hat{w} \) as the principal component, then regress the true response \( y \) on the principal component \( t = tq \), where \( q \) can be considered as an adjustment factor to make the predicted response \( \hat{y} \) or \( t \) be as close as possible to the true response. Finally, we can obtain the maximum correntropy criterion based regression (MCCR) model: \( y = tq = Xwq \). We denote \( \hat{B} = \hat{w}q \) as regression coefficient.

A key factor in the MCCR is the design of kernel parameter \( \alpha \), which directly decides the localization ability of Gaussian kernel correntropy metric. The kernel width parameter \( \sigma \) is adaptively updated during each iteration as follows:

\[ \sigma^2 = \frac{1}{n} \sum_{i=1}^n (\xi_i - x_i^T \hat{w})^2, \]  \hspace{1cm} (15)

which is the average of approximation errors.

The detailed procedure of the proposed MCCR algorithm is summarized in Algorithm 1.

**Algorithm 1. MCCR algorithm. Training process:**

1: Compute the MCCR weight \( w \):

- Initialize the weight: \( w^{(0)} = (X^T X + \lambda I)^{-1} X^T y \).
- Set \( t = 1 \), iteratively run following steps until convergence
  1.1: Update the kernel width: \( \sigma^{(t-1)} \sigma^2 = \sum_{i=1}^n (\xi_i - x_i^T w^{(t-1)})^2/ln \) by (15).
  1.2: Update the auxiliary vector \( a^{(t)} \) by (11).
  1.3: Update the weight vector \( w^{(t)} \) by (14).
  1.4: \( t = t + 1 \), go to step 1.1.
- Output the MCCR weight \( w = w^{(t)} \).

2: Calculate the score \( t = Xw \), and regress \( y \) on \( t: y = tq \).

3: The resulting regression model is: \( y = tq = Xwq \), and \( \hat{B} = wq \) is the regression coefficient.

**Prediction process:** For new data \( X_{new} \), the prediction is:

\[ y_{new} = X_{new} B = X_{new} wq. \]

### 3. Experimental

#### 3.1. Data sets

Data set 1 (Tablet) consists of NIR spectra from 310 pharmaceutical tablet samples with a relative active substance content (%w/w) in the range 4.6–9.8% [15]. The transmittance spectra have 404 variables collected in the range 7400–10507 cm\(^{-1}\).

Data set 2 (Corn) consists of 80 NIR spectra of corn measured on spectrometers m5 [16]. The spectra are collected in 2 nm intervals within the spectral range 1100–2498 nm. Reference values include moisture, oil, protein and starch contents in the range 9.38–10.99%, 3.09–3.83%, 7.65–9.71% and 62.83–66.47%, respectively.

Data set 3 (Tablet 2002) is the basis of the Software ShootOut 2002 session [18]. It contains 655 pharmaceutical tablets, which are divided into a calibration set (155 spectra) and test set (460 spectra). The data includes tablets with a wide ASSAY range, 152–239 mg, for developing a calibration. The spectral region of interest for calibration for the ASSAY content is 1050–1630 nm [17].

Data set 4 (Meat) is the NIR transmittance spectra of meat samples [19]. The spectra have been recorded on a Tectator Infratec Food and Feed Analyzer working in the wavelength range 850–1050 nm. For each sample, the data consists of a 100 channel absorbance spectrum and the contents of moisture, fat and protein. These three contents are determined by analytic chemistry. The data contains 172 training samples and 43 testing samples.

The spectra of the aforementioned four data sets are shown in Fig. 1. It should be noted that the Data Set 3 (Tablet 2002) contains some known outliers, such as regression outliers and spectral outliers [17], while Data Sets 1, 2 and 4 are known not to contain outliers.

