

# A robust PCR method for high-dimensional regressors

Mia Hubert\* and Sabine Verboven†

Revised version: November 28, 2002

SHORT TITLE: Robust Principal Component Regression

KEYWORDS: Principal Component Analysis, Principal Component Regression, Robust Regression, Multivariate Calibration

---

\* (Corresponding author) Assistant Professor, Department of Mathematics, Katholieke Universiteit Leuven, Celestijnenlaan 200B, B-3001 Leuven, Belgium, mia.hubert@wis.kuleuven.ac.be, tel. +32/16327048, fax. +32/16327998

† Assistant, Department of Mathematics and Computer Science, University of Antwerp, Universiteitsplein 1, B-2610 Wilrijk, Belgium, sabine.verboven@ua.ac.be.

# A robust PCR method for high-dimensional regressors

SHORT TITLE: Robust Principal Component Regression

KEYWORDS: Principal Component Analysis, Principal Component Regression, Robust Regression, Multivariate Calibration

## Abstract

We consider the multivariate calibration model which assumes that the concentrations of several constituents of a sample are linearly related to its spectrum. Principal component regression (PCR) is widely used for the estimation of the regression parameters in this model. In the classical approach it combines principal component analysis (PCA) on the regressors with least squares regression. Both stages yield however very unreliable results when the data set contains outlying observations. We present a robust PCR method which also consists of two parts. First we apply a robust PCA method for high-dimensional data on the regressors, and then we regress the response variables on the scores using a robust regression method. A robust RMSECV value and a robust  $R^2$ -value are proposed as exploratory tools to select the number of principal components. Also the prediction error is estimated in a robust way. Moreover we introduce several diagnostic plots which are helpful to visualize and classify the outliers. The robustness of RPCR is demonstrated through simulations and the analysis of a real data set.

# 1 Introduction

The main application of multivariate calibration is the prediction of constituents concentrations of a sample, based on its spectrum. This spectrum can be obtained via several techniques such as Fluorescence spectrometry, Near InfraRed spectrometry (NIR), Nuclear Magnetic Resonance (NMR), Ultra-Violet spectrometry (UV), Energy dispersive X-Ray Fluorescence spectrometry (ED-XRF), etc. Since a spectrum typically ranges over a large number of wavelengths, it is a high-dimensional vector with hundreds of components. The number of concentrations on the other hand is usually limited to at most, say, five. In the univariate approach, only one concentration at a time is modelled and analyzed. Here, we also consider the more general problem, where several concentrations together are to be estimated. This has the advantage that the covariance structure between the concentrations is also taken into account which is appropriate when the concentrations are known to be strongly inter-correlated with each other [32, 17]. As argued in [17] the multivariate approach can also lead to better predictions if the calibration data for one important concentration, say  $y_1$ , are imprecise. When this variable is highly correlated with some other constituents which are easier to measure precisely, then a joint calibration may give better understanding of the calibration data and better predictions for  $y_1$  than a separate univariate calibration for this analyte. Moreover, as we will show in the analysis of the Biscuit Dough data set in Section 5, the multivariate calibration can be very important to detect outlying samples which would not be discovered by separate regressions.

The statistical model under consideration says that there exists a linear relationship between a high-dimensional set of regressors (the spectra) and a small set of response variables (the concentrations). Let  $(x_1, \dots, x_p)$  denote the predictor variables, and  $(y_1, \dots, y_q)$  the response variables with  $q \geq 1$ , then we assume that each response variable follows its own regression model. Thus for each  $j = 1, \dots, q$

$$y_j = \beta_{0j} + \sum_{l=1}^p \beta_{lj}x_l + \varepsilon_j \quad (1)$$

where the error term  $\boldsymbol{\varepsilon} = (\varepsilon_1, \dots, \varepsilon_q)'$  satisfies  $E(\boldsymbol{\varepsilon}) = \mathbf{0}$  and  $Cov(\boldsymbol{\varepsilon}) = \Sigma_{\boldsymbol{\varepsilon}}$ . Note that we print column vectors in bold and that we indicate the transpose of a vector  $\mathbf{v}$  (or a matrix  $A$ ) with  $\mathbf{v}'$  (resp.  $A'$ ). We assume that we have independently drawn  $n$  observations  $(\mathbf{x}_i, \mathbf{y}_i)$ , with  $\mathbf{x}_i = (x_{i1}, \dots, x_{ip})'$  and  $\mathbf{y}_i = (y_{i1}, \dots, y_{iq})'$  from model (1) which can then alternatively

be written as

$$\mathbf{y}_i = \boldsymbol{\beta}_0 + \mathcal{B}'\mathbf{x}_i + \varepsilon_i \quad (2)$$

with  $\boldsymbol{\beta}_0 = (\beta_{01}, \dots, \beta_{0q})'$  the unknown  $q$ -dimensional intercept vector and  $\mathcal{B}$  the unknown  $p \times q$  slope matrix. In terms of the  $n \times p$  design matrix  $X_{n,p} = (\mathbf{x}_1, \dots, \mathbf{x}_n)'$  and the  $n \times q$  response matrix  $Y_{n,q} = (\mathbf{y}_1, \dots, \mathbf{y}_n)'$  we can equivalently say that

$$Y_{n,q} = \mathbf{1}_n \boldsymbol{\beta}_0' + X_{n,p} \mathcal{B}_{p,q} + \mathcal{E}_{n,q}. \quad (3)$$

Here,  $\mathbf{1}_n$  is the column vector with all its  $n$  components equal to 1, and  $\mathcal{E} = (\boldsymbol{\varepsilon}_1, \dots, \boldsymbol{\varepsilon}_n)'$ . Note that different rows of  $\mathcal{E}_{n,q}$  are uncorrelated, but that different variables for the same sample can be correlated. To clearly indicate the dimensions of the matrices involved, we will often mark their dimensions using subscripts as in (3). Remark that in the statistical literature, models (1) to (3) are known as the multivariate multiple regression model. The term ‘multiple’ indicates that we consider more than one  $x$ -regressor, whereas ‘multivariate’ indicates the presence of several response variables. We will also use this terminology throughout. If there is only one  $y$ -variable involved, we can write model (2) in the more familiar notation

$$y_i = \beta_0 + \boldsymbol{\beta}'\mathbf{x}_i + \varepsilon_i \quad (4)$$

with  $\boldsymbol{\beta} = (\beta_1, \dots, \beta_p)'$  the  $p$ -dimensional slope vector, and  $\beta_0$  the intercept. The errors  $\varepsilon_i$  are then assumed to satisfy  $E[\varepsilon_i] = 0$  and  $Var[\varepsilon_i] = \sigma^2$ . Unless otherwise needed, we will however not use this simplified notation but mostly write down the formulas for the general multivariate setting (2).

To estimate the regression parameters in models (2) and (4), a variety of methods have been developed, among which multiple linear regression (MLR), principal component regression (PCR) and partial least squares (PLS). The unbiased MLR least squares estimator has no unique minimum when  $p > n$ , and when  $p < n$ , it has a large variance if the regressors are highly correlated. Therefore biased estimators, such as PCR and PLS, offer a good solution because they first reduce the dimensionality of the design matrix. In this paper we will concentrate on the PCR approach, being a very appealing method because of its transparency and its easy computation.

Very shortly written, classical PCR (CPCR) starts by replacing the large number of explanatory variables  $x_j$  by a small number of loading vectors, which correspond to the first (classical) principal components of  $X_{n,p}$ . They are computed as the eigenvectors that

correspond to the largest eigenvalues of the covariance matrix of  $X$ . Then the response variables  $y_j$  are regressed onto these components using MLR.

Both the PCA stage and the regression stage in CPCR are however very sensitive to outlying observations, which are very likely to occur at real data. Note that in the following we will flag an observation as being an outlier if it does not follow the model that fits the majority of the data points. Its occurrence can e.g. be due to a measurement error, to a change in the experimental conditions, or to the fact that that sample belongs to another population than the one under study. So it does not necessarily imply that the sample is wrong or bad, although this terminology is sometimes abusively used.

When outliers are present in the data, the classical covariance matrix is inflated towards them and the first principal components will be directed to these outlying points. Consequently the PCR loading vectors will not adequately represent the directions which contain most of the information of  $X_{n,p}$ . A simple example which illustrates this effect is presented in Figure 1. It contains the Hertzsprung-Russell diagram of 47 stars, of which the logarithm of their light intensity and the logarithm of their surface temperature are measured [24]. The four outlying observations are giant stars, and clearly deviate from the main sequence stars. On this plot we have superimposed the classical tolerance ellipse, defined as the set of  $p$ -dimensional points  $\mathbf{x}$  whose *Mahalanobis distance*

$$\text{MD}(\mathbf{x}) = \sqrt{(\mathbf{x} - \bar{\mathbf{x}})' S_x^{-1} (\mathbf{x} - \bar{\mathbf{x}})} \quad (5)$$

equals  $\sqrt{\chi_{2,0.975}^2}$ , the square root of the 0.975 quantile of the  $\chi^2$  distribution with  $p = 2$  degrees of freedom. In (5) we use the classical estimates for the location and shape of the data, which are the mean  $\bar{\mathbf{x}}$  and the empirical covariance matrix  $S_x = \frac{1}{n-1} \sum_{i=1}^n (\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^t$  of the  $\mathbf{x}_i$ . On Figure 1 we see that this tolerance ellipse is highly rotated to accommodate the giant stars. The first principal component, which corresponds to the largest axis of this tolerance ellipse, consequently lies in the direction of these outliers. On the other hand, a robust PCA method yields a tolerance ellipse which captures the covariance structure of the majority of the data points. On Figure 1 this robust tolerance ellipse is obtained by applying the highly robust Minimum Covariance Determinant (MCD) estimator of location and scatter [22] to the data, yielding  $\hat{\boldsymbol{\mu}}_{\text{MCD}}$  and  $\hat{\boldsymbol{\Sigma}}_{\text{MCD}}$ , and by plotting the points  $\mathbf{x}$  whose *robust distance*

$$D(\mathbf{x}) = D(\mathbf{x}, \hat{\boldsymbol{\mu}}_{\text{MCD}}, \hat{\boldsymbol{\Sigma}}_{\text{MCD}}) = \sqrt{(\mathbf{x} - \hat{\boldsymbol{\mu}}_{\text{MCD}})' \hat{\boldsymbol{\Sigma}}_{\text{MCD}}^{-1} (\mathbf{x} - \hat{\boldsymbol{\mu}}_{\text{MCD}})} \quad (6)$$

is equal to  $\sqrt{\chi_{2,0.975}^2}$ . More details on the MCD estimator are presented in Section 3.2.

[Figure 1 about here]

Besides the PCA stage, it is also well known that the least squares MLR method is highly sensitive to outliers. Both outliers in the space of the response variables, and in the space of the explanatory variables can unduly influence the parameter estimates. See e.g. [24] for many illustrations on real data.

In this paper we propose a robust PCR method, by combining a robust PCA algorithm for high-dimensional data [11] with robust regression [22, 25]. First we will reconsider the CPCR approach in Section 2. In Section 3 we introduce our robust RPCR method. A robust RMSECV value and a robust  $R^2$ -measure are presented in Section 4 and used to select the number of principal components. Also the prediction error will be estimated in a robust way. Section 5 introduces several diagnostic plots which are helpful to visualize and classify the outliers, and illustrates them on a real example. The results of a simulation study can be found in Section 6. Finally a conclusion is formulated in Section 7.

