# Robust Methods for Partial Least Squares Regression

M. Hubert *and K. Vanden Branden

9th October 2003 (Revised version)

SUMMARY

Partial Least Squares Regression (PLSR) is a linear regression technique developed to deal with high-dimensional regressors and one or several response variables. In this paper we introduce robustified versions of the SIMPLS algorithm being the leading PLSR algorithm because of its speed and efficiency. Because SIMPLS is based on the empirical cross-covariance matrix between the response variables and the regressors and on linear least squares regression, the results are affected by abnormal observations in the data set. Two robust methods, RSIMCD and RSIMPLS, are constructed from a robust covariance matrix for high-dimensional data and robust linear regression. We introduce robust RMSECV and RMSEP values for model calibration and model validation. Diagnostic plots are constructed to visualize and classify the outliers. Several simulation results and the analysis of real data sets show the effectiveness and the robustness of the new approaches. Because RSIMPLS is roughly twice as fast as RSIMCD, it stands out as the overall best method.

KEY WORDS : Partial Least Squares Regression; SIMPLS; Principal Component Analysis; Robust Regression.

*Correspondence to: M. Hubert, Assistant Professor, Department of Mathematics, Katholieke Universiteit Leuven, W. de Croylaan 54, B-3001 Leuven, Belgium, mia.hubert@wis.kuleuven.ac.be, tel.+32/16322023, fax.+32/16322831

# 1 Introduction

The PLS (NIPALS) regression technique was originally developed for econometrics by Wold ([29], [30]), but has become a very popular algorithm in other fields such as chemometrics, social science, food industry, etc. It is used to model the linear relation between a set of regressors and a set of response variables, which can then be used to predict the value of the response variables for a new sample. A typical example is multivariate calibration where the $x$-variables are spectra and the $y$-variables are the concentrations of certain constituents.

Throughout the paper we will print column vectors in bold and the transpose of a vector $\boldsymbol{v}$ or a matrix $V$ as $\boldsymbol{v}'$ or $V'$. Sometimes, the dimension of a matrix will be denoted using a subscript, e.g. $X_{n,p}$ stands for a $(n \times p)$ dimensional matrix. We apply the notation $(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)' = X_{n,p}$ for the regressors and $(\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n)' = Y_{n,q}$ for the response variables. The merged data set $(X_{n,p}, Y_{n,q})$ will be denoted as $Z_{n,m}$, with $m = p + q$.

The linear regression model we consider is:

$$\boldsymbol{y}_i = \boldsymbol{\beta}_0 + \mathcal{B}'_{q,p}\boldsymbol{x}_i + \boldsymbol{e}_i, \tag{1}$$

where the error terms $\boldsymbol{e}_i$ satisfy $E(\boldsymbol{e}_i) = \mathbf{0}$ and $\mathrm{Cov}(\boldsymbol{e}_i) = \Sigma_e$ of size $q$. The unknown $q$-dimensional intercept is denoted as $\boldsymbol{\beta}_0 = (\beta_{01}, \ldots, \beta_{0q})'$ and $\mathcal{B}_{p,q}$ represents the unknown slope matrix. Typically in chemometrics, the number of observations $n$ is very small (some tens), whereas the number of regressors is very numerous (some hundreds, thousands). The number of response variables $q$ is in general limited to at most five.

Because multicollinearity is present, the classical multiple linear regression (MLR) estimates have too large of a variance, hence biased estimation procedures, such as Principal Component Regression (PCR) and Partial Least Squares Regression (PLSR) [13], are then performed. In this paper, we will focus on PLSR. We will use the notation PLS1 when there is only one response variable and the notation PLS2 otherwise.

It is well known that the popular algorithms for PLSR (NIPALS [29] and SIMPLS [2]) are very sensitive to outliers in the data set. This will also be clearly shown in our simulation study in Section 6. In this paper, we introduce several robust methods for PLSR which are resistant to outlying observations. Some robustified versions of the PLS1 and PLS2 algorithms have already been proposed in the past. A first algorithm [28] has been developed by replacing the different univariate regression steps in the PLS2 algorithm by some robust alternatives. Iteratively reweighted algorithms have been obtained in [1] and [16]. These algorithms are only valid for a one-dimensional response variable and they are not resistant to leverage points. In [5], a robust PLS1 method is obtained by robustifying the sample covariance matrix of the $x$-variables and the sample cross-covariance matrix between the $x$- and $y$-variables. For this, the highly robust Stahel-Donoho estimator ([23],[3]) is used, but unfortunately it can not be applied to high-dimensional regressors ($n \ll p$) because

the subsampling scheme used to compute the estimator starts by drawing subsets of size $p+2$. Moreover the method can not be extended to PLS2. Recently, a robust method for PCR which also applies to high-dimensional $x$-variables and multiple $y$-variables has been introduced in [9].

In this paper we present several robustifications of the SIMPLS algorithm. SIMPLS is very popular because it is faster than PLS1 and PLS2 as implemented using NIPALS, and the results are easier to interpret. Moreover, if there is only one response variable ($q = 1$), SIMPLS and PLS1 yield the same results. An outline of the SIMPLS algorithm is presented in Section 2. We recall that SIMPLS depends on the sample cross-covariance matrix between the $x$- and $y$-variables, and on linear least squares regression. In Section 3, we introduce several robust methods which are obtained by using a robust covariance matrix for high-dimensional data sets ([8]), and a robust regression method. The proposed algorithms are fast compared to previous developed robust methods and they can handle cases where $n \ll p$ and $q \geq 1$. Section 4 discusses the selection of the number of components and the model validation. In Section 5 we introduce several diagnostic plots which can help us to identify the outliers and classify them in several types. The robustness of the proposed algorithms is demonstrated in Section 6 with several simulations. In Section 7 we apply one of the new methods on a real data set. We finish with some conclusions in Section 8.

# 2 The SIMPLS algorithm

The SIMPLS method assumes that the $x$- and $y$-variables are related through a bilinear model:

$$\boldsymbol{x}_i = \bar{\boldsymbol{x}} + P_{p,k}\tilde{\boldsymbol{t}}_i + \boldsymbol{g}_i \tag{2}$$

$$\boldsymbol{y}_i = \bar{\boldsymbol{y}} + \mathcal{A}'_{q,k}\tilde{\boldsymbol{t}}_i + \boldsymbol{f}_i. \tag{3}$$

In this model, $\bar{\boldsymbol{x}}$ and $\bar{\boldsymbol{y}}$ denote the mean of the $x$- and the $y$-variables. The $\tilde{\boldsymbol{t}}_i$ are called the scores which are $k$-dimensional, with $k \ll p$, whereas $P_{p,k}$ is the matrix of $x$-loadings. The residuals of each equation are represented by the $\boldsymbol{g}_i$ and $\boldsymbol{f}_i$ respectively. The matrix $\mathcal{A}_{k,q}$ represents the slope matrix in the regression of $\boldsymbol{y}_i$ on $\tilde{\boldsymbol{t}}_i$. Note that in the literature regarding PLS, the matrix $\mathcal{A}_{k,q}$ is usually denoted as $Q'_{k,q}$ and the columns of $Q_{q,k}$ by the $y$-loadings $\boldsymbol{q}_a$. We prefer to use another notation because $\boldsymbol{q}_a$ will be used for the PLS weight vector, see (4).

The bilinear structure (2) and (3) implies a two-steps algorithm. After mean-centering the data, SIMPLS will first construct $k$ latent variables $\tilde{T}_{n,k} = (\tilde{\boldsymbol{t}}_1, \ldots, \tilde{\boldsymbol{t}}_n)'$ and secondly, the responses will be regressed onto these $k$ variables. We will refer to the columns of $\tilde{T}_{n,k}$ as the components.

Consider first the construction of the components. In contrast with PCR, the $k$ components are not solely determined based on the $x$-variables. They are obtained as a linear combination of the $x$-variables which have maximum covariance with a certain linear combination of the $y$-variables. More precisely, let $\tilde{X}_{n,p}$ and $\tilde{Y}_{n,q}$ denote the mean-centered data matrices, with $\tilde{\boldsymbol{x}}_i = \boldsymbol{x}_i - \bar{\boldsymbol{x}}$ and $\tilde{\boldsymbol{y}}_i = \boldsymbol{y}_i - \bar{\boldsymbol{y}}$. The normalized PLS weight vectors $\boldsymbol{r}_a$ and $\boldsymbol{q}_a$ ($\|\boldsymbol{r}_a\| = \|\boldsymbol{q}_a\| = 1$) are then defined as the vectors that maximize for each $a = 1, \ldots, k$

$$\mathrm{cov}(\tilde{Y}_{n,q}\boldsymbol{q}_a, \tilde{X}_{n,p}\boldsymbol{r}_a) = \boldsymbol{q}_a' \frac{\tilde{Y}_{q,n}' \tilde{X}_{n,p}}{n-1} \boldsymbol{r}_a = \boldsymbol{q}_a' S_{yx} \boldsymbol{r}_a \tag{4}$$

where $S_{yx}' = S_{xy} = \frac{\tilde{X}_{p,n}' \tilde{Y}_{n,q}}{n-1}$ is the empirical cross-covariance matrix between the $x$- and $y$-variables. The elements of the scores $\tilde{\boldsymbol{t}}_i$ are then defined as linear combinations of the mean-centered data: $\tilde{t}_{ia} = \tilde{\boldsymbol{x}}_i' \boldsymbol{r}_a$, or equivalently $\tilde{T}_{n,k} = \tilde{X}_{n,p} R_{p,k}$ with $R_{p,k} = (\boldsymbol{r}_1, \ldots, \boldsymbol{r}_k)$. The maximization problem of (4) has one straightforward solution: $\boldsymbol{r}_1$ and $\boldsymbol{q}_1$ are the first left and right singular vectors of $S_{xy}$. This implies that $\boldsymbol{q}_1$ is the dominant eigenvector of $S_{yx}S_{xy}$ and $\boldsymbol{r}_1 = S_{xy}\boldsymbol{q}_1$. To obtain more than one solution, the components $\tilde{X}\boldsymbol{r}_j$ are required to be orthogonal:

$$\boldsymbol{r}_j' \tilde{X}' \tilde{X} \boldsymbol{r}_a = \sum_{i=1}^{n} \tilde{t}_{ij} \tilde{t}_{ia} = 0, \; a > j. \tag{5}$$