#### 3.2. Model selection

The proposed MCCR algorithm is compared with the following algorithms: (1) the original and some modified PLS algorithms, such as radial basis functions PLS (RBPLS) [20], powered PLS (PPLS) [5], PLS with orthogonal scatter correction (PLS2OSC) [4], sparse matrix transform based PLS (SMTPLS) [6], continuum power regression (CRP) [21]; (2) some robust regression algorithms, such as robust continuum regression (RCR) [22], partial robust M-regression (PRM) [23], and robust SIMPLS (RISIMPLS) [24]. All calculations are conducted using MATLAB R2014a on a computer with Intel Core i5-3570 3.4 GHz CPU and 8 GB RAM. The Matlab codes for RBPLS, CPR, RCR, PRM come from the TOMCAT toolbox [25] at http://chemometria.us.edu.pl/RobustToolbox/. The Matlab codes for PPLS and PLS2OSC are available in the corresponding references [4,5]. The Matlab codes for RISIMPLS are available at MATLAB toolbox LIBRA https://wis.kuleuven.be/stat/robust/LIBRA [29]. All data are mean-centered. Each data set is divided into a calibration set and a prediction set. The calibration set is used to build the calibration model and also to select algorithm parameters, and the prediction set is only used for independent prediction. The model is evaluated by the root mean square errors of prediction (RMSEP) on the prediction set.

In the experiments, the continuum regression parameter \( \delta \) is fixed as 0.5 for the CPR and RCR. When \( \delta = 0.5 \), the CPR and RCR reduce to PLS and robust PLS, respectively [21,22]. It should be noted that RCR reduces to robust PLS by projection pursuit, which is still different from either PRM or RISIMPLS. For SMTPLS, the number of rotations is fixed as 500 [6]. For PPLS, the lower bound of parameter \( \gamma \) is fixed to be 0,
and the upper limit of parameter $\gamma$ should be chosen carefully [5]. Besides the above-mentioned fixed parameters, all of the RBFPLS, PPLS, PLS2OSC, RCR, PRM, RSIMPLS algorithms have two parameters, in which one is the number of latent variables ($p_1$) in common and another is the algorithm’s built-in tuning parameter ($p_2$). The number of latent variables varies from 1 to 15. In RBFPLS, the tuning parameter $\sigma$ specifying widths of Gaussian kernels varies in the set \{2$^{-4}$, 2$^{-3}$, ..., 2$^{-1}$\}. In PPLS, the upper bound of parameter $\delta$ is chosen in the range [0.9: 0.01: 1]. In PLS2OSC, the number of orthogonal components changes from 1 to 8. In RCR, the tuning parameter $\alpha$ controls the percentage of outliers in the data, which is chosen in the set \{0, 5%, 10%, ..., 35%\}. In PRM, the tuning constant $c$ of the weighting function is chosen in the set \{2$^0$, 2$^1$, ..., 2$^3$\}. In RSIMPLS, parameter $\alpha$ corresponding to the fraction of normal samples is chosen in the set \{50%, 55%, 60%, ..., 100%\}. In MCCR, the regularization parameter $\lambda$ is chosen in the set \{10$^{-1}$, 10$^{-2}$, ..., 10$^{-8}$\}.

We perform 10-fold cross-validation on the calibration set to pick the two parameters by grid search. In detail, we first compute the predicted residual error sum of squares (PRESS) values corresponding to different parameter pairs ($p_1$, $p_2$), then find the parameter pair ($p_1^\text{min}$, $p_2^\text{min}$) corresponding to the minimum PRESS. Finally, the optimal algorithm’s tuning parameter is chosen as $p_2^\text{min}$. While the optimal number of latent variables doesn’t set to the latent variable corresponding to the minimum PRESS, $p_1^\text{min}$. It is chosen based on a trade-off between minimizing the PRESS and limiting the model complexity. We denote the PRESS under $p_2^\text{min}$ as PRESS$_2$. The smallest model (fewest number of latent variables) such that the PRESS$_2$ for this model is not significantly greater than the minimum PRESS$_2$ is adopted. We use the F-test at 95% confidence level to test the significance of incremental changes in PRESS$_2$ [26,27].

4. Results and discussion

4.1. Tablet data set

The 310 tablet spectra are divided into 210 calibration samples and 100 prediction samples based on the Kennard-Stone (KS) algorithm [28]. The RMSEP of different algorithms under different numbers of latent variables are shown in Fig. 2, where the RMSEP is truncated at 1.5 for display. It can be seen that most of the algorithms show very bad RMSEP results under the first 2 or 3 latent variables, and need more than four latent variables to achieve acceptable results, while MCCR doesn’t need to select the number of latent variables. Although the minimum RMSEP of PLS, RBFLPLS, SMPLS, PLS2OSC, PPLS, PRM and RSIMPLS are almost the same with MCCR, it is very difficult to choose the number of latent variables for the PLS-based methods. Usually, the cross validation method is used to choose the number of latent variables. By using the 10-fold cross validation with an F-test at 95% confidence level, the optimal number of latent variable, the optimal algorithm’s tuning parameter and corresponding RMSEP for different
algorithms are shown in Table 1. In the table, the first value in the parentheses refers to the number of latent variables, and the second value in the parentheses refers to the additional algorithm's parameter, i.e., Gaussian kernel width $\sigma$ for RBFPLS, upper bound of parameter $\gamma$ for PPLS, the number of OSC components for PLS2OSC, percentage of outliers for RCR, the tuning constant $c$ for PRM, parameter $\alpha$ corresponding to the fraction of normal samples for RSIMPLS. Compared with different algorithms under the selected latent variable, the MCCR shows the best result.