## 2 Classical principal component regression

Classical PCR (CPCR) starts by centering the data through the mean  $\bar{\mathbf{x}}$  of the  $x$ -variables and the mean  $\bar{y}$  of the  $y$ -variables. Let the centered observations be denoted by

$$\tilde{\mathbf{x}}_i = \mathbf{x}_i - \bar{\mathbf{x}} \tag{7}$$

$$\tilde{y}_i = y_i - \bar{y}. \tag{8}$$

Then, in order to cope with the multicollinearity in the  $x$ -variables, the first  $k$  principal components of  $X_{n,p}$  are computed. It is well known that these loading vectors  $\tilde{P}_{p,k} = (\mathbf{p}_1, \dots, \mathbf{p}_k)'$  are the  $k$  eigenvectors that correspond to the  $k$  largest eigenvalues of the empirical covariance matrix  $S_x = \frac{1}{n-1} \tilde{X}'_{p,n} \tilde{X}_{n,p}$ . Hence they are uncorrelated, orthogonal and they determine a new coordinate system in the  $k$ -dimensional subspace that they span. Next, the  $k$ -dimensional scores of each data point  $\tilde{\mathbf{t}}_i$  are computed as the coordinates of the projections of  $\tilde{\mathbf{x}}_i$  onto this subspace, or equivalently

$$\tilde{\mathbf{t}}_i = \tilde{P}'_{k,p} \tilde{\mathbf{x}}_i. \tag{9}$$

In the final step, the centered response variables  $\tilde{\mathbf{y}}_i$  are regressed onto  $\tilde{\mathbf{t}}_i$  using MLR. We thus fit the linear model

$$\tilde{\mathbf{y}}_i = \mathcal{A}'\tilde{\mathbf{t}}_i + \tilde{\boldsymbol{\varepsilon}}_i$$

which does not contain an intercept term because the data are mean-centered. We obtain parameter estimates

$$\hat{\mathcal{A}}_{k,q} = (T'T)_{k,k}^{-1}T'_{k,q}\tilde{Y}_{q,q}$$

and fitted values

$$\begin{aligned}\hat{\mathbf{y}}_i &= \hat{\mathcal{A}}'_{q,k}\tilde{\mathbf{t}}_i + \bar{\mathbf{y}} \\ &= \hat{\mathcal{A}}'_{q,k}\tilde{P}'_{k,p}(\mathbf{x}_i - \bar{\mathbf{x}}) + \bar{\mathbf{y}}\end{aligned}$$

by (7), (8) and (9). The unknown regression parameters in model (2) are then estimated as

$$\begin{aligned}\hat{\mathcal{B}}_{p,q} &= \tilde{P}_{p,k}\hat{\mathcal{A}}_{k,q} \\ \hat{\boldsymbol{\beta}}_0 &= \bar{\mathbf{y}} - \hat{\mathcal{B}}'_{q,p}\bar{\mathbf{x}}.\end{aligned}$$

Finally the covariance matrix of the errors can be estimated as the empirical covariance matrix of the residuals

$$\begin{aligned}S_{\boldsymbol{\varepsilon}} &= \frac{1}{n-1} \sum_{i=1}^n \mathbf{r}_i \mathbf{r}'_i \\ &= \frac{1}{n-1} \sum_{i=1}^n (\mathbf{y}_i - \hat{\mathbf{y}}_i)(\mathbf{y}_i - \hat{\mathbf{y}}_i)' \\ &= S_y - \hat{\mathcal{A}}' S_t \hat{\mathcal{A}}\end{aligned}\tag{10}$$

with  $S_y$  and  $S_t$  being the empirical covariance matrices of the  $y$ - and the  $t$ -variables. Note that equality (10) follows from the fact that the fitted MLR values are orthogonal to the MLR residuals [14].

### 3 Robust principal component regression

#### 3.1 Robust PCA in low dimensions

Analogously to CPCr we first apply a robust PCA method to the  $x$ -variables. When the number of regressors  $p$  is smaller than the data size  $n$ , we can e.g. use the Minimum Covariance Determinant (MCD) estimator [22, 6], as we have illustrated in Section 1. This

location and covariance estimator is very popular because of its high resistance towards outliers and because a fast algorithm has recently been developed for its computation [26]. To define the MCD estimator we consider subsets of size  $h$  out of the whole data set (of size  $n$ ). The number  $h$  determines the robustness of the estimator and should be at least  $\lceil (n + p + 1)/2 \rceil$ . The MCD estimator then seeks for that  $h$ -subset whose classical covariance matrix has minimal determinant. The MCD location estimate is given by the mean  $\bar{\mathbf{x}}_h$  of that optimal  $h$ -subset, and the MCD scatter estimator by its covariance matrix  $\hat{\Sigma}_h$ , multiplied by a consistency factor. In [26] this consistency factor is defined as the median of the squared robust distances

$$(\mathbf{x}_i - \bar{\mathbf{x}}_h)' \hat{\Sigma}_h^{-1} (\mathbf{x}_i - \bar{\mathbf{x}}_h) \quad (11)$$

divided by  $\chi_{p,0.5}^2$  which is the median of the  $\chi_2$  distribution with  $p$  degrees of freedom. We have slightly modified this consistency factor into the  $h$ -th order statistic of the squared robust distances, divided by  $\chi_{p,h/n}^2$ .

Based on the raw MCD estimate, a reweighting step can be added which increases the finite-sample efficiency considerably [26]. Briefly written, each data point receives a weight 1 if it belongs to the robust tolerance ellipse (6) and weight zero otherwise. The reweighted MCD estimator then equals the classical mean and covariance matrix of the data points with weight 1. Further on, we make no distinction anymore between the raw and the reweighted MCD estimator, as we assume that we use always the reweighted one.

The first  $k$  eigenvectors of the MCD-estimator, sorted in descending order of the eigenvalues, then yield robust loadings. Note that this approach is already implemented in S-PLUS 6.0, Insightful Corporation, Washington.

It is intuitively clear that the MCD estimator can resist  $n - h$  outliers. More formally it is said that the MCD estimator has a breakdown value of  $(n - h + 1)/n$  which means that we need at least  $n - h + 1$  outliers to make the estimates worthless. For a scatter matrix this means that its largest eigenvalue becomes arbitrarily large, or that its smallest eigenvalue becomes arbitrarily close to zero [24].

### 3.2 Robust PCA in high dimensions

For high-dimensional regressors ( $p > n$ ) we can not use the MCD anymore because the determinant of a covariance matrix of  $h < p$  observations will always be zero and thus can not be minimized. Therefore we apply the ROBPCA method [11] on the design matrix  $X_{n,p}$ .

ROBPCA is a robust PCA method which combines projection pursuit ideas with MCD covariance estimation in lower dimensions. Here, we will briefly outline the most important stages of the algorithm, the details of which can be found in [11].

First, the  $x$ -data are preprocessed by reducing their data space to the affine subspace spanned by the  $n$  observations. This can be easily performed using a singular value decomposition of  $X_{n,p}$ . As a result, the data are represented using at most  $n - 1 = \text{rang}(\tilde{X}_{n,p})$  variables without loss of information.

In the second step of the ROBPCA algorithm, a measure of outlyingness is computed for each data point. This is obtained by projecting the high-dimensional data points on many univariate directions. On every direction a robust center and scale of the projected data points is computed, and for every data point its standardized distance to that center is measured. Finally for each data point its largest distance over all the directions is considered. The  $h$  data points with smallest outlyingness are then retained, and some updates of this  $h$ -subset are performed. From the covariance matrix of the final  $h$ -subset, the number of principal components to retain,  $k$ , is selected.

The last stage of ROBPCA consist of projecting the data points onto this  $k$ -dimensional subspace and of computing their center and shape by means of the reweighted MCD estimator. The eigenvectors of this scatter matrix then determine the robust principal components, and the MCD location estimate serves as a robust center.

Summarizing, the ROBPCA method applied to  $X_{n,p}$  yields robust principal components which can be collected in a loading matrix  $P_{p,k}$  with orthogonal columns, and a robust center  $\hat{\boldsymbol{\mu}}_x$ . From these, robust scores can be derived as

$$\mathbf{t}_i = P'_{k,p}(\mathbf{x}_i - \hat{\boldsymbol{\mu}}_x).$$

Note that the MCD scatter matrix defines a metric in the PCA subspace. Let  $L$  denote the diagonal matrix which contains the eigenvalues  $l_j$  of the MCD scatter matrix, sorted from largest to smallest. Using (6) we then define the *score distance* of a  $p$ -dimensional point  $\mathbf{x}$

with respect to  $\hat{\boldsymbol{\mu}}_x, P$  and  $L$  as

$$\begin{aligned}
\text{SD}(\mathbf{x}) &= \text{SD}(\mathbf{x}, \hat{\boldsymbol{\mu}}_x, P, L) = \text{D}(\mathbf{t} = P'(\mathbf{x} - \hat{\boldsymbol{\mu}}_x), \mathbf{0}, L) \\
&= \left( \sqrt{(\mathbf{x} - \hat{\boldsymbol{\mu}}_x)' P_{p,k} L_{k,k}^{-1} P'_{k,p} (\mathbf{x} - \hat{\boldsymbol{\mu}}_x)} \right) \\
&= \sqrt{\mathbf{t}' L^{-1} \mathbf{t}} \\
&= \sqrt{\sum_{j=1}^k \frac{t_j^2}{l_j}}.
\end{aligned} \tag{12}$$

The ROBPCA method is orthogonally and location equivariant. When we apply a translation and/or an orthogonal transformation (rotation, reflection) to the observations  $\mathbf{x}_i$  the robust center is also shifted and the loadings are rotated accordingly. Hence the scores do not change under this type of transformations. Let  $A_{p,p}$  define the orthogonal transformation, thus  $A$  is of full rank and  $A' = A^{-1}$ , and  $\hat{\boldsymbol{\mu}}_x$  and  $P_{p,k}$  the ROBPCA center and loading matrix for the original  $\mathbf{x}_i$ . Then the ROBPCA center and loadings for the transformed data  $A\mathbf{x}_i + \mathbf{v}$  are equal to  $A\hat{\boldsymbol{\mu}}_x + \mathbf{v}$  and  $AP$ . Consequently the scores remain the same under these transformations:

$$\mathbf{t}_i(A\mathbf{x}_i + \mathbf{v}) = P' A' (A\mathbf{x}_i + \mathbf{v} - (A\hat{\boldsymbol{\mu}}_x + \mathbf{v})) = P'(\mathbf{x}_i - \hat{\boldsymbol{\mu}}_x) = \mathbf{t}_i(\mathbf{x}_i).$$

Although these properties seem very natural for a PCA method, they are not shared by some other robust PCA estimators such as the resampling by half-means and the smallest half-volume methods of Egan and Morgan [8].

### 3.3 Robust regression

In the second stage of our RPCR method we regress  $\mathbf{y}_i$  on  $\mathbf{t}_i$  using a robust regression method. Note that here we fit a regression model with intercept:

$$\mathbf{y}_i = \boldsymbol{\alpha}_0 + \mathcal{A}' \mathbf{t}_i + \check{\boldsymbol{\varepsilon}}_i \tag{13}$$

with  $\text{Cov}(\check{\boldsymbol{\varepsilon}}) = \Sigma_{\check{\boldsymbol{\varepsilon}}}$ . If the PCR model (2) only contains one response variable ( $q = 1$ ), regression model (13) simplifies to

$$y_i = \alpha_0 + \boldsymbol{\alpha}' \mathbf{t}_i + \check{\varepsilon}_i \tag{14}$$

with  $\check{\sigma}_{\varepsilon}$  the scale of the errors. To estimate the parameters in (14) we propose to use the reweighted LTS estimator [22] for which a fast algorithm is available [27]. The raw LTS

estimator minimizes the sum of the  $h$  smallest squared residuals, i.e.

$$(\hat{\boldsymbol{\alpha}}, \hat{\alpha}_0)_{\text{LTS}} = \underset{\boldsymbol{\alpha}, \alpha_0}{\operatorname{argmin}} \sum_{i=1}^h (r^2(\boldsymbol{\alpha}, \alpha_0))_{i:n}$$

where  $r_{1:n}^2 \leq r_{2:n}^2 \leq \dots \leq r_{n:n}^2$  denote the ordered squared residuals. An initial estimate of the error dispersion is given by

$$\hat{\sigma}_0 = c_h \sqrt{\frac{1}{h} \sum_{i=1}^h (r^2(\hat{\boldsymbol{\alpha}}, \hat{\alpha}_0)_{\text{LTS}})_{i:n}}$$

with  $c_h$  a consistency factor for normally distributed errors [23]. The reweighted LTS estimator then corresponds to the least squares estimator applied to the observations whose absolute standardized residual is not too large. More precisely we set

$$w_i = 0 \text{ if } |r_i(\hat{\boldsymbol{\alpha}}, \hat{\alpha}_0)_{\text{LTS}}/\hat{\sigma}_0| > 2.5 \quad (15)$$

and  $w_i = 1$  otherwise and obtain the final estimates  $(\hat{\boldsymbol{\alpha}}, \hat{\alpha}_0)$  as the vector which minimizes  $\sum_{i=1}^n w_i (y_i - \boldsymbol{\alpha}'\mathbf{t}_i - \alpha_0)^2$ . The final squared scale estimate is then given by

$$\hat{\sigma}_{\xi}^2 = \left( \sum_{i=1}^n w_i (y_i - \hat{\boldsymbol{\alpha}}'\mathbf{t}_i - \hat{\alpha}_0)^2 \right) / \left( \sum_{i=1}^n w_i - k \right).$$

If  $q > 1$  we use the MCD-regression estimator [25]. It starts by computing the reweighted MCD estimator on the  $(\mathbf{t}_i, \mathbf{y}_i)$  jointly, leading to a  $(k+q)$ -dimensional location estimate  $\hat{\boldsymbol{\mu}} = (\hat{\boldsymbol{\mu}}_t, \hat{\boldsymbol{\mu}}_y)'$  and a scatter estimate  $\hat{\Sigma}_{k+q, k+q}$  which can be split into a scatter estimate of the  $t$ -variables, the  $y$ -variables and of the cross-covariance between the  $t$ 's and  $y$ 's:

$$\hat{\Sigma}_{\text{MCD}} = \begin{pmatrix} \hat{\Sigma}_t & \hat{\Sigma}_{ty} \\ \hat{\Sigma}_{yt} & \hat{\Sigma}_y \end{pmatrix}.$$