To satisfy this condition we first introduce the $x$-loading $\boldsymbol{p}_j$ that describes the linear relation between the $x$-variables and the $j$th component $\tilde{X}\boldsymbol{r}_j$. It is computed as

$$\begin{aligned} \boldsymbol{p}_j &= (\boldsymbol{r}_j' \tilde{X}' \tilde{X} \boldsymbol{r}_j)^{-1} \tilde{X}' \tilde{X} \boldsymbol{r}_j \\ &= (\boldsymbol{r}_j' S_x \boldsymbol{r}_j)^{-1} S_x \boldsymbol{r}_j \end{aligned} \tag{6}$$

with $S_x$ the empirical covariance matrix of the $x$-variables. This definition implies that (5) is fulfilled when $\boldsymbol{p}_j' \boldsymbol{r}_a = 0$ for $a > j$. The PLS weight vector $\boldsymbol{r}_a$ thus has to be orthogonal to all previous $x$-loadings $P_{a-1} = [\boldsymbol{p}_1, \ldots, \boldsymbol{p}_{a-1}]$. Consequently, $\boldsymbol{r}_a$ and $\boldsymbol{q}_a$ are computed as the first left and right singular vectors of $S_{xy}$ projected on a subspace orthogonal to $P_{a-1}$. This projection is performed by constructing an orthonormal base $\{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{a-1}\}$ of $\{\boldsymbol{p}_1, \ldots, \boldsymbol{p}_{a-1}\}$. Next, $S_{xy}^{a-1}$ is deflated:

$$S_{xy}^{a} = S_{xy}^{a-1} - \boldsymbol{v}_a (\boldsymbol{v}_a' S_{xy}^{a-1}) \tag{7}$$

and $\boldsymbol{r}_a$ and $\boldsymbol{q}_a$ are the first left and right singular vectors of $S_{xy}^a$. We start this iterative algorithm with $S_{xy} = S_{xy}^1$ and repeat this process until $k$ components are obtained. The choice of the number of components $k$ will be discussed in Section 4.

In the second stage of the algorithm, the responses are regressed onto these $k$ components. The formal regression model under consideration is thus:

$$\boldsymbol{y}_i = \boldsymbol{\alpha}_0 + \mathcal{A}_{q,k}' \tilde{\boldsymbol{t}}_i + \boldsymbol{f}_i \tag{8}$$

where $E(\boldsymbol{f}_i) = 0$ and $\mathrm{Cov}(\boldsymbol{f}_i) = \Sigma_f$. Multiple linear regression (MLR) [11] provides estimates:

$$\begin{aligned} \hat{\mathcal{A}}_{k,q} &= (S_t)^{-1} S_{ty} = (R'_{k,p} S_x R_{p,k})^{-1} R'_{k,p} S_{xy} && (9) \\ \hat{\boldsymbol{\alpha}}_0 &= \bar{\boldsymbol{y}} - \hat{\mathcal{A}}'_{q,k} \bar{\bar{\boldsymbol{t}}} && (10) \\ S_f &= S_y - \hat{\mathcal{A}}'_{q,k} S_t \hat{\mathcal{A}}_{k,q} && (11) \end{aligned}$$

where $S_y$ and $S_t$ stand for the empirical covariance matrix of the $y$- and $t$-variables. Note that here MLR denotes the classical least squares regression with multiple $x$-variables, and when $q > 1$, with multiple $y$-variables (also known as multivariate multiple linear regression). Because $\bar{\bar{\boldsymbol{t}}} = 0$, the intercept $\boldsymbol{\alpha}_0$ is thus estimated by $\bar{\boldsymbol{y}}$. By plugging in $\tilde{\boldsymbol{t}}_i = R'_{k,p}(\boldsymbol{x}_i - \bar{\boldsymbol{x}})$ in (3), we obtain estimates for the parameters in the original model (1), i.e.

$$\begin{aligned} \hat{\mathcal{B}}_{p,q} &= R_{p,k} \hat{\mathcal{A}}_{k,q} && (12) \\ \hat{\boldsymbol{\beta}}_0 &= \bar{\boldsymbol{y}} - \hat{\mathcal{B}}'_{q,p} \bar{\boldsymbol{x}}. && (13) \end{aligned}$$

Finally, also an estimate of $\Sigma_e$ is provided by rewriting $S_f$ in terms of the original parameters:

$$S_e = S_y - \hat{\mathcal{B}}' S_x \hat{\mathcal{B}}. \quad (14)$$

Note that for a univariate response variable ($q = 1$), the parameter estimate $\hat{\mathcal{B}}_{p,1}$ can be rewritten as the vector $\hat{\boldsymbol{\beta}}$, whereas the estimate of the error variance $S_e$ simplifies to $\hat{\sigma}_e^2 = s_e^2$.

## Example: Fish data

We will illustrate the SIMPLS algorithm on a low-dimensional example introduced in [14]. This data set contains $n = 45$ measurements of fish. For each fish, the fat concentration was measured, whereas the $x$-variables consist of highly multicollinear spectra at nine wavelengths. The goal of the analysis is to model the relation between the single response variable, fat concentration, and these spectra. The $x$-variables are shown in Figure 1. On this figure we have highlighted the observations which have the most outlying spectrum. It was reported by Naes [14] that observations 39 to 45 are outlying. We see that all of them, except observation 42, have indeed a spectrum which deviates from the majority.

[Figure 1 about here]

It has been shown in [5] and [6] that three components are sufficient to perform the PLS regression. So, if we carry out the SIMPLS algorithm with $k = 3$, we obtain the regression diagnostic plot shown in Figure 2(a). On the horizontal axis this plot displays the Mahalanobis distance of a data point in the $t$-space, which we therefore call its score distance $\mathrm{SD}_{i(k)}$. It is defined by

$$\mathrm{SD}_{i(k)}^2 = \tilde{\boldsymbol{t}}'_i S_t^{-1} \tilde{\boldsymbol{t}}_i. \quad (15)$$

Note that this distance $\mathrm{SD}^2_{i(k)}$ depends on $k$, because the scores $\tilde{\boldsymbol{t}}_i = \tilde{\boldsymbol{t}}_{i(k)}$ are obtained from a PLS model with $k$ components. On the vertical axis, the standardized concentration residuals $\frac{r_{i(k)}}{s_e}$ with $r_{i(k)} = y_i - \hat{\beta}_0 - \hat{\boldsymbol{\beta}}'\boldsymbol{x}_i$ are displayed. We define outlying data points in the $t$-space as those observations whose score distance exceeds the cutoff-value $\sqrt{\chi^2_{k,0.975}}$ (because the squared Mahalanobis distances of normally distributed scores are $\chi^2_k$-distributed). Regression outliers have an absolute standardized residual which exceeds $\sqrt{\chi^2_{1,0.975}} = 2.24$. The SIMPLS diagnostic plot suggests that observations 43, 44 and 45 are outlying in the $t$-space, whereas observations 1 and 43 can be classified as regression outliers. Their standardized residual is however not that large, so they are rather borderline cases.

[Figure 2 about here]

Figure 2(b) shows the robust diagnostic plot, based on the robust PLS method that will be described in Section 3. For a precise definition of this plot, we refer to Section 5. Here, we can clearly identify the observations with outlying spectrum (1, 12, 39, 40, 41, 43, 44 and 45). Moreover, the robust PLS method finds several regression outliers which can not be seen on the SIMPLS plot.

# 3 Robustified versions of the SIMPLS algorithm

## 3.1 Robust covariance estimation in high dimensions

The SIMPLS method does not detect the outliers because both stages in the algorithm are not resistant towards outlying observations. The scores $\tilde{\boldsymbol{t}}_i$ are calculated based on the sample cross-covariance matrix $S_{xy}$ between the $x$- and $y$-variables and the empirical covariance matrix $S_x$ of the $x$-variables, which are highly susceptible to outliers, as is the least squares regression performed in the second stage of the algorithm. This will also be clear from the simulation study in Section 6. In this section, we robustify the SIMPLS method by replacing the sample cross-covariance matrix $S_{xy}$ by a robust estimate of $\Sigma_{xy}$ and the empirical covariance matrix $S_x$ by a robust estimate of $\Sigma_x$, and by performing a robust regression method instead of MLR. Two variants of SIMPLS will be proposed, RSIMCD and RSIMPLS. These algorithms can be applied for one or several response variables ($q \geq 1$).

Our estimators are thus based on robust covariance matrices for high-dimensional data. For this, we will use the ROBPCA method which has recently been developed [8].

Suppose we want to estimate the center and scatter of $n$ observations $\boldsymbol{z}_i$ in $m$ dimensions, with $n < m$. Because we are dealing with data sets with a very large number of variables, we can not rely on a first class of well-known robust methods to estimate the covariance structure of a sample. If the dimension of the data were small ($m < n$), we could for example apply

the Minimum Covariance Determinant (MCD) estimator of location and scatter [18]. The principle of the MCD method is to minimize the determinant of the sample covariance matrix of $h$ observations with $\lceil n/2 \rceil < h < n$, for which a fast algorithm (FAST-MCD) exists [21]. The center of the $\boldsymbol{z}_i$ is then estimated by the mean $\bar{\boldsymbol{z}}_h$ and their scatter by the empirical covariance matrix $S_h$ of the optimal $h$-subset (multiplied with a consistency factor). To increase the finite-sample efficiency, a reweighting step can be added as well. An observation receives zero weight if its robust squared distance $(\boldsymbol{z}_i - \bar{\boldsymbol{z}}_h)' S_h^{-1} (\boldsymbol{z}_i - \bar{\boldsymbol{z}}_h)$ exceeds $\chi^2_{m,0.975}$. The reweighted MCD estimator is then defined as the classical mean and covariance matrix of those observations with weight equal to 1.