Next, we investigate the difference between MCCR and PLS in terms of regression coefficient. We plot their regression coefficients in Fig. 3, where the number of latent variable for PLS is 4. It can be seen that the regression coefficient curve of MCCR is noisier than that of PLS due to the multiple regression operation in weight computation.

### 4.2. Corn data set

The 80 NIR spectra are divided into 60 calibration samples and 20 prediction samples based on the KS algorithm. The RMSEP of different algorithms under different number of latent variables for four contents (moisture, oil, protein, starch) are shown in Table 2. It can be seen that MCCR shows the best overall performance on the Corn data set. In detail, it provides the best results for the prediction of oil, protein, starch contents, and the second best result on the moisture content. It should be noted that MCCR doesn’t need to select the number of latent variables. However, other algorithms usually need more than 5 latent variables and need different numbers of latent variables for different contents. So, these algorithms need to select an optimal number of latent variable for each content separately. If the selected number of latent variables is inappropriate, the corresponding calibration model may be seriously deteriorated. While for MCCR, it is free of the selection of the number of latent variables.

### 4.3. Tablet 2002 data set

This pharmaceutical tablet dataset is the basis of the Software ShootOut 2002 session [18]. The data contains not only some univariate ASSAY regression outliers and also spectral outliers due to errors encountered during measuring [17]. We compare the proposed MCCR with other modified PLS algorithms and robust regression methods. The RMSEP of different algorithms under different numbers of latent variables are shown in Fig. 4. Almost all of the latent variable based algorithms are much more influenced by the number of latent variables and their RMSEPs change dramatically under different latent variables. Particularly, many algorithms show bad RMSEP results under the first three latent variables, and need more than four latent variables to achieve acceptable results. While MCCR doesn’t need to select the number of latent variables. In the following, we present the results under the optimal parameters. As we know, the pharmaceutical tablet dataset contains outliers. When outliers are present, the outliers also

| Method   | RMSEP  |
|----------|--------|
| PLS      | 0.3818 (4) |
| RBFPLS   | 0.4037 (10,2) |
| PPLS     | 0.3806 (4,0.92) |
| PLS2OSC  | 0.3771 (1,5) |
| SMTPLS   | 0.3746 (3) |

### Table 1

The optimal RMSEP of different algorithms for Tablet data set.

| Method   | RMSEP  |
|----------|--------|
| CPR      | 0.5531 (4) |
| RCR      | 0.6644 (4,35%) |
| PRM      | 0.3869 (4,2) |
| RSIMPLS  | 0.3906 (4,90%) |
| MCCR     | 0.3529 (10^-4) |

### Table 2

The optimal RMSEP of different algorithms for Corn data set.

| Content   | Moisture | Oil | Protein | Starch |
|-----------|----------|-----|---------|--------|
| PLS       | 0.0086 (13) | 0.0561 (12) | 0.1028 (12) | 0.1763 (13) |
| RBFPLS    | 0.0297 (12,28) | 0.0737 (14,26) | 0.1400 (13,28) | 0.3313 (12,28) |
| PPLS      | 0.0003 (6,1) | 0.0521 (12,94) | 0.1468 (12,0,09) | 0.1468 (12,93) |
| PLS2OSC   | 0.0068 (10,8) | 0.0324 (10,8) | 0.1028 (11,1) | 0.1192 (10,11,8) |
| SMTPLS    | 0.0351 (9) | 0.0531 (7) | 0.1013 (4) | 0.1689 (6) |
| CPR       | 0.2738 (6) | 0.1196 (6) | 0.2674 (6) | 0.9530 (5) |
| RCR       | 0.3024 (11,25%) | 0.1117 (10,15%) | 0.2456 (10,15%) | 1.0807 (8,5%) |
| PRM       | 0.0086 (13,28) | 0.0561 (12,92) | 0.1112 (13,23) | 0.1765 (13,28) |
| RSIMPLS   | 0.0357 (7,75%) | 0.0596 (8,95%) | 0.1333 (9,95%) | 0.2272 (9,100%) |
| MCCR      | 0.0056 (10^-8) | 0.0281 (10^-8) | 0.0716 (10^-6) | 0.1004 (10^-8) |