In analogy with the MLR estimates which are based on the empirical covariance matrix of the joint  $(\mathbf{t}_i, \mathbf{y}_i)$  variables [14], robust parameter estimates are then obtained as

$$\begin{aligned} \hat{\mathbf{A}}_{k,q} &= \hat{\Sigma}_t^{-1} \hat{\Sigma}_{ty} \\ \hat{\boldsymbol{\alpha}}_0 &= \hat{\boldsymbol{\mu}}_y - \hat{\mathbf{A}}' \hat{\boldsymbol{\mu}}_t \\ \hat{\Sigma}_{\xi} &= \hat{\Sigma}_y - \hat{\mathbf{A}}' \hat{\Sigma}_t \hat{\mathbf{A}}. \end{aligned} \quad (16)$$

Note indeed the correspondence of (16) with (10). In [25] it is shown that this regression estimator inherits the breakdown value of the MCD estimator, and that its efficiency can

also be increased by performing a reweighting step. In line with the author's conclusions we adopt their 'location-regression' reweighting scheme. To apply this, each data point receives a zero weight  $w_i$  if its initial *residual distance* is unusually large, i.e.

$$w_i = 0 \text{ if } \text{RD}_i > \sqrt{\chi_{q,0.99}^2} \quad (17)$$

with

$$\mathbf{r}_i = \mathbf{y}_i - \hat{\mathcal{A}}' \mathbf{t}_i - \hat{\boldsymbol{\alpha}}_0 \quad (18)$$

$$\text{RD}_i = D(\mathbf{r}_i, \mathbf{0}, \hat{\Sigma}_{\boldsymbol{\varepsilon}}) = \sqrt{\mathbf{r}_i' \hat{\Sigma}_{\boldsymbol{\varepsilon}}^{-1} \mathbf{r}_i}. \quad (19)$$

All other observations obtain a weight  $w_i = 1$ . The reweighted MCD-regression parameters then correspond to the MLR estimates based on those observations with weight 1. The final residual distances are obtained by filling in the reweighted estimates for  $\mathcal{A}$  in (16), (18) and (19). We do not introduce a different notation for the final estimates and residual distances, but assume that the reweighting step is indeed applied.

The fitted values now satisfy

$$\hat{\mathbf{y}}_i = \hat{\mathcal{A}}'_{q,k} \mathbf{t}_i + \hat{\boldsymbol{\alpha}} \quad (20)$$

$$= \hat{\mathcal{A}}'_{q,k} P'_{k,p} (\mathbf{x}_i - \hat{\boldsymbol{\mu}}_x) + \hat{\boldsymbol{\alpha}} \quad (21)$$

from which the regression parameters in model (2) are derived:

$$\hat{\mathcal{B}}_{p,q} = P_{p,k} \hat{\mathcal{A}}_{k,q} \quad (22)$$

$$\hat{\boldsymbol{\beta}}_0 = \hat{\boldsymbol{\alpha}} - \hat{\mathcal{B}}_{p,q} \hat{\boldsymbol{\mu}}_x. \quad (23)$$

Finally we set

$$\hat{\Sigma}_{\boldsymbol{\varepsilon}} = \hat{\Sigma}_{\boldsymbol{\varepsilon}}. \quad (24)$$

Note that equations (20) through (24) can also be rewritten for a univariate response  $q = 1$ .

Remark 1:

As for the MCD estimator, the robustness of the RPCR algorithm depends on the value of  $h$  which is chosen in the ROBPCA algorithm and in the LTS and MCD-regression. Although it is not really necessary, we prefer to use the same value in both steps. In our Matlab

implementation the user can either choose  $h$  from the start, or choose a value of  $\alpha$ , with  $0 \leq \alpha \leq 0.5$ , such that  $1 - \alpha$  corresponds with the percentage of outliers that the algorithm should be able to resist. We then set  $h$  as the maximum of  $h_1 = \lceil \alpha n \rceil$  and  $h_2 = \lceil \frac{n+k+q+1}{2} \rceil$  where  $h_2$  is the required minimal value for the MCD-regression estimator to have a positive breakdown value. When a large proportion of contamination is presumed,  $\alpha$  should be chosen close to 0.5. Otherwise an intermediate value for  $\alpha$ , such as 0.75, is recommended because it increases the finite-sample efficiency [26, 5]. Our default choice is therefore also  $\alpha = 0.75$ .

Remark 2:

Both the reweighted MCD estimator, explained in Section 3.1, and the robust regression estimators (LTS, MCD-regression) use weights with attain only two values 0 or 1. This so-called hard rejection rule is very popular [26, 27, 16, 1, 5]. A more smoothed weight function could also be used. In [15] it is shown that the breakdown value of the reweighted estimator (and thus the resistance to outliers) remains the same when a redescending weight function is selected. A popular weight function is the biweight function

$$w_i = (1 - \tilde{r}_i^2)^2 I(|\tilde{r}_i| \leq c).$$

Here  $\tilde{r}_i$  should be read as the root of the robust distance (11) when reweighting the MCD estimator, as the standardized residual  $r_i/\hat{\sigma}_0$  for the LTS regression, and as the residual distance (19) for the MCD-regression estimator. The constant  $c$  is a tuning parameter whose value can be chosen to attain a certain efficiency at the uncontaminated model. In [16] it is however mentioned that the performance of reweighted  $S$ -estimators of location and covariance [29] did not benefit from using this biweight function, hence the authors recommended to use hard rejection.

In the regression step, one could also use the raw LTS estimator and the raw MCD-regression estimator as initial estimators for one-step Generalized M-estimators [30, 4, 1] which are also defined through a weight function. In [1] it is however shown that outliers can have a much higher influence on the one-step M-estimators than even on the raw estimator. In this paper we thus only considered the hard rejection rules, although other weighting schemes could be used without increasing the computational load of the procedure.

Remark 3:

The proposed RPCR method has several appealing equivariance properties. Following the definitions in [25] it is  $x$ -translation equivariant,  $x$ -orthogonally equivariant and  $y$ -affine equivariant. The first two properties tells us that when we apply a translation to the  $x$ -variables and/or an orthogonal transformation (rotation, reflection), the regression coefficients are transformed accordingly. More specifically, let  $\mathbf{v}$  be any  $p$ -dimensional vector,  $A_{p,p}$  any orthogonal matrix, and  $(\hat{\boldsymbol{\beta}}_0, \hat{\mathbf{B}})$  the RPCR estimates of the original  $(\mathbf{x}_i, \mathbf{y}_i)$  data, then

$$\begin{aligned}\hat{\mathbf{B}}(A\mathbf{x}_i + \mathbf{v}, \mathbf{y}_i) &= A\hat{\mathbf{B}} \\ \hat{\boldsymbol{\beta}}_0(A\mathbf{x}_i + \mathbf{v}, \mathbf{y}_i) &= \hat{\boldsymbol{\beta}}_0 - \hat{\mathbf{B}}^t A^t \mathbf{v}.\end{aligned}$$

These properties follow in a straightforward way from the orthogonal equivariance of the ROBPCA method. Robust PCR methods which are based on non equivariant PCA estimators, such as proposed in [20], therefore are also not  $x$ -equivariant.

The  $y$ -affine equivariance is inherited from the LTS and MCD-regression method and allows linear transformations of the  $y$ -variables, i.e.

$$\begin{aligned}\hat{\mathbf{B}}(\mathbf{x}_i, C\mathbf{y}_i + \mathbf{d}) &= \hat{\mathbf{B}}C^t \\ \hat{\boldsymbol{\beta}}_0(\mathbf{x}_i, C\mathbf{y}_i + \mathbf{d}) &= C\hat{\boldsymbol{\beta}}_0 + \mathbf{d}\end{aligned}$$

for any non-singular  $(q \times q)$  matrix  $C$  and any  $q$ -dimensional vector  $\mathbf{d}$ .

## 4 Selecting the number of scores

An important issue in PCR is the selection of the number of principal components, for which several methods have been proposed. A very popular approach consists of minimizing the root mean squared error of cross-validation RMSECV $_k$  criterion which, for one response variable ( $q = 1$ ), equals

$$\text{RMSECV}_k = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_{-i,k})^2} \quad (25)$$

with  $\hat{y}_{-i,k}$  the predicted value for observation  $i$ , left out from the data set to perform the PCR method with  $k$  principal components. Also  $m$ -fold cross-validation procedures are common, especially when the data size becomes too large for a leave-one-sample-out strategy [32, 10, 2]. The goal of the RMSECV $_k$  statistic is twofold. It yields an estimate of the root mean squared

prediction error  $E(y - \hat{y})^2$  when  $k$  components are used in the model, whereas the curve of  $\text{RMSECV}_k$  for  $k = 1, \dots, k_{\max}$  is a popular graphical tool to choose the optimal number of components.

This  $\text{RMSECV}_k$  statistic is however not suited at contaminated data sets because we do not want to include the prediction error of the outliers in (25). A straightforward robustification of (25) thus consists of ignoring the outliers, i.e. the observations that receive a final zero weight. For a fixed value of  $k$ , this yields a robust root mean squared error of prediction

$$\text{R-RMSEP}_k = \sqrt{\frac{1}{\sum_{i=1}^n w_{-i,k}} \sum_{i=1}^n w_{-i,k} (y_i - \hat{y}_{-i,k})^2} \quad (26)$$

where the predicted values  $\hat{y}_{-i,k}$  are now based on the robust fit  $\hat{\beta}_{-i,k}$  obtained by leaving out observation  $i$  from the RPCR analysis with  $k$  components. The weight  $w_{-i,k}$  is obtained as in (15), but based on the reweighted parameter estimates  $\hat{\beta}_{-i,k}$  and  $\hat{\sigma}_{-i,k}$ . Since the weights  $w_{-i,k}$  depend on  $k$ , there is however no guarantee that the  $\text{R-RMSEP}_{k_1}$  value (26) for a model with  $k_1$  components will be based on the same observations that determine the  $\text{R-RMSEP}_{k_2}$  value for a model with  $k_2$  components. This would make the two  $\text{R-RMSEP}$  values incomparable. Therefore to select the number of components we propose to plot the robust  $\text{R-RMSECV}_k$  value defined for  $k = 1, \dots, k_{\max}$ :

$$\text{R-RMSECV}_k = \sqrt{\frac{1}{n_w} \sum_{i=1}^n w_{-i} (y_i - \hat{y}_{-i,k})^2} \quad (27)$$

with

$$w_{-i} = \min_{k=1, \dots, k_{\max}} w_{-i,k} \quad \text{and} \quad n_w = \sum_{i=1}^n w_{-i}.$$

With this definition we eliminate an observation from the  $\text{R-RMSECV}_k$  statistic as soon as after cross-validation it shows up as an outlier in any of the models with  $k = 1$  to  $k = k_{\max}$  components.

In the multivariate setting we compute  $\text{R-RMSEP}_k$  and  $\text{R-RMSECV}_k$  for each component separately, but now the weights  $w_{-i,k}$  are computed from (17) to (19) with parameter estimates  $\mathcal{A}_{-i,k}$  and  $\hat{\Sigma}_{-i,k}$ . As an overall measure, we take the root of the average of the squared  $\text{R-RMSECV}_k$  values for each concentration. With  $\hat{y}_{-ij,k}$  the  $j$ th component of  $\hat{\mathbf{y}}_{-i,k}$

we then obtain

$$\text{R-RMSECV}_k = \sqrt{\frac{1}{n_w q} \sum_{j=1}^q \sum_{i=1}^n w_{-i} (y_{ij} - \hat{y}_{-ij,k})^2} \quad (28)$$

$$= \sqrt{\frac{1}{n_w q} \sum_{i=1}^n w_{-i} \|\mathbf{y}_i - \hat{\mathbf{y}}_{-i,k}\|^2}$$

$$= \sqrt{\frac{1}{n_w q} \text{trace} \left( \sum_{i=1}^n w_{-i} \mathbf{r}_{-i,k} \mathbf{r}'_{-i,k} \right)}. \quad (29)$$

From the  $\text{R-RMSECV}_k$  curve we select the  $k_{\text{opt}}$  value for which  $\text{R-RMSECV}_k$  is minimal or attains a plateau. Analogously to (26) we then estimate the overall prediction error with

$$\text{R-RMSEP}_{k_{\text{opt}}} = \sqrt{\frac{1}{q \sum_{i=1}^n w_{-i,k}} \sum_{j=1}^q \sum_{i=1}^n w_{-i,k} (y_{ij} - \hat{y}_{-ij,k_{\text{opt}}})^2}.$$

These  $\text{R-RMSEP}_k$  and  $\text{R-RMSECV}_k$  values are rather time consuming, because for every choice of  $k$  they require the whole RPCR procedure to be performed  $n$  times. On a Windows PC (AMD Athlon, 800 Mhz, 256 MB RAM) it takes e.g. only 10 seconds to compute the regression parameters for  $n = 40, p = 600$  and  $k = 2$ , but the  $\text{R-RMSEP}_2$  statistic needs approximately 6 minutes, whereas 21 minutes are required to draw the  $\text{R-RMSECV}_k$  curve for  $k = 1$  to  $k = 4$ . Therefore as a faster alternative we also propose to maximize a robust  $R^2$ -value which is computed without cross-validation. As a byproduct of a robust regression method, several robust  $R^2$ -values have been proposed for the univariate case  $q = 1$  [23, 31]. For the raw LTS estimator this  $R^2$ -value is defined as

$$R^2 = 1 - \frac{\sum_{i=1}^h (r^2)_{i:n}}{\sum_{i=1}^h ((y - \hat{\mu})^2)_{i:n}} \quad (30)$$

with  $\hat{\mu}$  the univariate location LTS estimator of the  $y_i$ . We see that it contains the ratio of a robust variance estimate of the residuals to a robust variance estimate of the response variable, and thus (30) expresses in a robust way the proportion of the  $y$ -variability which is explained by the fitted model.