However, when $m > n$, the MCD estimator is not applicable anymore because the covariance matrix of $h < m$ data points is always singular. For such high-dimensional data sets, projection pursuit algorithms have been developed [12], [7]. ROBPCA combines the two approaches. Using projection pursuit ideas as in Donoho [3] and Stahel [23], it computes the outlyingness of every data point and then considers the empirical covariance matrix of the $h$ data points with smallest outlyingness. The data are then projected onto the subspace $K_0$ spanned by the $k_0 \ll m$ dominant eigenvectors of this covariance matrix. Next, the MCD method is applied to estimate the center and the scatter of the data in this low-dimensional subspace. Finally these estimates are backtransformed to the original space and a robust estimate of the center $\hat{\boldsymbol{\mu}}_z$ of $Z_{n,m}$ and of its scatter $\hat{\Sigma}_z$ are obtained. This scatter matrix can be decomposed as

$$\hat{\Sigma}_z = P^z L^z (P^z)' \tag{16}$$

with robust $Z$-eigenvectors $P^z_{m,k_0}$ and $Z$-eigenvalues $\mathrm{diag}(L_{k_0,k_0})$. Note that the diagonal matrix $L^z$ contains the $k_0$ largest eigenvalues of $\hat{\Sigma}_z$ in decreasing order. Then $Z$-scores $T^z$ can be obtained from $T^z = (Z - \mathbf{1}_n \hat{\boldsymbol{\mu}}_z') P^z$. For all details about ROBPCA, we refer to [8].

## 3.2 Robust PLS

### 3.2.1 Robust scores

To obtain robust scores we first apply ROBPCA on $Z_{n,m} = (X_{n,p}, Y_{n,q})$. This yields a robust estimate of the center of $Z$, $\hat{\boldsymbol{\mu}}_z = (\hat{\boldsymbol{\mu}}_x', \hat{\boldsymbol{\mu}}_y')'$, and an estimate of its shape, $\hat{\Sigma}_z$, which can be split into

$$\hat{\Sigma}_z = \begin{pmatrix} \hat{\Sigma}_x & \hat{\Sigma}_{xy} \\ \hat{\Sigma}_{yx} & \hat{\Sigma}_y \end{pmatrix}. \tag{17}$$

We estimate the cross-covariance matrix $\Sigma_{xy}$ by $\hat{\Sigma}_{xy}$ and compute the PLS weight vectors $\boldsymbol{r}_a$ as in the SIMPLS algorithm, but now starting with $\hat{\Sigma}_{xy}$ instead of $S_{xy}$. In analogy with (6) the $x$-loadings $\boldsymbol{p}_j$ are defined as $\boldsymbol{p}_j = (\boldsymbol{r}_j' \hat{\Sigma}_x \boldsymbol{r}_j)^{-1} \hat{\Sigma}_x \boldsymbol{r}_j$. Then the deflation of the scatter

matrix $\hat{\Sigma}_{xy}^{a}$ is performed as in SIMPLS. In each step, the robust scores are calculated as:

$$t_{ia} = \breve{\boldsymbol{x}}_i' \boldsymbol{r}_a = (\boldsymbol{x}_i - \hat{\boldsymbol{\mu}}_x)' \boldsymbol{r}_a \tag{18}$$

where $\breve{\boldsymbol{x}}_i$ are the robustly centered observations.

**Remark 1** When performing the ROBPCA method on $Z_{n,m}$, we need to determine $k_0$, which should be a good approximation of the dimension of the space spanned by the $x$- and $y$-variables. If $k$ is known, we set $k_0 = \min(k, 10) + q$. The number $k + q$ represents the sum of the number of $x$-loadings that gives a good approximation of the dimension of the $x$-variables, and the number of response variables. The maximal value $k_{\max} = 10$ is included to ensure a good efficiency of the FAST-MCD method in the last stage of ROBPCA, but may be increased if enough observations are available.

When analyzing a specific data set, $k_0$ could be chosen by looking at the eigenvalues of the empirical covariance matrix of the $h$ observations with the smallest outlyingness. By doing this, one should keep in mind that it is logical that $k_0$ is larger than the number of components $k$ that will be retained in the regression step.

**Remark 2** When $p + q < n$, we can directly compute the reweighted MCD-estimator on $Z_{n,m}$. The construction of the pairs of PLS weight vectors is then closely related to an inter-battery method of Tucker (1958). The influence functions of the resulting PLS weight vectors, which measure the infinitesimal effect of one outlier on the estimates, appear to be bounded [26]. This illustrates the robustness of this approach towards point contamination.

### 3.2.2 Robust regression

Once the scores are derived, a robust linear regression is performed. The regression model is the same as in (8), but now based on the robust scores $\boldsymbol{t}_i$:

$$\boldsymbol{y}_i = \boldsymbol{\alpha}_0 + \mathcal{A}_{q,k}' \boldsymbol{t}_i + \breve{\boldsymbol{f}}_i. \tag{19}$$

Note that when $q = 1$, well-known robust methods such as the LTS regression [18] could be used. This approach is followed in [9]. Here, we propose two methods that can be used for regression with one or multiple response variables. Throughout we only use the notation for the multivariate setting, but both approaches apply as well when $y_i$ is a scalar instead of a vector. The first multivariate regression that we discuss is the MCD regression method [22]. The second one uses additional information from the previous ROBPCA step, and hence will be called the ROBPCA regression.

**MCD regression**

The classical MLR estimates for the regression model presented in (19) can be written in terms of the covariance $\Sigma$ and the center $\boldsymbol{\mu}$ of the joint variables $(t, y)$:

$$\boldsymbol{\mu} = \begin{pmatrix} \boldsymbol{\mu}_t \\ \boldsymbol{\mu}_y \end{pmatrix}, \qquad \Sigma = \begin{pmatrix} \Sigma_t & \Sigma_{ty} \\ \Sigma_{yt} & \Sigma_y \end{pmatrix}. \tag{20}$$

If the center $\boldsymbol{\mu}$ is estimated by the sample mean $(\bar{\boldsymbol{t}}', \bar{\boldsymbol{y}}')'$ and the covariance $\Sigma$ by the sample covariance matrix of $(t, y)$, the classical estimates satisfy equations (9)-(11) if we replace $\bar{\bar{\boldsymbol{t}}}$ in (10) by $\bar{\boldsymbol{t}}$. Robust regression estimates are obtained by replacing the classical mean and covariance matrix of $(t, y)$ by the reweighted MCD estimates of center and scatter [22]. It is moreover recommended to reweigh these initial regression estimates in order to improve the finite-sample efficiency. Let $\boldsymbol{r}_{i(k)}$ be the residual of the $i$th observation based on the initial estimates that were calculated with $k$ components. If $\hat{\Sigma}_{\check{f}}$ is the initial estimate for the covariance matrix of the errors, then we define the robust distance of the residuals as:

$$\mathrm{RD}_{i(k)} = (\boldsymbol{r}'_{i(k)} \hat{\Sigma}^{-1}_{\check{f}} \boldsymbol{r}_{i(k)})^{1/2}. \tag{21}$$

The weights $c_{i(k)}$ are computed as

$$c_{i(k)} = I(\mathrm{RD}^2_{i(k)} \leq \chi^2_{q,0.975}) \tag{22}$$

with $I$ the indicator function. The final regression estimates are then calculated as in classical MLR, but only based on those observations with weight $c_{i(k)}$ equal to 1. The robust residual distances $\mathrm{RD}_{i(k)}$ are recomputed as in (21) and also the weights $c_{i(k)}$ are adapted.

Analogously to (12)-(14), robust parameters for the original model (1) are then given by:

$$\hat{\mathcal{B}}_{p,q} = R_{p,k} \hat{\mathcal{A}}_{k,q} \tag{23}$$

$$\hat{\boldsymbol{\beta}}_0 = \hat{\boldsymbol{\alpha}}_0 - \hat{\mathcal{B}}'_{q,p} \hat{\boldsymbol{\mu}}_x \tag{24}$$

$$\hat{\Sigma}_e = \hat{\Sigma}_{\check{f}}. \tag{25}$$

The resulting method is called RSIMCD.

**Remark 3** Both MCD and ROBPCA assume that the data set contains at least $h$ good observations. We will therefore use the same value for $h$ in the two steps of our algorithm, although this is not necessary. To perform MCD on $(t, y)$ it is required that $h > k + q$. With $k_{\max} = 10$ and $h > \lceil \frac{n}{2} \rceil$, this condition is certainly fulfilled if $\lceil \frac{n}{2} \rceil \geq 10 + q$. This is usually not a problem because $q$ is very small.

The value for $h$ influences the robustness of our estimates. It should be larger than $[\frac{n+k_0+1}{2}]$ in the ROBPCA step and larger than $[\frac{n+k+q+1}{2}]$ in the MCD regression [19]. In our Matlab implementation, we have therefore set $h = \max([\alpha n], [\frac{n+10+q+1}{2}])$, with $\alpha = 0.75$ as default value. It is possible to increase or decrease the value of $\alpha$ where $(1 - \alpha)$ represents the fraction of outliers the algorithm should be able to resist.

**ROBPCA regression**

The simulation study in Section 6 shows that RSIMCD is highly robust to many types of outliers. Its computation time is mainly determined by applying ROBPCA on the $(x, y)$-variables and MCD on the $(t, y)$-variables. Now, we introduce a second robust SIMPLS algorithm which avoids the computation of the MCD on $(t, y)$ by using additional information from the ROBPCA step.

The MCD regression method starts by applying the reweighted MCD estimator on $(t, y)$ to obtain robust estimates of their center $\boldsymbol{\mu}$ and scatter $\Sigma$. This reweighted MCD corresponds to the mean and the covariance matrix of those observations which are considered not to be outlying in the $(k + q)$-dimensional $(t, y)$ space.