Fig. 3. The regression coefficient of PLS (a) and MCCR (b).
affect the cross-validation results for parameter selection. Therefore, the optimal parameters are chosen based on a robust PRESS rather than the classical PRESS. Here, the robust PRESS is represented as an $x\%$ trimmed PRESS, with $x\%$ covering the percentage of outliers. In this experiment, we set $x = 10$. The optimal number of latent variable and corresponding RMSEP for different algorithms are shown in Table 3, where MCCR provides the best result.

As this data contains many multivariate outliers [17], we employ the robust principal component analysis (RPCA) algorithm (available at MATLAB toolbox LIBRA [29]) to detect the outliers. It detects 85 outliers in the test set. After deleting these outliers, the prediction RMSEP of different algorithms are shown in Table 4. It can be seen that the performance of each algorithm largely improves when the outliers are eliminated. Now, the PRM algorithm provides the best result, and the proposed MCCR also achieves comparable prediction performance. In general, for a real problem, it is unclear whether the data contains outlier or not. If an algorithm can obtain good results for both noisy and clean data sets, the algorithm could be more practical in a variety of circumstances. In this sense, the proposed MCCR method would be promising.

4.4. Meat data set

We build calibration models for moisture, fat and protein contents, respectively. The optimal RMSEP of different algorithms for three contents are shown in Fig. 5, where the RMSEP is truncated at 3.5 for display. Here, for MCCR, the regularization parameter is set as $\lambda = 10^{-6}$. MCCR shows slightly better results than other algorithms on protein content, and provides the second best results on moisture and fat contents.

5. Conclusions

In this paper, we have proposed a maximum correntropy criterion based regression (MCCR) model for multivariate calibration. It employs a nonlinear correntropy-based metric to replace the traditional linear least-squares metric in modeling the spectra-concentrate relations. The nonlinear Gaussian function used in the MCC framework leads to an accurate estimation of the regression relation. Experimental results on four public NIR data sets demonstrate the effectiveness of the proposed MCCR.

Acknowledgements

This work was in part supported by the National Natural Science Foundation of China under Grants no. 41501392 and no. 11371007, by the Natural Science Foundation of Hubei Province under Grant no. 2009CDB387, by the Strategic Priority Research Program of the CAS (No. XDB02060001), and by the special fund of State Key Joint Laboratory of Environment Simulation and Pollution Control under...
Appendix A. Proof of Proposition 2

Proof 1. According to (11), because \( \alpha^{(i+1)} \) is the solution of minimizing \( J(w^{(i)}, \alpha) \), then we obtain:

\[
J(w^{(i)}, \alpha^{(i+1)}) \leq J(w^{(i)}, \alpha^{(i)}).
\]

Similarly, according to (12), because \( w^{(i+1)} \) is the solution of minimizing \( J(w, \alpha^{(i+1)}) \), we have:

\[
J(w^{(i+1)}, \alpha^{(i+1)}) \leq J(w^{(i)}, \alpha^{(i+1)}).
\]

Summarizing the Eqs. (16) and (17), we have

\[
J(w^{(i+1)}, \alpha^{(i+1)}) \leq J(w^{(i)}, \alpha^{(i)}).
\]

In this way, the cost function decreases at each alternating minimization step. Therefore, the sequence \( J(w^{(i)}, \alpha^{(i)}) \), \( i = 1, 2, \ldots \) is non-increasing. Moreover, it was shown in [9] that correntropy is bounded. Thus, it can be verified that \( J(w^{(i)}, \alpha^{(i)}) \) is also bounded. Consequently, we can conclude that the sequence \( J(w^{(i)}, \alpha^{(i)}) \) converges.

Appendix B. Supplementary data

Supplementary data associated with this article can be found in the online version at http://dx.doi.org/10.1016/j.chemolab.2016.12.002.