Because we are using the reweighted LTS estimator, we define the robust  $R^2$ -measure as the classical  $R^2$ -value based on the data points with nonzero weight (15) in any of the models with  $k = 1$  to  $k = k_{\text{max}}$  components. Let  $r_{i,k}$  denote the  $i$ th residual when  $k$  scores are used in RPCR,  $w_{i,k}$  its final LTS weight and  $w_i = \min_{k=1, \dots, k_{\text{max}}} w_{i,k}$ . The robust  $R_k^2$  is then

defined as

$$R_k^2 = 1 - \frac{\sum_{i=1}^n w_i r_{i,k}^2}{\sum_{i=1}^n w_i (y_i - \bar{y}_w)^2} \quad (31)$$

with  $\bar{y}_w = (\sum_i w_i y_i) / (\sum_i w_i)$ .

For the multivariate case  $q > 1$  we generalize (31) to

$$R_k^2 = 1 - \frac{\sum_{j=1}^q \sum_{i=1}^n w_i r_{ij,k}^2}{\sum_{j=1}^q \sum_{i=1}^n w_i (y_{ij} - \bar{y}_j)^2}. \quad (32)$$

with  $\bar{y}_j = (\sum_i w_i y_{ij}) / (\sum_i w_i)$ . This  $R_k^2$  curve becomes flat when the addition of a component does not reduce the residual variance significantly. The number of optimal components  $k_{\text{opt}}$  can then again be selected as the smallest  $k$  such that  $R_{k_{\text{opt}}}^2$  attains a certain value, e.g. 80%, or until the curve becomes nearly flat. We will illustrate this in the next session.

## 5 Diagnostic plots

To visualize the outcomes of a PCR analysis on a real data set, we can use several graphical displays, such as loading and scores plots and residual plots. In this section we will also introduce several two- and three-dimensional diagnostic plots which are helpful to visualize and classify the observed data points into regular observations and outliers. To illustrate these concepts, we analyze the Biscuit Dough data set [19, 18] preprocessed as in [13]. This data set consists of 40 NIR spectra of biscuit dough with measurements every 2 nanometers, from 1200nm up to 2400nm. The responses are the percentages of 4 constituents in the biscuit dough:  $y_1 = \text{fat}$ ,  $y_2 = \text{flour}$ ,  $y_3 = \text{sucrose}$  and  $y_4 = \text{water}$ . Because there is a significant correlation among the regressors, we decided to perform a multivariate regression. Figure 2(a) illustrates the positive correlation between flour and water, whereas Figure 2(b) shows the strong negative correlation between flour and sucrose.

[Figure 2 about here]

The robust R-RMSECV $_k$  curve (28) and the  $R_k^2$  curve (32) are plotted in Figure 3. Based on this figure, we decided to retain 2 components. We then attain a  $R^2$ -value of 92%, whereas the estimated overall prediction error RMSEP $_2$  equals 0.90. Here we have set  $k_{\text{max}} = 4$  since otherwise we would include too many parameters in the model. With  $q = 4$  and  $k = 4$  we have already  $kq = 16$  parameters for the slope matrix  $\mathcal{A}$ ,  $q = 4$  parameters for the

intercept and  $q(q-1)/2 = 6$  for the error covariance matrix  $\Sigma_\epsilon$ , which gives a total of 26 parameters. To avoid overfitting we require that the number of parameters (26) be smaller than  $h = [0.75n] = 30$ .

[Figure 3 about here]

The two loadings from both the CPCr and the RPCr method are depicted in Figure 4(a) and (b). The estimated regression coefficients are shown in Figure 5(a) to (d). On these plots we can clearly see the differences between the classical and the robust analysis. For example, on Figure 4(b) where the second loading vector is plotted, we notice (between wavelengths 1390 and 1440) a large discrepancy in the C-H bend. On the other hand, the classical and robust calibration vectors of water, shown in Figure 5(d), do not differ too much.

[Figures 4 and 5 about here]

Next, we present the diagnostic plots which give us information about how well each data point fits the model. Figure 6(a) contains the robust PCA diagnostic plot as introduced in [11]. On the horizontal axis we mark for each data point its score distance  $SD_i = SD(\mathbf{x}_i)$ , defined in (12). Observations whose score distance exceeds the cut-off value  $\sqrt{\chi_{k,0.975}^2}$  are outlying within the PCA subspace. The orthogonal distance  $OD_i$

$$OD_i = \|\mathbf{x}_i - \hat{\boldsymbol{\mu}}_x - P_{p,k}T'_{k,i}\| \quad (33)$$

is shown on the  $y$ -axis and measures how far a data point lies from the PCA subspace. The overview in Table 1 allows us to distinguish four types of observations.

[Table 1 about here]

On the robust PCA diagnostic plot for the Biscuit Dough data set in Figure 6(a), we see that there are no leverage points, but there are a few orthogonal outliers, namely observations 23, 7 and 20. We can also make a diagnostic plot based on the classical PCA analysis. The definitions for the score distance and the orthogonal distance are then analogous to (12) and (33), if we replace the loadings and scores with their classical estimates. Figure 6(b) shows this CPCA diagnostic plot. We see that CPCA has accommodated the orthogonal outliers and has turned observation 23 into a good leverage point.

[Figure 6 about here]

Next we consider the regression diagnostic plot, which is obtained as a byproduct of the multivariate regression method. Analogously to the regression diagnostic plots in [28, 25] we plot the residual distance (19) versus the score distance (12). We can then classify the data points according to Table 2. Observations whose residual distance exceeds  $\sqrt{\chi_{q,0.975}^2}$  are called vertical outliers or bad leverage points, depending on their score distance.

[Table 2 about here]

For the Biscuit Dough data set, we obtain very different regression plots for the robust (Figure 6(c)) and the classical PCR (Figure 6(d)) approach. RPCR shows that observation 21 has an extremely high residual distance. Other vertical outliers are 23, 7, 20, 24, and 22, whereas there are a few boundary cases. CPCr on the other hand tries to make all the residuals as small as possible, resulting in exposing only one clear vertical outlier 21 (number 7 being a boundary case as well). As in Figure 6(b) also observation 23 is distinguished, but rather due to its large score distance than due to its residual distance, so it seems to fit the regression model rather well.

The three-dimensional diagnostic plots in Figure 7 combine the PCA and the regression diagnostic plots by plotting for each observation the triple  $(SD_i, OD_i, RD_i)$ . It is particularly interesting to create such three-dimensional plots with interactive software packages (such as Matlab or S-PLUS) which allow you to rotate and spin the whole figure. Note that to be able to better distinguish the data points in Figure 7(b), we did not adjust the scale of the axes to those of the robust plot, as we did in Figures 6(b) and (d).

[Figure 7 about here]

To illustrate the usefulness of the single multivariate regression approach compared to doing several univariate regressions for each response variable separately, we have created some additional residual plots as well. First, we wondered whether the large outlyingness of observation 21 in Figure 6(c) was due to a large residual with respect to one of the concentrations. Therefore, we computed for each data point  $i = 1, \dots, 40$  and for each response variable  $y_j$  ( $j = 1, \dots, 4$ ) the univariate standardized residual

$$sr_{i,j} = \frac{y_{i,j} - \hat{y}_{i,j}}{\hat{\sigma}_j}$$

with  $\hat{\sigma}_j^2$  the  $j$ -th diagonal element of  $\hat{\Sigma}_\epsilon$ . The resulting univariate residual plots can be seen in Figure 8(a) to (d). Since the standardized residuals can be positive or negative, we indicate both a positive and a negative cut-off value at  $\pm\sqrt{\chi_{1,0.975}^2} = \pm 2.24$ .

We see that observation 21 never shows up as a large outlier. It is only by using the full covariance structure of the residuals in (19) that this extreme data point is found. Note that in this particular example the same happens at the classical analysis. The residual plots in Figure 9(a) to (d) also do not reveal observation 21 as an atypical sample.

[Figures 8 and 9 about here]

Next we ignored the multivariate response structure completely and performed four univariate calibrations. First we created the R-RMSECV curves for each constituent separately, from which we decided to retain 6 components for fat, 2 for flour, and 3 for sucrose and water. The resulting residual plots are shown in Figure 10(a) to (d). Again, there is no clear indication that sample 21 is very outlying.

A similar conclusion was obtained by Bilodeau and Duchesne [3] who introduced a robust method of the SUR model, which is like a multivariate regression model but with some constraints on the design matrix and the regression parameters. They analyzed a data set with two response variables, the annual gross investment of two corporations. Measures were taken from 1935 until 1954, with two independent variables: the outstanding shares and the real capital stock at the beginning of the year. An analysis of each company separately did reveal some good leverage points, whereas combining the information of the two corporations yielded two clear outliers which were hidden in the univariate analyzes.

[Figure 10 about here]

If we compare the prediction ability of the multivariate regression versus the four univariate regressions, we notice that the estimated prediction errors are close for all analytes except fat. The univariate analyzes yield  $\text{R-RMSEP}_6(\text{fat}) = 0.53$ ,  $\text{R-RMSEP}_2(\text{flour}) = 0.56$ ,  $\text{R-RMSEP}_3(\text{sucrose}) = 1.12$  and  $\text{R-RMSEP}_3(\text{water}) = 0.30$ . The multivariate approach on the other hand yields  $\text{R-RMSEP}_6(\text{fat}) = 1.17$ ,  $\text{R-RMSEP}_2(\text{flour}) = 0.45$ ,  $\text{R-RMSEP}_3(\text{sucrose}) = 1.23$  and  $\text{R-RMSEP}_3(\text{water}) = 0.35$ . When we analyze all the components together we thus only need two components to obtain almost the same prediction accuracy as the univariate regressions with 2 to 6 components. Fat seems to be the only concentration which is better

fitted by its own model.

To conclude this example, we also show some figures to illustrate that our RPCR method is hardly influenced by outliers, contrary to CPCR. For this we also estimated the PCA loadings and the calibration vectors on a reduced data set from which most striking outliers found by RPCR were excluded, namely observations 7, 20, 21, 22, 23 and 24. On Figure 11(a) we have plotted the robust second loading vector based on the full data set and the one based on the reduced data set, whereas Figure 11(b) contains the classical results. Notice how the C-H bend in Figure 11(b) becomes much more apparent when we apply CPCR on the reduced data set. The different estimates for the fat coefficients are plotted in Figure 11(c) and (d). As we would expect from a robust method, we see that the estimates for RPCR are hardly distinguishable, whereas the CPCR results clearly differ depending on whether or not outliers are present in the data set. The same phenomenon was observed for the other constituents which we therefore do not include.

Note that the RPCR and CPCR estimates on the reduced data set are very comparable but still differ slightly. This is because the complete RPCR method does not correspond with a reweighted CPCR procedure after elimination of the outliers. Reweighting steps are included throughout the procedure, but the data points which receive a zero weight in the ROBPCA step are not necessarily the same as those that are excluded in the regression step. This is because influential observations for the principal components are not necessarily regression outliers, and vice versa. In our example this implies e.g. that observation 21 is not yet excluded to estimate the (robust) principal components. CPCR on the reduced data set on the other hand gives zero weight to this data point from the start.

[Figure 11 about here]

## 6 Simulation study

To study more thoroughly the performance and the robustness of CPCR and RPCR, we have done an extensive simulation study.