To obtain the robust scores $\boldsymbol{t}_i$, we first applied ROBPCA to the $(x, y)$-variables, and obtained a $k_0$-dimensional subspace $K_0$ which represented these $(x, y)$-variables well. Because the scores were then constructed to summarize the most important information given in the $x$-variables, we might expect that outliers with respect to this $k_0$-dimensional subspace are often also outlying in the $(t, y)$ space. Hence, we will estimate the center $\boldsymbol{\mu}$ and the scatter $\Sigma$ of the $(t, y)$-variables as the weighted mean and covariance matrix of those $(\boldsymbol{t}_i, \boldsymbol{y}_i)$ whose corresponding $(\boldsymbol{x}_i, \boldsymbol{y}_i)$ are not outlying to $K_0$:

$$\hat{\boldsymbol{\mu}} = \begin{pmatrix} \hat{\boldsymbol{\mu}}_t \\ \hat{\boldsymbol{\mu}}_y \end{pmatrix} = \sum_{i=1}^{n} w_i \begin{pmatrix} \boldsymbol{t}_i \\ \boldsymbol{y}_i \end{pmatrix} / (\sum_{i=1}^{n} w_i) \tag{26}$$

$$\hat{\Sigma} = \begin{pmatrix} \hat{\Sigma}_t & \hat{\Sigma}_{ty} \\ \hat{\Sigma}_{yt} & \hat{\Sigma}_y \end{pmatrix} = \sum_{i=1}^{n} w_i \begin{pmatrix} \boldsymbol{t}_i \\ \boldsymbol{y}_i \end{pmatrix} \begin{pmatrix} \boldsymbol{t}_i' & \boldsymbol{y}_i' \end{pmatrix} / (\sum_{i=1}^{n} w_i - 1) \tag{27}$$

with $w_i = 1$ if observation $i$ is not identified as an outlier by applying ROBPCA on $(x, y)$, and $w_i = 0$ otherwise.

The question remains how these weights $w_i$ are determined. When we apply ROBPCA, we can identify two types of outliers: those who are outlying within $K_0$, and those who are lying far from $K_0$ (see [8] for a graphical sketch). The first type of outliers can be easily identified as those observations whose robust distance $D_{i(k_0)} = \sqrt{(\boldsymbol{t}_i^z)'(L^z)^{-1}\boldsymbol{t}_i^z}$ exceeds $\sqrt{\chi^2_{k_0,0.975}}$. Here $L^z$ is defined as in (16) with $Z = (X, Y)$.

To determine the second type of outliers, we consider for each data point its orthogonal distance $\mathrm{OD}_i = \|(\boldsymbol{z}_i - \hat{\boldsymbol{\mu}}_z) - P^z\boldsymbol{t}_i^z\|$ to the subspace $K_0$. The distribution of these orthogonal distances are difficult to determine exactly, but motivated by the central limit theorem, it appears that the squared orthogonal distances are roughly normally distributed. Hence, we estimate their center and variance with the univariate MCD, yielding $\hat{\mu}_{\mathrm{od}^2}$ and $\hat{\sigma}^2_{\mathrm{od}^2}$. We then set $w_i = 0$ if

$$\mathrm{OD}_i > \sqrt{\hat{\mu}_{\mathrm{od}^2} + \hat{\sigma}_{\mathrm{od}^2} z_{0.975}}, \tag{28}$$

with $z_{0.975} = \Phi^{-1}(0.975)$, the 97.5% quantile of the Gaussian distribution. Another approximation is explained in [8]. One can of course also plot the orthogonal distances to see whether some of them are much larger than the others. We recommend this last approach when interaction is possible in the analysis of a particular data set.

Having identified the observations with weight 1, we thus compute $\hat{\boldsymbol{\mu}}$ and $\hat{\Sigma}$ from (26) and (27). Then, we proceed as in the MCD regression method. We plug these estimates in (9) to (11), compute residual distances as in (21) and perform a reweighted MLR. This reweighting step has the advantage that it might again include observations with $w_i = 0$ which are not regression outliers. We will refer to this algorithm as the RSIMPLS method.

**Remark 4** Both proposed robust PLS algorithms have several equivariance properties. The ROBPCA method is orthogonally equivariant, which means that orthogonal transformations of the data (rotations, reflections) transform the loadings appropriately and leave the scores unchanged. Consequently, it can easily be derived that RSIMCD and RSIMPLS are equivariant for translations and orthogonal transformations in $x$ and $y$. More precisely, let $\boldsymbol{u} \in \mathbb{R}^p$, $\boldsymbol{v} \in \mathbb{R}^q$, $C$ any $p$-dimensional orthogonal matrix and $D$ any $q$-dimensional orthogonal matrix. If $(\hat{\boldsymbol{\beta}}_0, \hat{\mathcal{B}})$ denotes the estimates by running RSIMCD or RSIMPLS on the original data $(\boldsymbol{x}_i, \boldsymbol{y}_i)$, it holds that:

$$\begin{aligned} \hat{\mathcal{B}}(C\boldsymbol{x}_i + \boldsymbol{u}, D\boldsymbol{y}_i + \boldsymbol{v}) &= C\hat{\mathcal{B}}D' \\ \hat{\boldsymbol{\beta}}_0(C\boldsymbol{x}_i + \boldsymbol{u}, D\boldsymbol{y}_i + \boldsymbol{v}) &= D\hat{\boldsymbol{\beta}}_0 + \boldsymbol{v} - D\hat{\mathcal{B}}'C'\boldsymbol{u}. \end{aligned}$$

**Remark 5** Instead of using hard-rejection rules for defining outliers, we could also apply continuous weights between 0 and 1. But this introduces additional choices (weight functions, cut-off values) and the results are not necessarily improved. For a more detailed discussion, see [9].

**Remark 6** We have also developed a robust PLS1 algorithm following the approach developed in [5] but by replacing the Stahel-Donoho estimator with the ROBPCA covariance matrix. However, the results were in general not better than those obtained with RSIMPLS or RSIMCD. Hence we prefer RSIMPLS because it can also be applied when $q \geq 1$.

## 3.3 Comparison

In Section 6, we present the results of a thorough simulation study where we compare the performance of these two different robust methods. These simulations indicate that RSIMCD and RSIMPLS are very comparable, but if we compare the mean CPU-time in seconds computed on a Pentium IV with 1.60 GHz (see Table 1) over five runs for different situations, we see that RSIMPLS is roughly twice as fast as RSIMCD. This is explained by

the fact that we apply FAST-MCD in the second stage of RSIMCD. Hence, in the following sections, we will mainly concentrate on RSIMPLS.

[Table 1 about here]

# 4 Model Calibration and Validation

## 4.1 Selecting the number of components

A very important issue in building the PLS model is the choice of the optimal number of components $k_{\text{opt}}$. The most common methods use a test set or cross-validation (CV). A test set should be independent from the training set which is used to estimate the regression parameters in the model, but should still be representative of the population. Let $n_T$ denote the total number of observations in the test set. For each $k$, the root mean squared error $\text{RMSE}_k$ for this test set can be computed as:

$$\text{RMSE}_k = \sqrt{\frac{1}{n_T q} \sum_{i=1}^{n_T} \sum_{j=1}^{q} (y_{ij} - \hat{y}_{ij(k)})^2}. \tag{29}$$

Here, the predicted values $\hat{y}_{ij(k)}$ of observation $i$ in the test set are based on the parameter estimates that are obtained from the training set using a PLS method with $k$ components. One then chooses $k_{\text{opt}}$ as the $k$-value which gives the smallest or a sufficiently small value for $\text{RMSE}_k$.

This $\text{RMSE}_k$ statistic can however attain unreliable values if the test set contains outliers, even if the fitted values are based on a robust PLS algorithm. Such outlying observations increase the $\text{RMSE}_k$ because they fit the model badly. Consequently, our decision about $k_{\text{opt}}$ might be wrong. Therefore, we propose to remove the outliers from the test set before computing $\text{RMSE}_k$. Formally, let $\boldsymbol{r}_{i(k)}$ be the residual for the $i$th observation in the test set and $c_{i(k)} = I(\boldsymbol{r}'_{i(k)} \hat{\Sigma}_e^{-1} \boldsymbol{r}_{i(k)} < \chi^2_{q,0.975})$. The weight $c_{i(k)}$ thus tells whether or not the $i$th observation is outlying with respect to the PLS model with $k$ components. Then, we select the test data points which are not outlying in any of the models by computing $c_i = \min_k c_{i(k)}$. Let $G_t$ denote the set of points for which $c_i = 1$, and let $n_t$ be its size: $|G_t| = n_t$. Finally, for each $k$, we define the robust $\text{RMSE}_k$ value as:

$$\text{R-RMSE}_k = \sqrt{\frac{1}{n_t q} \sum_{i \in G_t} \sum_{j=1}^{q} (y_{ij} - \hat{y}_{ij(k)})^2}. \tag{30}$$

This approach is fast because we only need to run the PLS algorithm once for each $k$. But an independent test set is only exceptionally available. This can be solved by splitting the

original data into a training and a test set. However, the data sets we consider generally have a limited number of observations and it is preferable that the number of observations in the training step is at least 6 to 10 times the number of variables. That is why we concentrate on the cross-validated $\text{RMSE}_k$, which will be denoted by $\text{RMSECV}_k$ [24], [9]. Usually, the 1-fold or leave-one-sample out statistic is obtained as in (29) but now the index $i$ runs over the set of all the observations, and the predicted values $\hat{y}_{ij(k)}$ are based on the PLS estimates obtained by removing the $i$th observation from the data set. The optimal number of components is then again taken as the value $k_{\text{opt}}$ for which $\text{RMSECV}_k$ is minimal or sufficiently small.

However, as we have argued for $\text{RMSE}_k$, also the $\text{RMSECV}_k$ statistic is vulnerable to outliers, so we also remove an outlying observation. Let $\hat{\boldsymbol{\mathcal{B}}}_{-i}$, $\hat{\boldsymbol{\beta}}_{0,-i}$ and $\hat{\Sigma}_{e,-i}$ denote the parameter estimates based on the data set without the $i$th observation, $\boldsymbol{r}_{-i(k)} = \boldsymbol{y}_i - \hat{\boldsymbol{\beta}}_{0,-i} - \hat{\boldsymbol{\mathcal{B}}}'_{-i}\boldsymbol{x}_i$ the $i$th cross-validated residual and $\text{RD}^2_{-i(k)} = \boldsymbol{r}'_{-i(k)}\hat{\Sigma}^{-1}_{e,-i}\boldsymbol{r}_{-i(k)}$ the squared cross-validated residual distance as in (21). Then analogously to (22) the cross-validated weight assigned to the $i$th observation is defined as

$$c_{-i(k)} = I(\text{RD}^2_{-i(k)} < \chi^2_{q,0.975}).$$

If $c_{-i(k)} = 0$, observation $(\boldsymbol{x}_i, \boldsymbol{y}_i)$ is recognized as a regression outlier in the PLS model with $k$ components. Several PLS models are constructed for $k = 1, \ldots, k_{\text{tot}}$ components with $k_{\text{tot}}$ the total or maximal number of components under consideration. Because we want to compare these $k_{\text{tot}}$ different models, we should evaluate their predictive power on the same set of observations. Hence, we could eliminate those observations which are outlying in any of the models by defining for each observation

$$c_{-i} = \min_k c_{-i(k)}. \tag{31}$$

Let $G_c$ denote the subset of observations for which $c_{-i} = 1$ with $|G_c| = n_c$, then an observation belongs to the set $G_c$ when it is observed as a regular sample in each of the $k_{\text{tot}}$ PLS models. For each $k$, we then define the robust $\text{RMSECV}_k$ value as:

$$\text{R-RMSECV}_k = \sqrt{\frac{1}{n_c q}\sum_{i\in G_c}\sum_{j=1}^{q}(y_{ij} - \hat{y}_{-ij(k)})^2} \tag{32}$$

with the cross-validated fitted value $\hat{\boldsymbol{y}}_{-i(k)} = \hat{\boldsymbol{\beta}}_{0,-i} + \hat{\boldsymbol{\mathcal{B}}}'_{-i}\boldsymbol{x}_i$. This approach has the advantage that any suspicious observation is discarded in the $\text{R-RMSECV}_k$ statistic. It is also followed in [17] to construct robust stepwise regression by means of a bounded-influence estimator for prediction. On the other hand, when the number of observations is small, increasing $k_{\text{tot}}$ can lead to sets $G_c$ for which the number of observations $n_c$ is small compared to the total

number of observations $n$. Let us e.g. consider the Fish data set. When we choose $k_{\text{tot}} = 5$, $n_c = 30$ (out of $n = 45$), but with $k_{\text{tot}} = 9$, $n_c = 23$, which is only half of the observations. To avoid such small calibration sets, we alternatively define

$$c_{-i} = \underset{k}{\text{median}}\, c_{-i(k)}. \tag{33}$$

With this definition, only those data points that are outlying with respect to most of the PLS models under consideration are removed. Note that when $k$ is even, we take the low-median of the $c_{-i,k}$ in order to obtain a weight $c_{-i}$ which is always exactly zero or one. For the Fish data set, we then obtain $n_c = 34$ when $k_{\text{tot}} = 9$. Figure 3 displays the R-RMSECV curve for the Fish data with the two different weight functions. We see that both curves do not differ very much, and they both indicate to select three components in the regression model, which is similar to the conclusion of the analysis in [5] and [6]. We also superimposed the R-RMSECV curves for the SIMPLS algorithm based on the same subsets $G_c$ of observations as RSIMPLS and again conclude that three components are sufficient.

[Figure 3 about here]

The drawback of cross-validation is its computation time, because for each $k$ the PLS algorithm has to be run $n$ times. To speed up the computations, we therefore fix $k_0$, the number of principal components that are selected in ROBPCA to obtain the robust center $\hat{\boldsymbol{\mu}}_z$ and scatter $\hat{\Sigma}_z$ in (17). If we then increase $k$, we only need to compute one extra component by deflating $S^a_{xy}$ once more as explained in (7). Fixing $k_0$ has the additional advantage that the weights $w_i$ that are needed in the ROBPCA regression do not change with $k$. To determine $k_0$, we first compute $k_{\text{tot}} \leq \min\{p, k_{\max} = 10\}$ that will be used as a maximal value for $k$ in the regression. The total number of parameters to be estimated equals $kq$ for the slope matrix $\hat{\mathcal{A}}$, $q$ for the intercept $\hat{\boldsymbol{\alpha}}_0$ and $q(q-1)/2$ (or 1 when $q = 1$) for $\hat{\Sigma}_e$. To avoid overfitting, we then require that

$$k_{\text{tot}}q + q + \frac{q(q-1)}{2} < h \tag{34}$$

where $h$ stands for the size of the subset that is used in the ROBPCA, or MCD regression, and which should be a lower bound for the number of regular observations out of $n$. Note that if $q = 1$, we have only one scale estimate $\hat{\sigma}_e$, hence we require that

$$k_{\text{tot}} + 2 < h.$$

Having determined $k_{\text{tot}}$, we then set $k_0 = k_{\text{tot}} + q$. For the Fish data, this implies that $k_{\text{tot}} = 9$ and $k_0 = 10$.

**Remark 7** In [9], also a robust $R_k^2$ value is defined to determine the optimal number of components. For $q = 1$, it is defined by

$$R_k^2 = 1 - \frac{\sum_{i \in G_t} r_{i(k)}^2}{\sum_{i \in G_t} (y_i - \bar{y}_c)^2}$$

with $\bar{y}_c = \sum_{i \in G_t} y_i / n_t$ and $G_t$ is defined as in (30) with the test set being equal to the full data set. In the multivariate case ($q > 1$), this is generalized to:

$$R_k^2 = 1 - \frac{\sum_{i \in G_t} \sum_{j=1}^{q} r_{ij(k)}^2}{\sum_{i \in G_t} \sum_{j=1}^{q} (y_{ij} - \bar{y}_j)^2}$$

where $\bar{y}_j = \sum_{i \in G_t} y_{ij} / n_t$. The optimal number of components $k_{\text{opt}}$ is then chosen as the smallest value $k$ for which $R_k^2$ attains e.g. 80% or the $R_k^2$ curve becomes nearly flat. This approach is fast because it avoids cross-validation, but merely measures the variance of the residuals instead of the prediction error.

## 4.2 Estimation of Prediction Error

Once the optimal number of components $k_{\text{opt}}$ is chosen, we can validate our model by estimating the prediction error. We therefore define R-RMSEP$_{k_{\text{opt}}}$ (Robust Root Mean Squared Error of Prediction), as in (32):

$$\text{R-RMSEP}_{k_{\text{opt}}} = \sqrt{\frac{1}{n_p q} \sum_{i \in G_p} \sum_{j=1}^{q} (y_{ij} - \hat{y}_{-ij(k_{\text{opt}})})^2} \tag{35}$$

where $G_p$ is now the subset of observations with non-zero weight $c_{-i(k_{\text{opt}})}$ in the PLS model with $k_{\text{opt}}$ components, and $|G_p| = n_p$. The fitted values are obtained with $k_0 = k_{\text{opt}} + q$ in ROBPCA. Note that using this definition, we include all the regular observations for the model with $k_{\text{opt}}$ components, which is more precise than the set $G_c$ that is used in (32) and which depends on $k_{\text{tot}}$. Hence, in general R-RMSEP$_{k_{\text{opt}}}$ will be different from R-RMSECV$_{k_{\text{opt}}}$.

For the Fish data set and RSIMPLS, we obtain R-RMSEP$_3$ = 0.51 based on $n_p = 33$ observations. If we perform SIMPLS, and calculate the R-RMSEP$_3$ value on the same set of observations, we obtain 0.82, hence the robust fit yields a smaller prediction error. To test whether this difference is significant, we applied the following bootstrap procedure. We have drawn 150 bootstrap samples of size $n_p = 33$ from the $(\boldsymbol{x}_i, \boldsymbol{y}_i)$ with $c_{-i(3)} = 1$. For each bootstrap sample we have refitted the model with three components, and we have computed the R-RMSEP$_3$ as in (35) with the set $G_p$ being fixed during all the computations. The standard deviation of these 150 R-RMSEP values was equal to $\hat{\sigma}_{\text{R-RMSEP}} = 0.09$ and can be used as an approximation to the true standard deviation of the R-RMSEP statistic. Because $0.82 > 0.51 + 2.5 * 0.09 = 0.735$ we conclude that the R-RMSEP$_3$ based on RSIMPLS is significantly different from R-RMSEP$_3$ obtained with SIMPLS (at the 1% level).

# 5 Outlier Detection

## 5.1 Regression diagnostic plot

To identify the outlying observations, we will first construct a regression diagnostic plot as in [20], [22] and [9]. Its goal is to identify outlying observations with respect to the regression model (19). In our two robust PLS methods, we perform a regression of the $q$-dimensional $\boldsymbol{y}_i$ on the $k$-dimensional $\boldsymbol{t}_i = \boldsymbol{t}_{i(k)}$ assuming $k = k_{\text{opt}}$. We can distinguish three types of outliers. These different types are represented in Figure 4(a) for the case of simple regression, so when $q = 1$ and $k = 1$. Good leverage points lie in the direction of the fitted line or subspace, but have outlying $t$-values. This is also the case for bad leverage points, that moreover do not fit the model well. Vertical outliers are only outlying in the $y$-space. The latter two types of outliers are known to be very influential for the classical least squares regression fit, because they cause the slope to be tilted in order to accommodate the outliers.

To measure the outlyingness of a point in the $t$-space, we consider its robustified mahalanobis distance, which we now call the score distance $\text{SD}_{i(k)}$, defined by

$$\text{SD}^2_{i(k)} \quad = \quad (\boldsymbol{t}_i - \hat{\boldsymbol{\mu}}_t)' \hat{\Sigma}_t^{-1} (\boldsymbol{t}_i - \hat{\boldsymbol{\mu}}_t)$$

where $\hat{\boldsymbol{\mu}}_t$ and $\hat{\Sigma}_t$ are derived in the regression step, see e.g. (26) and (27). When we perform SIMPLS, this score distance reduces to (15) because $\hat{\boldsymbol{\mu}}_{\tilde{t}} = \boldsymbol{0}$. This score distance is put on the horizontal axis of the regression diagnostic plot and exposes the good and the bad leverage points. By analogy with [20], [22] and [9], leverage points are those observations whose score distance exceeds the cut-off value $\sqrt{\chi^2_{k,0.975}}$. On the vertical axis, we put the residual distance $\text{RD}_{i,k}$:

$$\text{RD}^2_{i(k)} = \boldsymbol{r}'_{i(k)} \hat{\Sigma}_e^{-1} \boldsymbol{r}_{i(k)}$$

with $\boldsymbol{r}_{i(k)} = \boldsymbol{y}_i - \hat{\boldsymbol{\beta}}_0 - \hat{\mathcal{B}}' \boldsymbol{x}_i$ being the residual of the $i$th observation. For univariate response variables, this residual distance simplifies to the standardized residual $\text{RD}_{i(k)} = \frac{r_{i(k)}}{\hat{\sigma}_e}$. Vertical outliers and bad leverage points are now observations whose residual distance exceeds $\sqrt{\chi^2_{q,0.975}}$.