The results being comparable for other choices of  $n$ ,  $p$ ,  $\beta_0$ ,  $\mathcal{B}$  and  $\Sigma_\varepsilon$ , we here report the results for the following design: we generated 1000 data sets with  $n = 50$  observations, and  $p = 100$  regressors. The  $x$ -variables were generated from the multivariate normal distribution

$N_p(\mathbf{0}_p, \Sigma)$  with  $\Sigma = \text{diag}(5, 4, 3, 2, 1, 0.095, \dots, 0.001)$  and the  $y$ -variables from  $N_q(\mathbf{0}_q, I_{q,q})$ . Note that this choice corresponds with  $\beta_0 = \mathbf{0}_q$ ,  $\mathcal{B} = \mathbf{0}_{q,q}$  and  $\Sigma_\epsilon = I_{q,q}$  in model (2). The number of response variables were chosen as  $q = 2$  and  $q = 5$ .

Next we randomly contaminated 100 $\epsilon\%$  (with  $\epsilon = 0.1$  or  $\epsilon = 0.2$ ) of our observations according to the different types of outliers presented in Table 2. Let  $X_\epsilon$  and  $Y_\epsilon$  denote the corrupted design and response matrices, then they were constructed as follows:

1. Bad leverage points:

$$X_\epsilon \sim N_p(\tilde{\boldsymbol{\mu}}, \Sigma) \text{ with } \tilde{\boldsymbol{\mu}} = (0, 0, 0, 0, 1, 0, 0, \dots, 0)' \text{ and } Y_\epsilon \sim N_q(5\chi_{q,0.99}^2 \mathbf{1}_q, I_{q,q})$$

2. Vertical outliers:  $X_\epsilon \sim N_p(\mathbf{0}_p, \Sigma)$  and  $Y_\epsilon \sim N_q(\chi_{p+q,0.99}^2 \mathbf{1}_q, I_{q,q})$

3. Good leverage points:  $X_\epsilon \sim N_p(\tilde{\boldsymbol{\mu}}, \Sigma)$ , with  $\tilde{\boldsymbol{\mu}} = (0, 0, 0, 0, 1, 0, 0, \dots, 0)'$  and  $Y_\epsilon \sim N_q(\mathbf{0}_q, I_{q,q})$ .

For each simulation setup and for each of the 1000 generated data set, we computed the  $\beta_0$ ,  $\mathcal{B}$  and the  $\Sigma_\epsilon$  estimates according to CPCR and RPCR. Because the design matrix by construction induces 5 significant eigenvalues, we always used  $k = 5$ . The RPCR estimator was applied with the default  $h$ -value of  $h = [0.75 * n] = 37$ . In Tables 3 and 4 we report the bias and the MSE of the slope matrix  $\hat{\mathcal{B}}$ , the intercept vector  $\hat{\beta}_0$ , the diagonal elements of  $\Sigma_\epsilon$  and the off-diagonal elements of  $\Sigma_\epsilon$  following the definitions in [25].

### Discussion

The first columns of Tables 3 and 4 show that CPCR is the preferred method when there is no contamination in the data. But as soon as we introduce outliers, CPCR becomes very biased and unstable. The estimates of the slopes, the intercept and the error covariance of CPCR are all highly distorted when there are bad leverage points or vertical outliers. The RPCR slope and intercept estimates on the other hand are comparable to the uncontaminated situation, even at a high contamination level of 20%. Finally, the last column of Table 3 shows that good leverage points have almost no influence on the regression parameters. Only the MSE of the robust error covariance matrix has somewhat increased.

[Tables 3 and 4 about here]

## 7 Conclusion

We have presented a robust PCR method which is able to handle high-dimensional spectra and several concentration variables at once. Because fast algorithms are available for the two cornerstones of RPCR, namely ROBPCA and LTS/MCD-regression, our method can be applied to small as well as to large data sets. A user-friendly Matlab implementation can be obtained from the authors and will be available at the web sites [win-www.uia.ac.be/u/statis](http://win-www.uia.ac.be/u/statis) and [www.kuleuven.ac.be/ucs/research/stats.htm](http://www.kuleuven.ac.be/ucs/research/stats.htm).

Simulations and the analysis of a real data set have shown the robustness of RPCR towards outliers in the data and the usefulness of the multivariate approach. A robust RMSECV, RMSEP, and  $R^2$ -value and several diagnostic plots are introduced as valuable exploratory tools in the analysis of real chemometrical data and the detection of outliers.

Several issues remain open for research. In this paper we mainly focused on the estimation of the parameters and the detection of the outliers. Model validation is however also very important. First of all, it would be interesting to construct a faster and/or a more precise robust estimate of the prediction error than our actual cross-validated R-RMSEP statistic. For this we will investigate the use and the performance of a robust  $C_p$  measure [21] in RPCR. We will also evaluate replacing the trace of the determinant in (29) by its determinant.

Graphical techniques which are recently developed in robust cluster analysis [9] will be generalized to the PCA and PCR setting and might give good indications to choose the number of principal components  $k$  and the proportion of contamination  $\alpha$ .

Besides, we have developed robust PLS methods which are mainly based on ROBPCA [12]. A thorough comparison of RPCR and RPLS will be performed to investigate their accuracy and speed at different data configurations and levels of contamination.

Currently we are incorporating these algorithms in a Matlab library of robust functions for calibration and data analysis. This library will soon be available at the earlier mentioned web sites.

## References

- [1] Agulló J, Croux C, Van Aelst S. The multivariate least trimmed squares estimator. *Submitted.*

- [2] Beebe KR, Pell RJ, Seasholtz MB. *Chemometrics, a practical guide*. John Wiley: New York, 1998.
- [3] Bilodeau M, Duchesne P. Robust estimation of the SUR model. *The Canadian Journal of Statistics* 2000; **28**: 277–288.
- [4] Coakley CW, Hettmansperger TP. A bounded influence, high breakdown efficient regression estimator. *J. Am. Statist. Assoc.* 1993; **88**: 872–880.
- [5] Croux C, Haesbroeck G. Influence function and efficiency of the minimum covariance determinant scatter matrix estimator. *J. Multiv. Anal.* 1999; **71**: 161–190.
- [6] Croux C, Haesbroeck G. Principal components analysis based on robust estimators of the covariance or correlation matrix: influence functions and efficiencies. *Biometrika* 2000; **87**: 603–618.
- [7] De Jong S. SIMPLS: an alternative approach to partial least squares regression. *Chemometrics Intell. Lab. Syst.* 1993; **18**: 251–263.
- [8] Egan WJ, Morgan SL. Outlier detection in multivariate analytical chemical data. *Anal. Chem.* 1998; **70**: 2372–2379.
- [9] García-Escudero LA, Gordaliza A, Matrán C. Trimming tools in exploratory data analysis. To appear in *J. Comput. Graph. Statist.*
- [10] Hastie T, Tibshirani R, Friedman J. *The Elements of Statistical Learning*. Springer-Verlag: New York, 2001.
- [11] Hubert M, Rousseeuw PJ, Vanden Branden K. ROBPCA: a new approach to robust principal component analysis. *Catholic University Leuven, Technical report 2002-06* 2002. *Under revision*.
- [12] Hubert M, Vanden Branden K. Robust methods for partial least squares regression. *In preparation*.
- [13] Hubert M, Rousseeuw PJ, Verboven S. A fast method for robust principal components with applications to chemometrics. *Chemometrics Intell. Lab. Syst.* 2002; **60**: 101–111.

- [14] Johnson R, Wichern D. *Applied Multivariate Statistical Analysis* (4th Edition), Prentice Hall: New Jersey, 1998.
- [15] Lopuhaä HP, Rousseeuw PJ. Breakdown points of affine equivariant estimators of multivariate location and covariance matrices. *Ann. Statist.* 1991; **19**: 229–248.
- [16] Maronna R, Yohai V. The behavior of the Stahel-Donoho robust multivariate estimator. *J. Am. Statist. Assoc.* 1995; **90**: 330–341.
- [17] Martens H, Naes T. *Multivariate Calibration*, John Wiley: New York, 1998.
- [18] Marx BD, Eilers PH. Generalized linear regression on sampled signals and curves: a P-spline approach. *Technometrics* 1999; **41**: 1–13.
- [19] Osborne BG, Fearn T, Miller AR, Douglas S. Application of near infrared reflectance spectroscopy to the compositional analysis of biscuits and biscuit dough. *J. Scient. Food Agric.* 1984; **35**: 99–105.
- [20] Pell RJ. Multiple outlier detection for multivariate calibration using robust statistical techniques. *Chemometrics Intell. Lab. Syst.* 2000; **52**: 87–104.
- [21] Ronchetti E, Staudte RG. A robust version of Mallows’s  $C_p$ . *J. Am. Statist. Assoc.* 1994; **89**: 550–559.
- [22] Rousseeuw PJ. Least median of squares regression. *J. Am. Statist. Assoc.* 1984; **79**: 871–880.
- [23] Rousseeuw PJ, Hubert M. Recent developments in PROGRESS, in  *$L_1$ -Statistical Procedures and Related Topics*, edited by Y. Dodge. Institute of Mathematical Statistics Lecture Notes-Monograph Series, Volume 31, Hayward: California, 1997, 201–214.
- [24] Rousseeuw PJ, Leroy A. *Robust Regression and Outlier Detection*, John Wiley: New York, 1987.
- [25] Rousseeuw PJ, Van Aelst S, Van Driessen K, Agulló A. Robust multivariate regression. *Technical Report, University of Antwerp* 2000. *Under revision.*
- [26] Rousseeuw PJ, Van Driessen K. A fast algorithm for the minimum covariance determinant estimator. *Technometrics* 1999; **41**: 212–223.

- [27] Rousseeuw PJ, Van Driessen K. Computing LTS regression for large data sets. To appear in *Estadistica*.
- [28] Rousseeuw PJ, Van Zomeren BC. Unmasking multivariate outliers and leverage points. *J. Am. Statist. Assoc.* 1990; **85**: 633–651.
- [29] Rousseeuw PJ, Yohai V. Robust regression by means of S estimators. *Robust and Non-linear Time Series Analysis. Lecture Notes in Statist.* 1984; **26**: 256–272. Springer, New York.
- [30] Simpson DG, Ruppert D, Carroll RJ. On one-step GM estimates and stability of inferences in linear regression. *J. Am. Statist. Assoc.* 1992; **85**: 633–651.
- [31] *Splus 6 Robust Library User's Guide*, Insightful Corporation: Seattle, WA, 2001.
- [32] Vandeginste BGM, Massart DL, Buydens LMC et al., *Handbook of Chemometrics and Qualimetrics: Part B*, Elsevier, Amsterdam, 1998.

CLASSICAL AND ROBUST TOLERANCE ELLIPSE (97.5%)

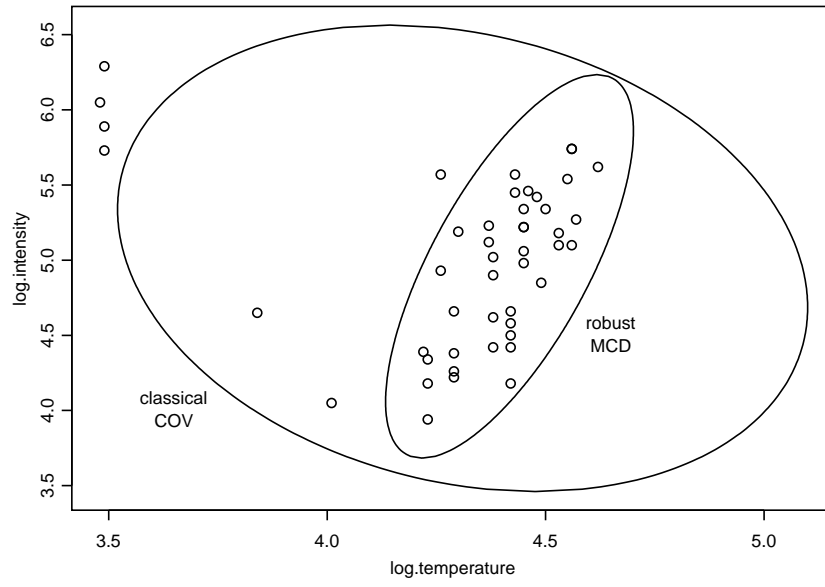
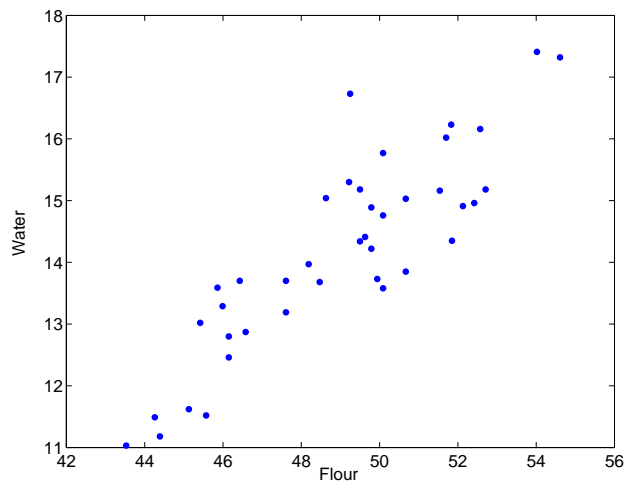
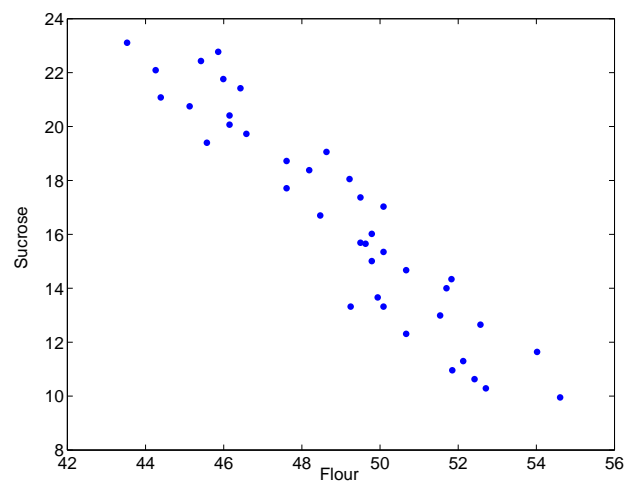


Figure 1: Classical and robust tolerance ellipse of the Stars data set.