[Figure 4 about here]

If the regression parameters are well estimated, i.e. if they are not influenced by the outliers, this diagnostic plot should thus look as in Figure 4(b) for $q = 1$, and as in Figure 4(c) for $q > 1$. Let us look again at Figure 2(b) which shows the regression diagnostic plot of the Fish data with RSIMPLS. On this plot we see six clear bad leverage points $(1, 12, 41, 43, 44, 45)$, two vertical outliers $(3, 10)$, two good leverage points $(39, 40)$ and three borderline cases. The diagnostic plot from SIMPLS is however not very informative. Some bad leverage points $(44, 45)$ are converted into good leverage points which illustrates that the least squares regression is tilted to accommodate the outliers.

### 5.2 Score diagnostic plot

Next, similarly as in [10] and [8], we can also classify the observations with respect to the PCA model (2). This yields the score diagnostic plot. On the horizontal axis, we place again for each observation its score distance $\mathrm{SD}_{i(k)}$. On the vertical axis, we put the orthogonal distance of an observation to the $t$-space:

$$\mathrm{OD}_{i(k)} = \|\breve{\boldsymbol{x}}_i - P_{p,k}\boldsymbol{t}_i\|.$$

This allows us again to identify three types of outliers. Bad PCA-leverage points have outlying $\mathrm{SD}_{i(k)}$ and $\mathrm{OD}_{i(k)}$, good PCA-leverage points have only outlying $\mathrm{SD}_{i(k)}$, whereas orthogonal outliers have only outlying $\mathrm{OD}_{i(k)}$. The latter ones are not yet visible on the regression diagnostic plot. They have the property that they lie far from the $t$-space, but they become regular observations after projection in the $t$-space. Hence, they will not badly influence the computation of the regression parameters, but they might influence the loadings.

For the Fish data set this diagnostic plot is presented in Figure 5(a) for SIMPLS and in Figure 5(b) for RSIMPLS. The horizontal line is computed as in (28). The outliers detected in the regression diagnostic plot (Figure 2(b)) are all recognized as leverage points in this score diagnostic plot. Furthermore we detect observation 42 as an orthogonal outlier. We also detect two other orthogonal outliers $(10, 28)$. For SIMPLS, this score diagnostic plot (Figure 5(a)) also discovers sample 42 as an orthogonal outlier, but observations 43–45 are classified as good leverage points.

[Figure 5 about here]

Note that the regression and the score diagnostic plot can also be combined into one three-dimensional figure exposing $(\mathrm{SD}_{i(k)}, \mathrm{OD}_{i(k)}, \mathrm{RD}_{i(k)})$, see also [9].

## 6 Simulation Study

To compare the different algorithms, we have performed several simulations with low- and high-dimensional data sets. For each situation, we generated 1000 data sets. First, we consider the case without contamination. The data sets were then generated according to the bilinear model (2) and (3), with:

$$\begin{aligned} T &\sim N_k(\mathbf{0}_k, \Sigma_t), \text{ with } k < p, \\ X &= TI_{k,p} + N_p(\mathbf{0}_p, 0.1I_p) \\ Y &= T\mathcal{A} + N_q(\mathbf{0}_q, I_q), \text{ with } \mathcal{A} \sim N_q(\mathbf{0}_q, I_q). \end{aligned}$$

Here, $(I_{k,p})_{i,j} = 1$ for $i = j$ and 0 elsewhere. These simulation settings imply that $k_{\text{opt}} = k$. Next, we introduced different types of outliers by randomly replacing $n\epsilon$ of the $n$ observations with $\epsilon = 10\%$. The conclusions obtained for $\epsilon = 20\%$ contamination were similar to those for $\epsilon = 10\%$ and are therefore not included. If $T_\epsilon$, $X_\epsilon$ and $Y_\epsilon$ denote the contaminated data, the bad leverage regression points were constructed as:

$$T_\epsilon \sim N_k(\mathbf{10}_k, \Sigma_t) \tag{36}$$

$$X_\epsilon = T_\epsilon I_{k,p} + N_p(\mathbf{0}_p, 0.1 I_p) \tag{37}$$

whereas the $y$-variables were not changed. The vertical outliers were generated with the uncontaminated $t$-variables, but adjusted $y$-variables:

$$Y_\epsilon = T\mathcal{A}_{k,q} + N_q(\mathbf{10}_q, 0.1 I_q). \tag{38}$$

Finally, orthogonal outliers were constructed by putting

$$X_\epsilon = T I_{k,p} + N_p((\mathbf{0}_k, \mathbf{10}_{p-k}), 0.1 I_p) \tag{39}$$

and taking unadjusted $y$-variables.

In Table 2, we have listed the different choices for $n$, $p$, $q$, $k$ and $\Sigma_t$. In every simulation setup, we calculated the Mean Squared Error (MSE) of the slope matrix $\hat{\mathcal{B}}$, of the intercept $\hat{\boldsymbol{\beta}}_0$ and of the covariance matrix of the residuals $\hat{\Sigma}_e$ with SIMPLS, RSIMCD and RSIMPLS using $k$ components. We also computed the results for the PLS (NIPALS) algorithm, but these were quite similar to the results from SIMPLS and are therefore not included. If $q = 1$, we included the mean angle (denoted by mean(angle)) between the estimated slope and the true slope in the simulation results. Note that here, we mainly focus on the parameter estimates. In the simulation study in [4] we concentrate on the predictive performance of RSIMPLS for varying values of $k$.

[Table 2 about here]

[Table 3-5 about here]

## Discussion of the results

Tables 3-5 summarize the results of the different simulations. When no contamination is present, all the estimators perform well. SIMPLS yields the lowest MSE for the slope, except for $q = 1$ and high-dimensional regressors (Table 4). Here, RSIMPLS and RSIMCD surprisingly even give better results.

When the data set is contaminated, SIMPLS clearly breaks down which can be seen from all the MSE's which rise considerably. The bad leverage points are very damaging for all

the parameters in the model, whereas the intercept is mostly influenced by vertical outliers. The orthogonal outliers are mostly influential at the low-dimensional data sets.

In contrast with SIMPLS, the values for the robust algorithms do not change very much. In almost every setting, the differences between RSIMCD and RSIMPLS are very small. Both robust PLS methods are thus comparable, but as mentioned in Section 3.3 we prefer RSIMPLS because it is computationally more attractive than RSIMCD.

# 7 Example : Biscuit Dough Data

Finally, we apply RSIMPLS on the well-known high-dimensional Biscuit dough data [15]. The data originally contain four response variables, namely the concentration of fat, flour, sucrose and water of 40 biscuit dough samples. In our analysis, we have removed the variable fat because it is not very highly correlated with the other constituents, and it has a higher variance. Because the three remaining response variables are highly correlated (see [9] for some figures) and have variances of the same order, this data set seems appropriate for a multivariate analysis. The aim is to predict these three biscuit constituents ($q = 3$) based on the 40 NIR spectra with measurements every two nanometers, from 1200nm up to 2400nm. We have done the same preprocessing as suggested in [7] which results in a data set of NIR spectra in 600 dimensions. Observation 23 is known to be an outlier, but we will still consider the data set with all 40 observations.

To decide on the numbers of components $k_{\text{opt}}$ we have drawn the R-RMSECV curve with the median weight defined in (33) and with $k_{\text{tot}} = 7$ derived from (34). This yields $n_c = 25$. Figure 6 suggests to take $k_{\text{opt}} = 3$. The R-RMSECV curve for SIMPLS (based on the same $G_c$ as for RSIMPLS) is superimposed and reveals higher errors, but also suggests three components.

[Figure 6 about here]

We then performed RSIMPLS with $k_{\text{opt}} = 3$ and $k_0 = 6$, and obtained the robust diagnostic plot in Figure 7(b). Observation 21 stands out as a clear outlier with a very large robust residual distance around 60. Observation 23 is also recognized as a bad leverage point having the largest score distance. Further, we distinguish three bad leverage points $(7, 20, 24)$ with merely large score distances, and one vertical outlier (22) with a somewhat larger residual distance. There are also some borderline cases $(20, 33)$. With SIMPLS, we obtain the regression diagnostic plot in Figure 7(a). The three most extreme outliers $(21, 23, 7)$ seen from the robust analysis, are still detected, but their distances have changed enormously. Observation 21 now has a residual distance $\text{RD}_{21(3)} = 5.91$ and the score distance $\text{SD}_{23(3)} = 4.23$. Observation 23 is almost turned into a good leverage point, whereas case 7 is a

boundary case because its residual distance is only 3.71, which does not lie very far from $\sqrt{\chi^2_{3,0.975}} = 3.06$.

[Figure 7 about here]

The RSIMPLS score diagnostic plot is shown in Figure 8(b). Observations $7, 20, 21, 23$ and 24 are detected as bad PCA-leverage points. The score diagnostic plot for SIMPLS in Figure 8(b) only indicates 23 as a good PCA-leverage point.

[Figure 8 about here]

The robust prediction error R-RMSEP$_3 = 0.53$. If we compute R-RMSEP$_3$ with the fitted values obtained with SIMPLS and $G_p$ as in RSIMPLS, we obtain 0.70 for the prediction error. This shows that RSIMPLS yields a lower prediction error than SIMPLS, evaluated at the same subset of observations. To know whether this difference is significant, we applied the same bootstrap procedure as explained in Section 4.2, from which we derived the standard deviation $\hat{\sigma}_{\text{R-RMSEP}} = 0.12$. This yield approximately a significant difference at the 15% level.