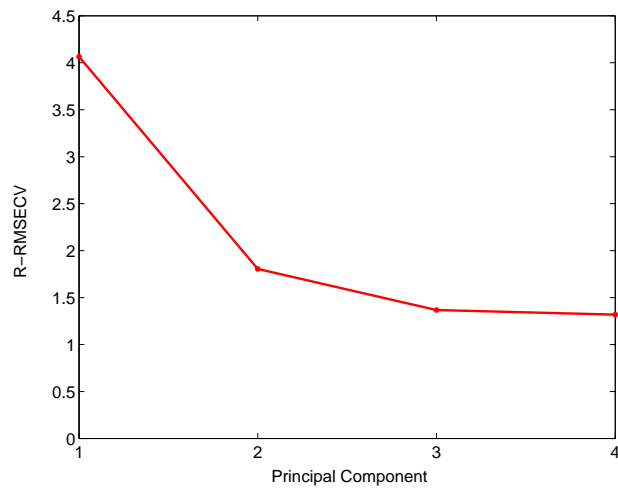


(a)

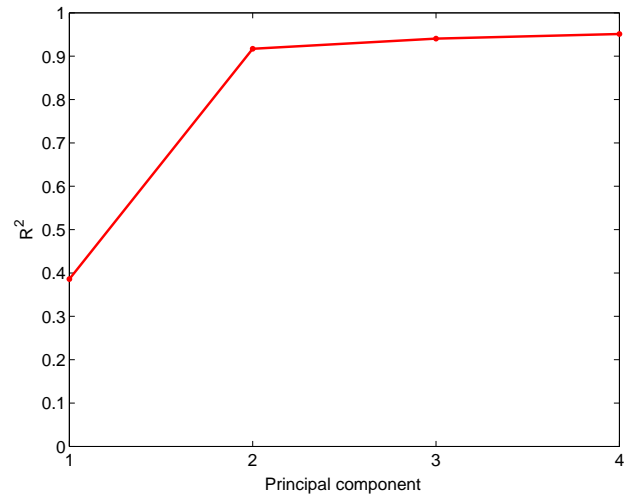


(b)

Figure 2: Scatter plot of some of the Biscuit Dough response variables showing a positive correlation between (a) water ( $y_4$ ) and flour ( $y_2$ ), and a negative correlation between (b) sucrose ( $y_3$ ) and flour ( $y_2$ ).



(a)



(b)

Figure 3: Robust RMSECV and robust  $R^2$  curve of the Biscuit NIR data set.

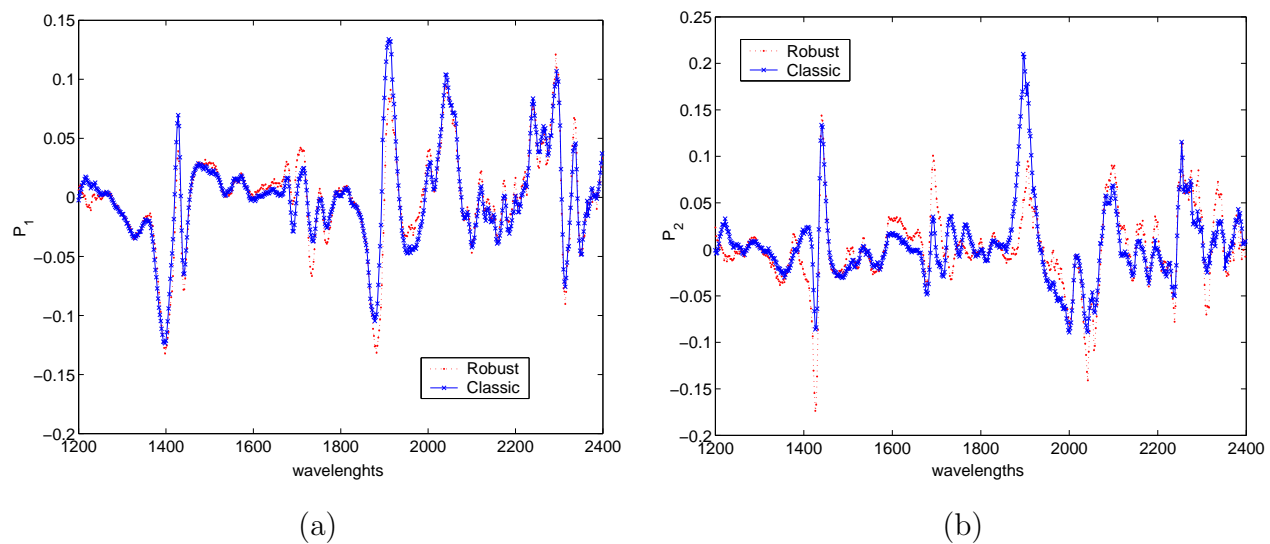


Figure 4: Robust and classical PCA analysis of the Biscuit NIR data set: (a) first loading vector; (b) second loading vector.

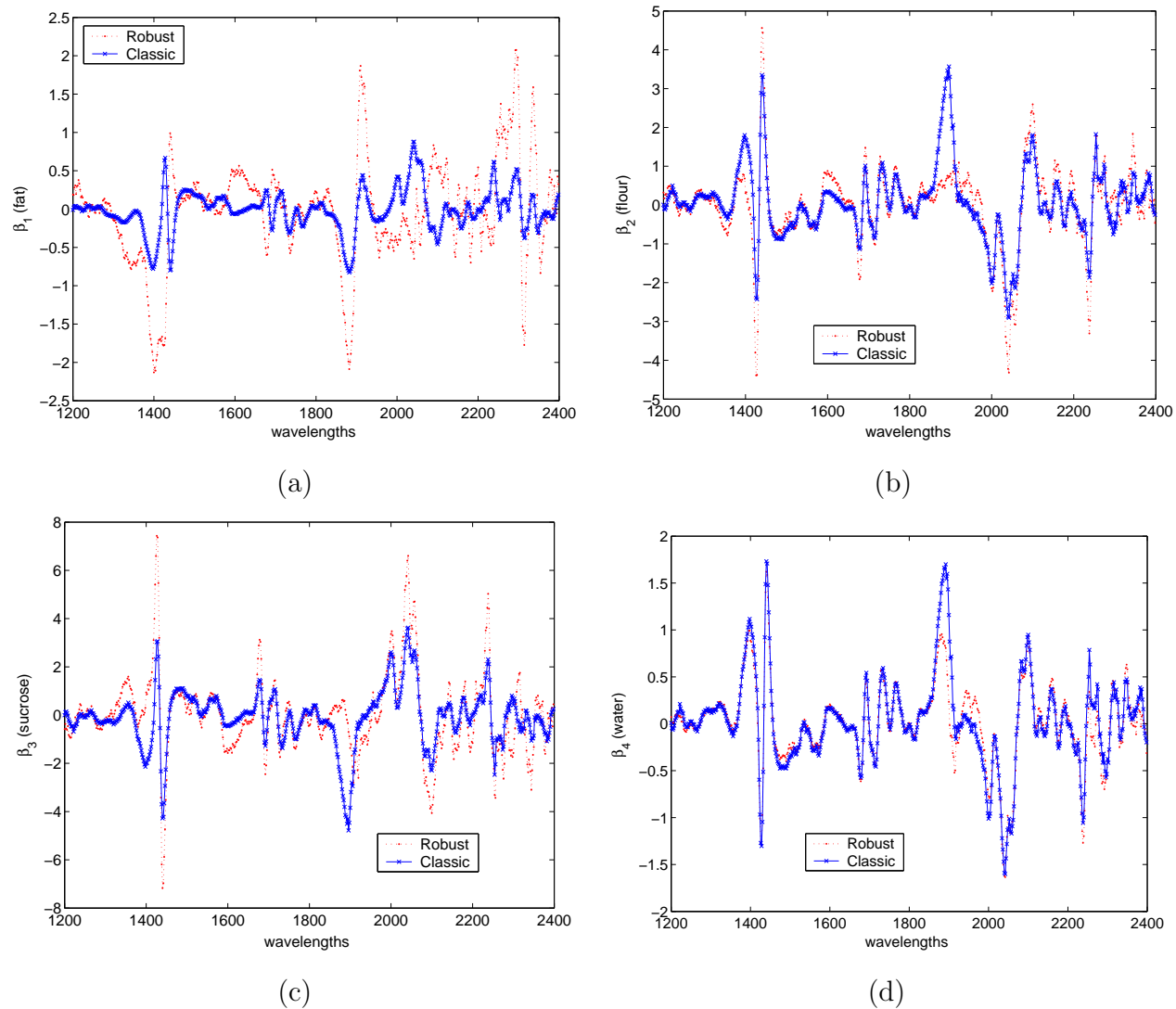


Figure 5: Robust and classical PCR analysis of the Biscuit NIR data set. Estimated calibration vectors for: (a) fat; (b) flour; (c) sucrose; (d) water.

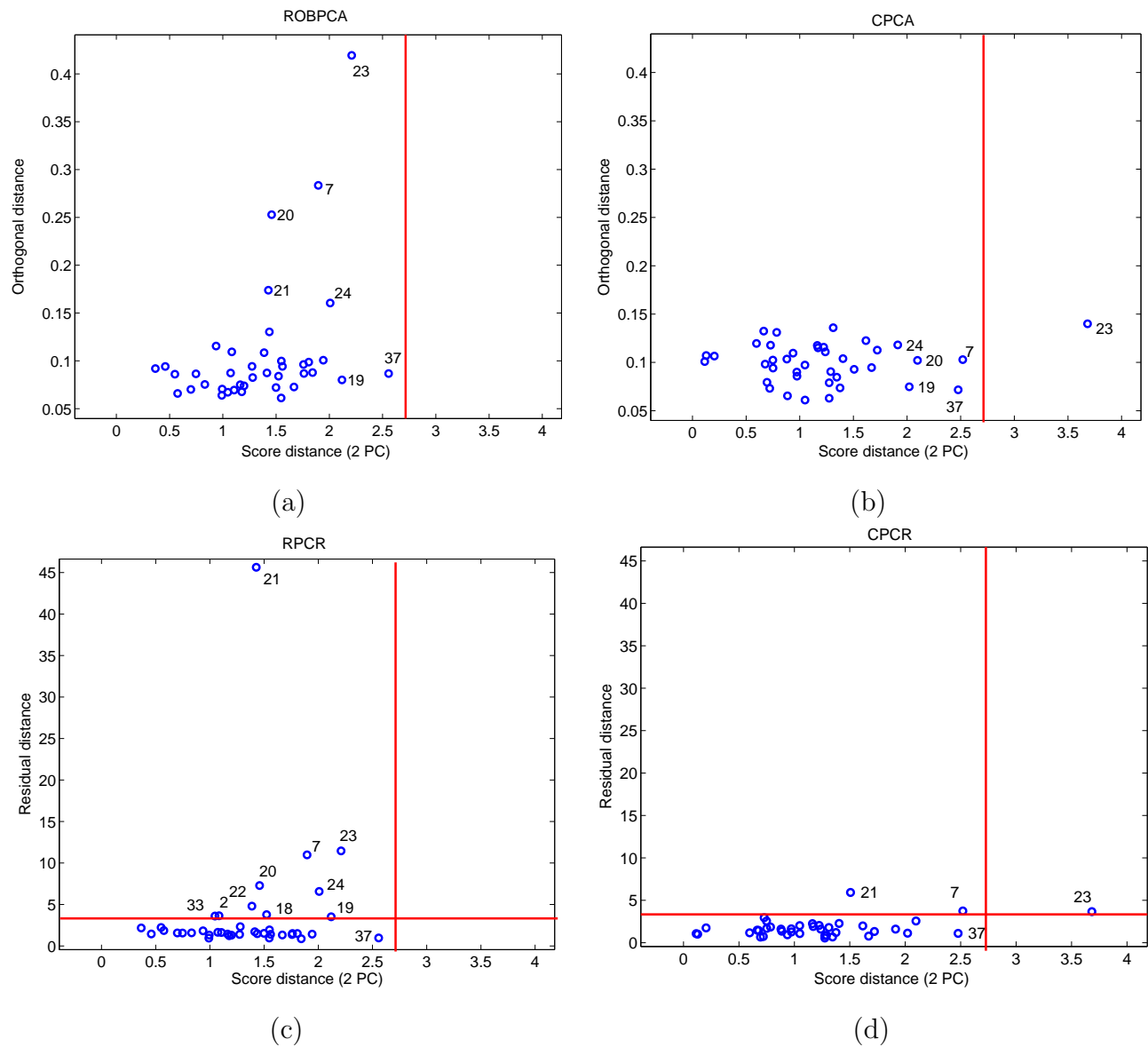
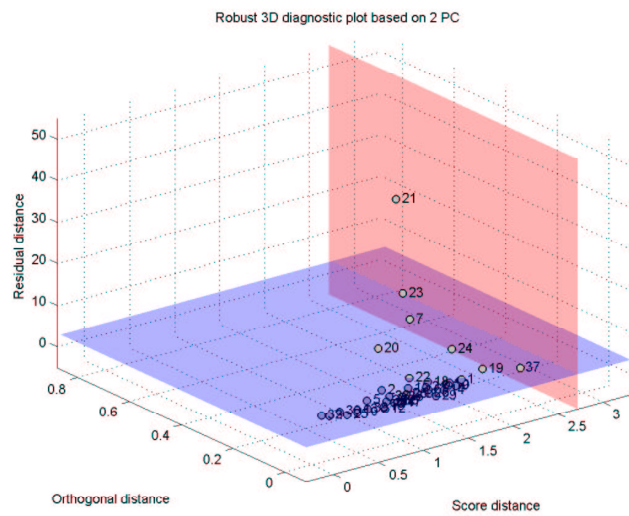
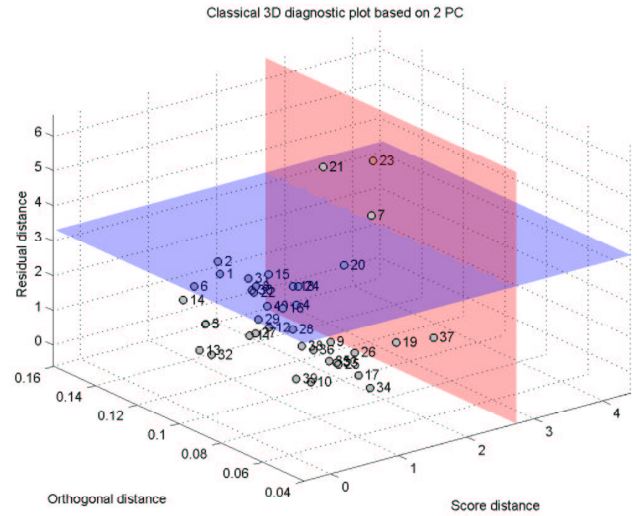


Figure 6: Diagnostic plots of the Biscuit NIR data set: (a) robust PCA plot; (b) classical PCA plot; (c) robust PCR plot; (d) classical PCR plot.

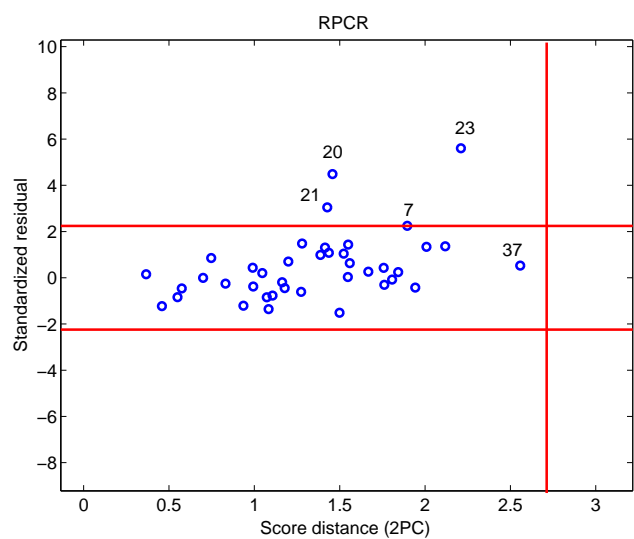


(a)

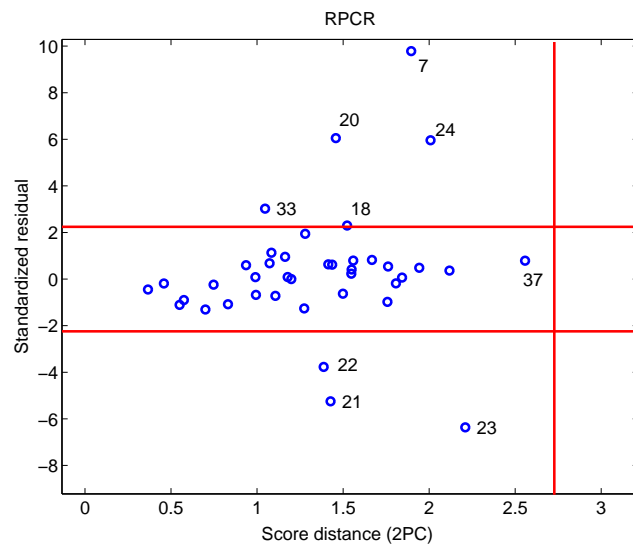


(b)

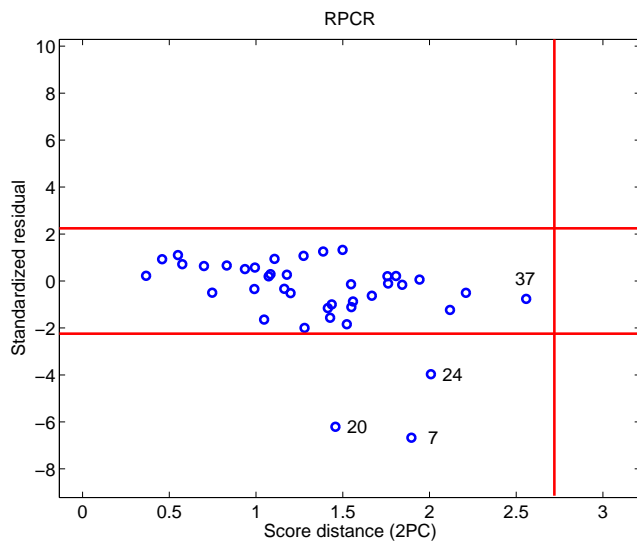
Figure 7: Three-dimensional diagnostic plots of the Biscuit NIR data set for: (a) RPCR; (b) CPCR.



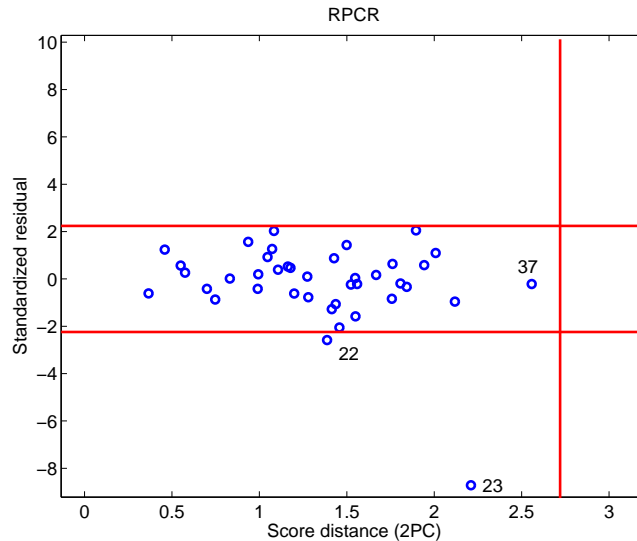
(a)



(b)

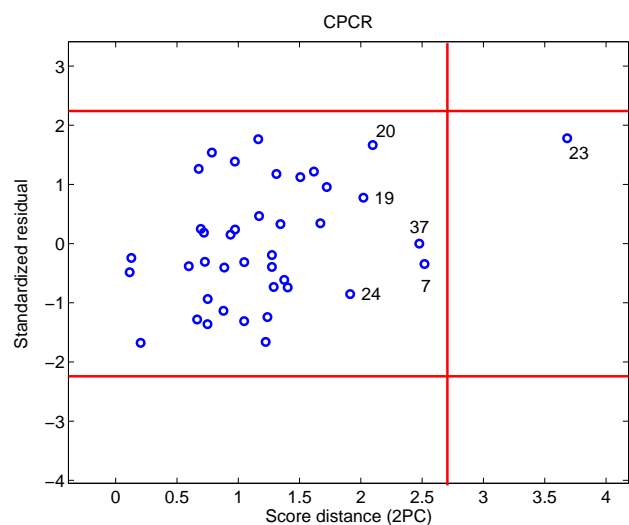


(c)

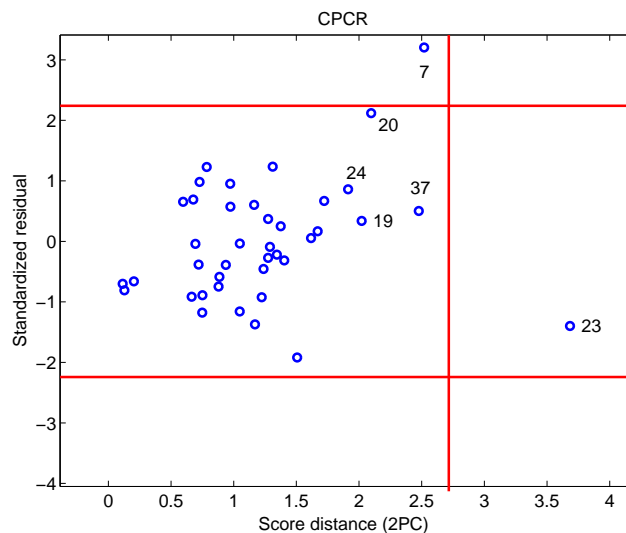


(d)

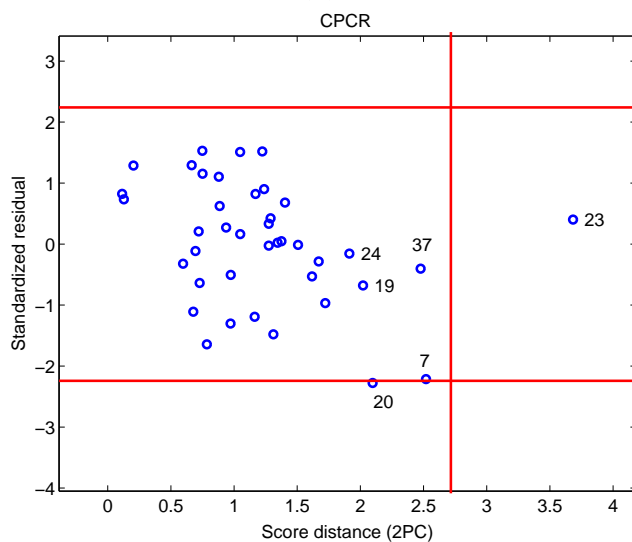
Figure 8: Univariate residual plots of the Biscuit NIR data set based on the multivariate RPCR for: (a)  $y_1 = \text{fat}$ ; (b)  $y_2 = \text{flour}$ ; (c)  $y_3 = \text{sucrose}$ ; (d)  $y_4 = \text{water}$ .