To finish this example, we illustrate that for this data set, it is worthwhile to consider the multivariate regression model where the three $y$-variables are simultaneously modelled instead of performing three univariate calibrations. First, we computed the univariate prediction errors based on the multivariate estimates. So we computed R-RMSEP$_3$ for each response variable separately ($j = 1, \ldots, 3$):

$$\text{R-RMSEP}_3 = \sqrt{\frac{1}{n_p} \sum_{i \in G_p} (y_{ij} - \hat{y}_{-ij(3)})^2}$$

where $\hat{y}_{-ij(3)}$ are the fitted values from the multivariate regression and $G_p$ is the subset of observations retained in the multivariate regression. We obtained R-RMSEP(flour) = 0.37, R-RMSEP(sucrose) = 0.82 and R-RMSEP(water) = 0.19. Then, we have applied RSIMPLS for the three concentrations separately. It turned out that three components were satisfactory for every response. These three univariate regressions resulted in R-RMSEP(flour) = 0.40, R-RMSEP(sucrose) = 0.95 and R-RMSEP(water) = 0.18. Also these latter prediction errors are based on the same subset $G_p$ from the multivariate approach. For flour and sucrose we thus obtain a higher prediction accuracy with the multivariate regression, whereas only water is slightly better fitted by its own model.

# 8 Conclusion

In this paper we have proposed two new robust PLSR algorithms based on the SIMPLS algorithm. RSIMCD and RSIMPLS can be applied to low- and high-dimensional regressor

variables, and to one or multiple response variables. First, robust scores are constructed, and then the analysis is followed by a robust regression step. Simulations have shown that they are resistant towards many types of contamination, whereas their performance is also good at uncontaminated data sets. We recommend RSIMPLS because it is roughly twice as fast as RSIMCD. A Matlab implementation of RSIMPLS is available at the web site `www.wis.kuleuven.ac.be/stat/robust.html` as part of the Matlab toolbox for Robust Calibration [27].

We have also proposed robust RMSECV curves to select the number of components, and a robust estimate of the prediction error. Diagnostic plots are introduced to discover the different types of outliers in the data and are illustrated on some real data sets. Also the advantage of the multivariate approach has been illustrated.

In [4], a comparative study is made between RSIMPLS and RPCR with emphasis on the predictive ability and the goodness-of-fit of these methods when varying the number of components $k$. Currently, we are developing faster algorithms to compute the R-RMSECV values to allow fast and robust model selection in multivariate calibration.

# References

[1] Cummins DJ, Andrews CW. Iteratively reweighted partial least squares: a performance analysis by Monte Carlo simulation. *J. Chemometrics* 1995; **9**:489–507.

[2] de Jong S. SIMPLS: an alternative approach to partial least squares regression. *Chemometrics Intell. Lab. Syst.* 1993; **18**:251–263.

[3] Donoho DL. *Breakdown Properties of Multivariate Location Estimators*, Ph.D. Qualifying paper, Harvard University, 1982.

[4] Engelen S, Hubert M, Vanden Branden K, Verboven S. *Robust PCR and robust PLS: a comparative study*; To appear in *Theory and Applications of Recent Robust Methods,* edited by M. Hubert, G. Pison, A. Struyf and S. Van Aelst, Series: Statistics for Industry and Technology, Birkhauser, Basel. Available at `http://www.wis.kuleuven.ac.be/stat/robust.html`.

[5] Gil JA, Romera R. On robust partial least squares (PLS) methods. *J. Chemometrics* 1998; **12**:365–378.

[6] Hardy AJ, MacLaurin P, Haswell SJ, de Jong S, Vandeginste BG. Double-case diagnostic for outliers identification. *Chemometrics Intell. Lab. Syst.* 1996; **34**:117–129.

[7] 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.

[8] Hubert M, Rousseeuw PJ, Vanden Branden K. ROBPCA: a new approach to robust principal component analysis; submitted to *Technometrics*. Under revision. Available at `http://www.wis.kuleuven.ac.be/stat`.

[9] Hubert M, Verboven S. A robust PCR method for high-dimensional regressors. *J. Chemometrics* 2003; **17**:438–452.

[10] Hubert M, Rousseeuw PJ, Verboven S. Robust PCA for high-dimensional data *In: Dutter R, Filzmoser P, Gather U, Rousseeuw PJ (eds.)*, *Developments in Robust Statistics*, 2003; Physika Verlag, Heidelberg, 169–179.

[11] Johnson R, Wichern D. *Applied Multivariate Statistical Analysis* (4th edn). Prenctice Hall: New Jersey, 1998.

[12] Li G, Chen Z. Projection-pursuit approach to robust dispersion and principal components: primary theory and Monte-Carlo. *J. Am. Statist. Assoc.* 1985; **80**:759–766.

[13] Martens H, Naes T. *Multivariate Calibration*. Wiley: Chichester, UK, 1998.

[14] Naes T. Multivariate calibration when the error covariance matrix is structured. *Technometrics* 1985; **27**:301–311.

[15] Osborne BG, Fearn T, Miller AR, Douglas S. Application of near infrared reglectance spectroscopy to the compositional analysis of biscuits and biscuit dough. *J. Scient. Food Agric.* 1984; **35**:99–105.

[16] Pell RJ. Multiple outlier detection for multivariate calibration using robust statistical techniques. *Chemometrics Intell. Lab. Syst.* 2000; **52**:87–104.

[17] Ronchetti E, Field C, Blanchard W. Robust linear model selection by cross-validation. *J. Am. Statist. Assoc.* 1997; **92**:1017–1023.

[18] Rousseeuw PJ. Least median of squares regression. *J. Am. Statist. Assoc.* 1984; **79**:871–880.

[19] Rousseeuw PJ, Leroy AM. *Robust Regression and Outlier Detection*. Wiley: New York, 1987.

[20] Rousseeuw PJ, van Zomeren BC. Unmasking multivariate outliers and leverage points. *J. Am. Statist. Assoc.* 1990; **85**:633-651.

[21] Rousseeuw PJ, Van Driessen K. A fast algorithm for the minimum covariance determinant estimator. *Technometrics* 1999; **41**:212–223.

[22] Rousseeuw PJ, Van Aelst S, Van Driessen, K, Agulló J. Robust multivariate regression 2002; submitted to *Technometrics*. Under revision. Available at `http://win-www.uia.ac.be/u/statis`.

[23] Stahel WA. *Robust Estimation: Infinitesimal Optimality and Covariance Matrix Estimators*, Ph.D. thesis, ETH, Zürich, 1981.

[24] Tenenhaus M. *La Régression PLS: Théorie et Pratique.* Éditions Technip: Paris, 1998.

[25] Tucker LR. An inter-battery method of factor analysis. *Psychometrika* 1958; **23**: 111–136.

[26] Vanden Branden K, Hubert M. The influence function of the classical and robust PLS weight vectors. Submitted, 2003. Available at `http://www.wis.kuleuven.ac.be/stat/robust.html`.

[27] Verboven S, Hubert M. A Matlab toolbox for robust calibration. In preparation, 2003.

[28] Wakeling IN, Macfie HJH. A robust PLS procedure. *J. Chemometrics* 1992; **6**:18 9–198.

[29] Wold H. Estimation of principal components and related models by iterative least squares. *Multivariate Analysis*, Academic Press, New York, 1966, 391–420.

[30] Wold H. Soft modelling by latent variables: the non-linear iterative partial least squares (NIPALS) approach. *Perspectives in Probability and Statistics (papers in honour of M.S. Bartlett on the occasion of his 65th birthday)* 1975; 117-142. Applied Probability Trust, Univ. Sheffield, Sheffield.

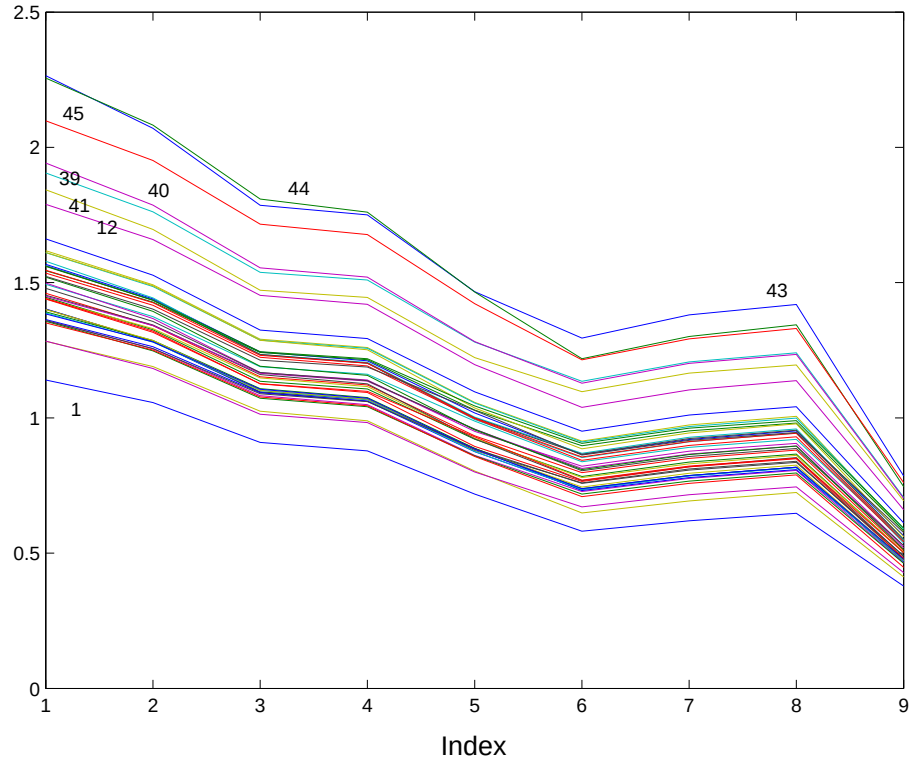
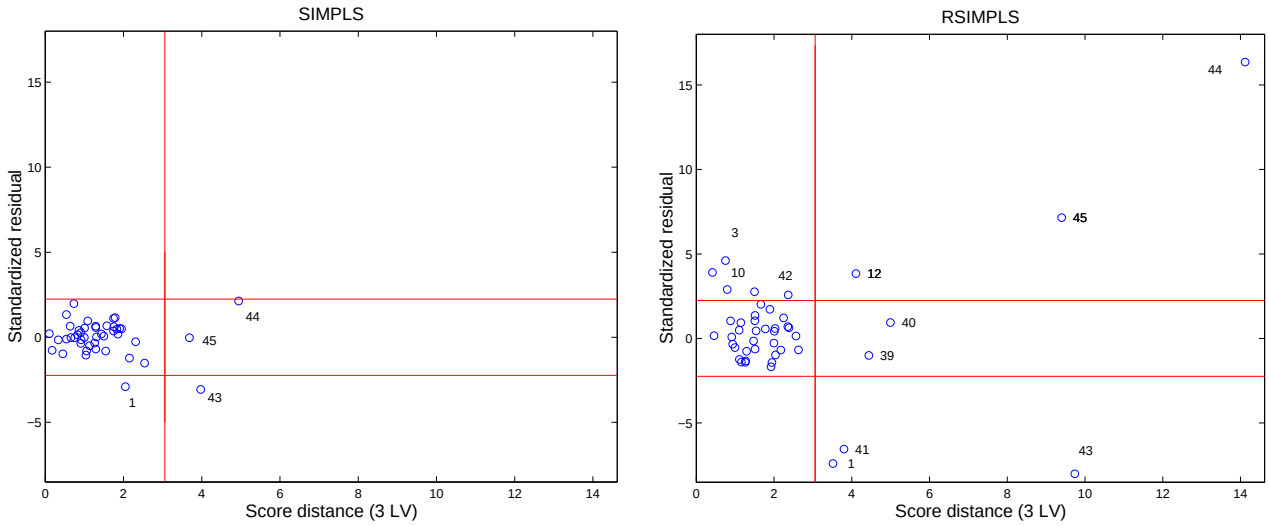