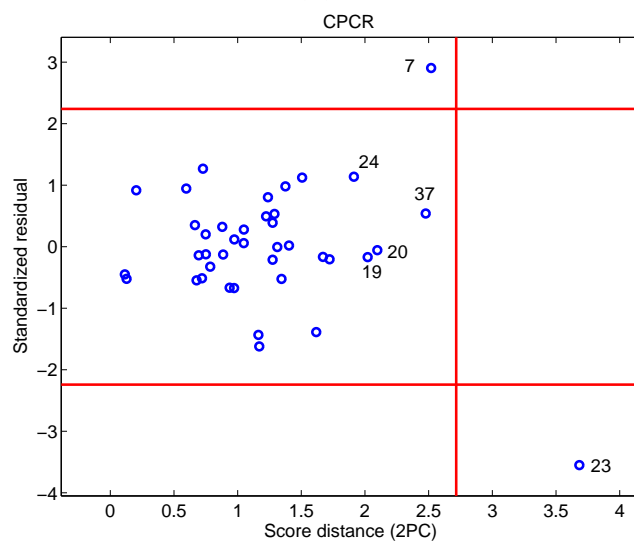
(a)



(b)



(c)



(d)

Figure 9: Univariate residual plots of the Biscuit NIR data set based on the multivariate PCR for: (a)  $y_1 = \text{fat}$ ; (b)  $y_2 = \text{flour}$ ; (c)  $y_3 = \text{sucrose}$ ; (d)  $y_4 = \text{water}$ .

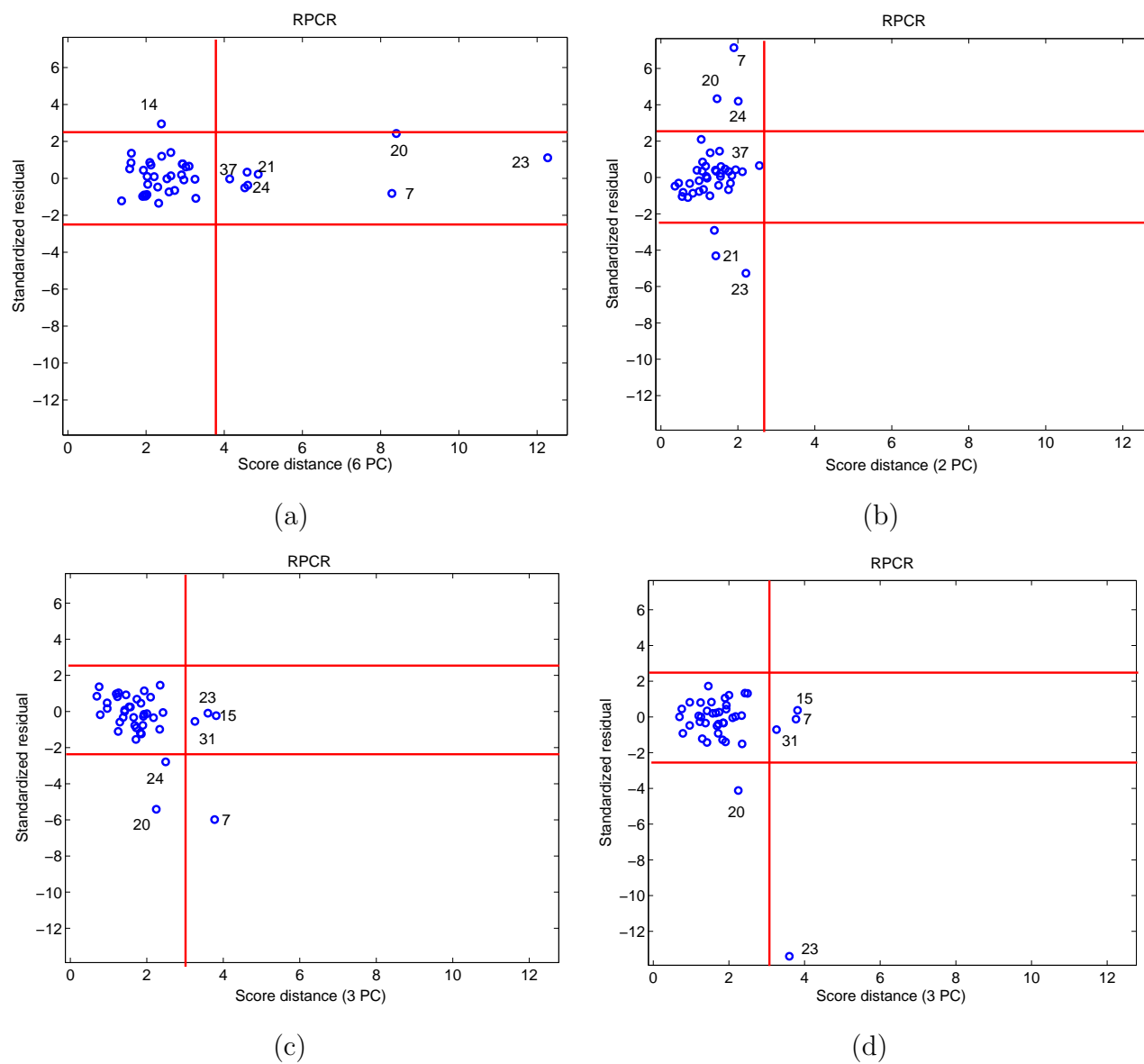


Figure 10: Univariate residual plots of the Biscuit NIR data set based on four univariate RPCR analyses for: (a)  $y_1 = \text{fat}$ ; (b)  $y_2 = \text{flour}$ ; (c)  $y_3 = \text{sucrose}$ ; (d)  $y_4 = \text{water}$ .

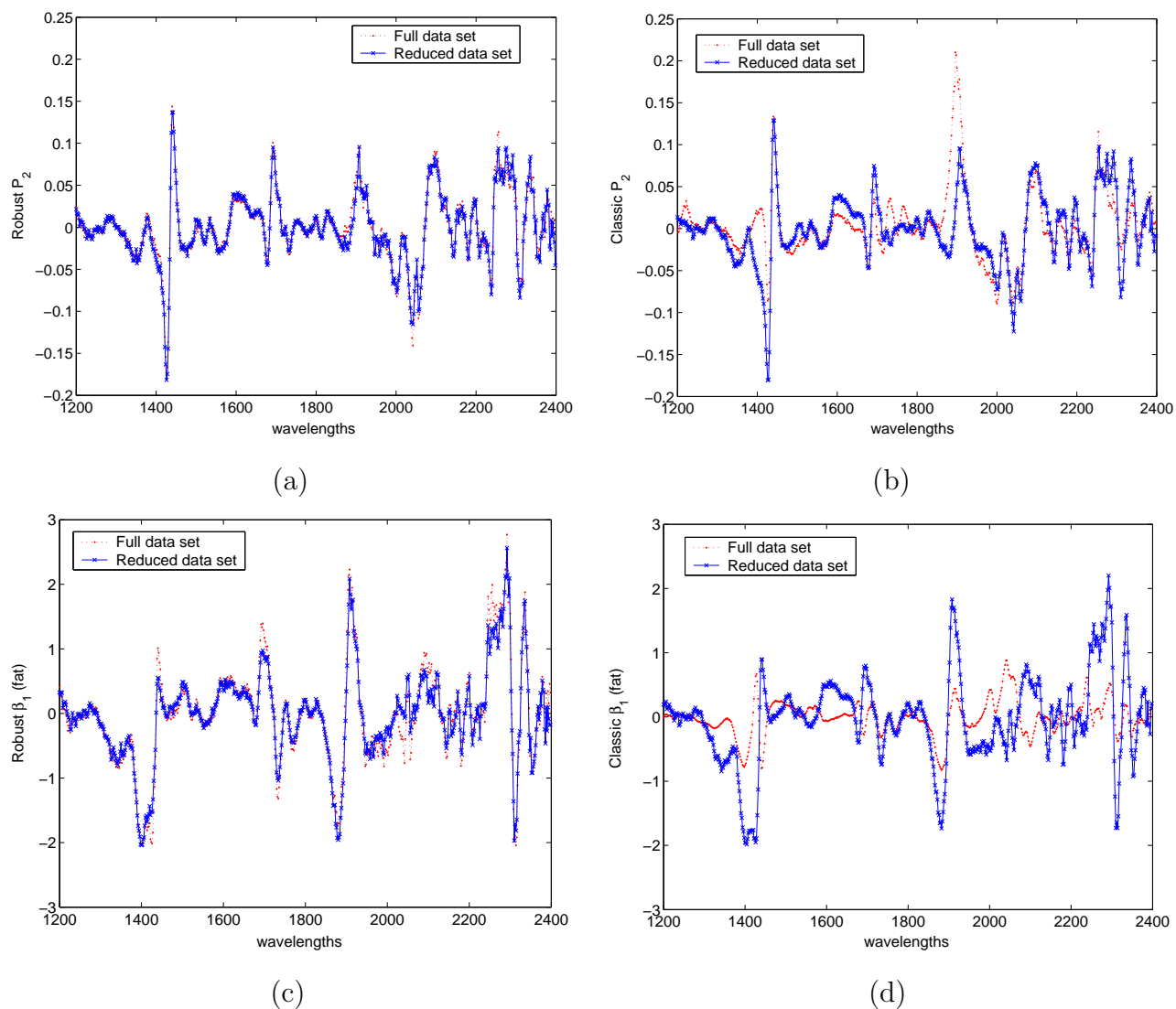


Figure 11: Analysis of the Biscuit NIR data set on the full and a reduced data set: (a) second robust loading vector; (b) second classical loading vector; (c) robust calibration vector for fat; (d) classical calibration vector for fat.

| Distances | small SD            | large SD                |
|-----------|---------------------|-------------------------|
| large OD  | orthogonal outlier  | bad PCA-leverage point  |
| small OD  | regular observation | good PCA-leverage point |

Table 1: Overview of the different types of observations based on their score distance and their orthogonal distance.

| Distances | small SD            | large SD            |
|-----------|---------------------|---------------------|
| large RD  | vertical outlier    | bad leverage point  |
| small RD  | regular observation | good leverage point |

Table 2: Overview of the different types of observations based on their score distance and their residual distance.

|               | $\epsilon = 0\%$ |        | $\epsilon = 10\%$   |        | $\epsilon = 20\%$    |        |
|---------------|------------------|--------|---------------------|--------|----------------------|--------|
|               | CPCR             | RPCR   | bad leverage points |        | good leverage points |        |
|               |                  |        | CPCR                | RPCR   | CPCR                 | RPCR   |
| bias(slope)   | 0.0006           | 0.0018 | 0.1360              | 0.0012 | 0.0006               | 0.0009 |
| MSE(slope)    | 0.0247           | 0.0707 | 0.9815              | 0.0572 | 0.0178               | 0.0441 |
| bias(interc)  | 0.0062           | 0.0118 | 0.1528              | 0.0095 | 0.0027               | 0.0063 |
| MSE(interc)   | 1.0905           | 2.2677 | 4.4901              | 2.0812 | 1.3807               | 1.7786 |
| bias(diag)    | 0.1024           | 0.0765 | 1.7905              | 0.1220 | 0.1099               | 0.2220 |
| MSE(diag)     | 2.3024           | 4.5027 | 178.51              | 6.0551 | 2.3707               | 9.3345 |
| bias(offdiag) | 0.0104           | 0.0105 | 1.8901              | 0.0038 | 0.0018               | 0.0132 |
| MSE(offdiag)  | 0.8464           | 2.8299 | 191.38              | 3.4655 | 0.8530               | 2.8594 |

Table 3: Simulation results for  $q = 2, n = 50$  and  $p = 100$ .

|               | $\epsilon = 0\%$ |        | $\epsilon = 10\%$ |        | $\epsilon = 20\%$   |        |
|---------------|------------------|--------|-------------------|--------|---------------------|--------|
|               | CPCR             | RPCR   | vertical outliers |        | bad leverage points |        |
|               |                  |        | CPCR              | RPCR   | CPCR                | RPCR   |
| bias(slope)   | 0.0006           | 0.0011 | 0.0040            | 0.0013 | 0.1822              | 0.0009 |
| MSE(slope)    | 0.0245           | 0.0623 | 0.3416            | 0.0551 | 1.7400              | 0.0438 |
| bias(interc)  | 0.0027           | 0.0077 | 1.1944            | 0.0066 | 0.2351              | 0.0051 |
| MSE(interc)   | 1.0768           | 2.0940 | 73.623            | 1.9380 | 8.8361              | 1.7812 |
| bias(diag)    | 0.1018           | 0.0113 | 11.521            | 0.0747 | 3.1392              | 0.1971 |
| MSE(diag)     | 2.3854           | 5.1105 | 6678.9            | 5.6529 | 533.58              | 7.5685 |
| bias(offdiag) | 0.0049           | 0.0071 | 11.703            | 0.0088 | 3.2206              | 0.0059 |
| MSE(offdiag)  | 0.9036           | 2.6948 | 6883.0            | 2.8375 | 551.75              | 2.8357 |

Table 4: Simulation results for  $q = 5, n = 50$  and  $p = 100$ .