Figure 1: The regressors of the Fish data set.




(a) (b)

Figure 2: Regression diagnostic plot for the Fish data set with: (a) SIMPLS; (b) RSIMPLS.

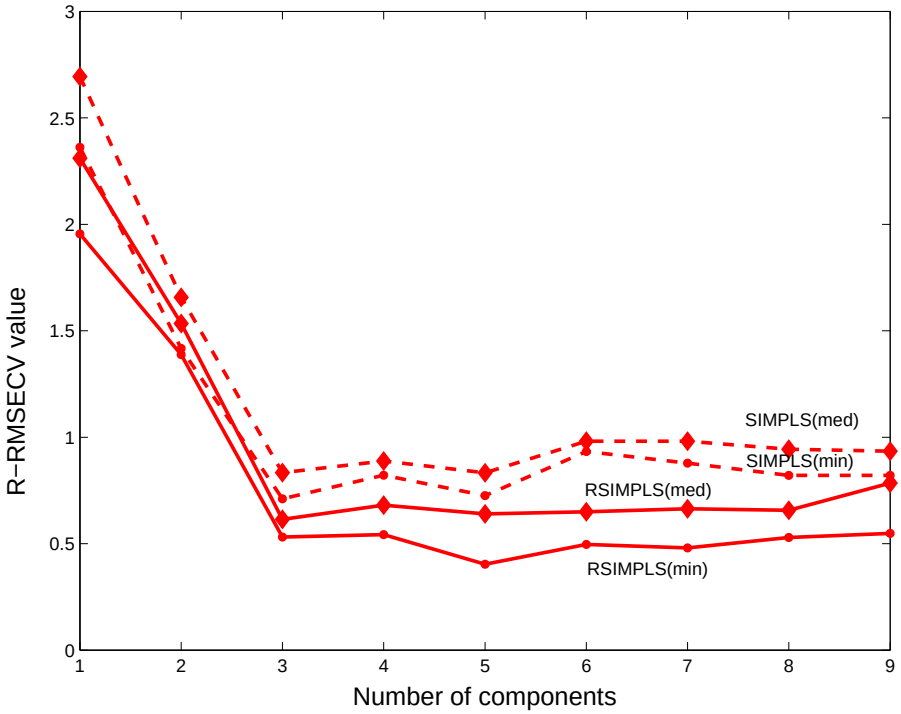
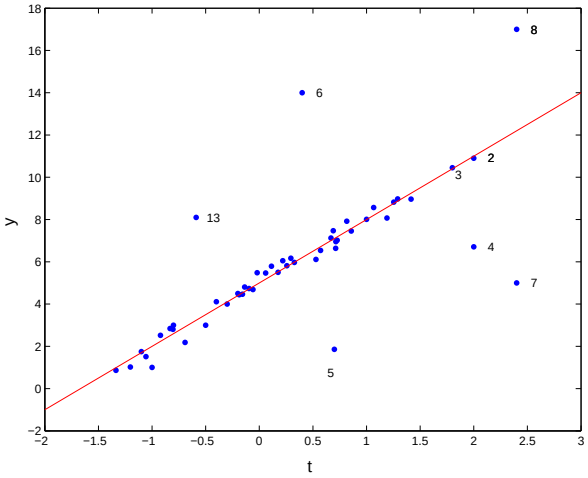
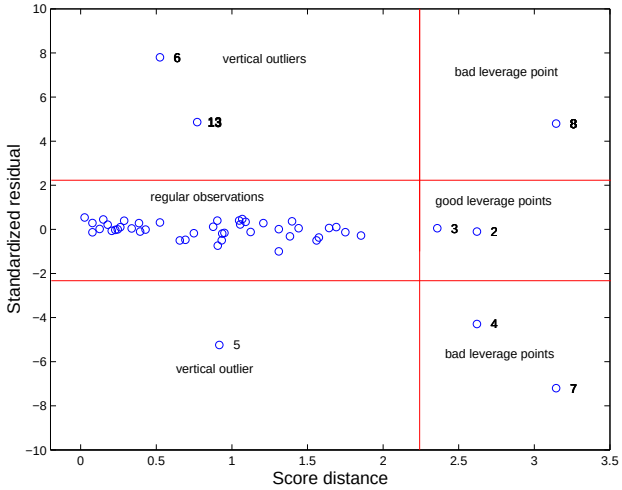


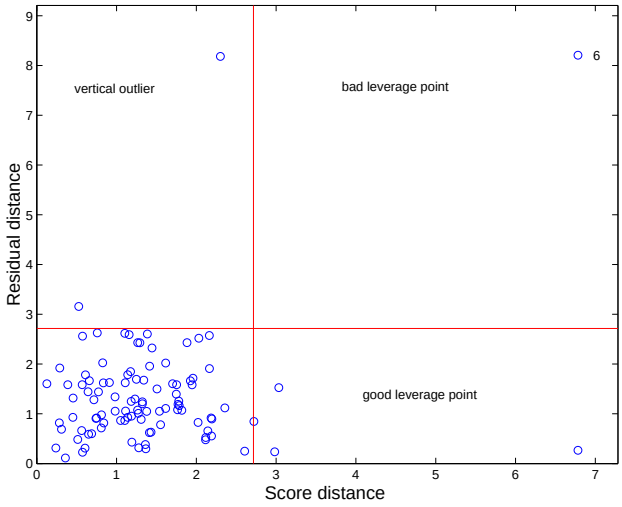

Figure 3: The R-RMSECV curves for the Fish data set: R-RMSECV curve for RSIMPLS based on (31) (solid line and •), for RSIMPLS based on (33) (solid line and ⧫), for SIMPLS based on (31) (dashed line and •) and for SIMPLS based on (33) (dashed line and ⧫).




(a)

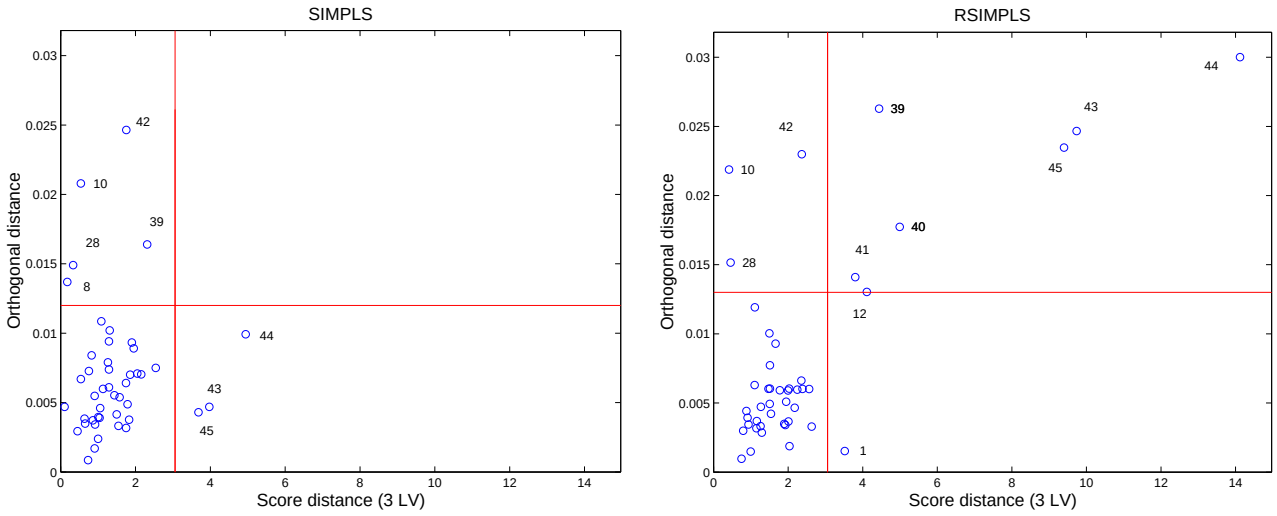



(b)




(c)

Figure 4: Different types of outliers in regression: (a) scatterplot in simple regression; (b) regression diagnostic plot for univariate response variable; (c) regression diagnostic plot for multivariate response variables.




(a) (b)

Figure 5: Score diagnostic plot for the Fish data set with: (a) SIMPLS; (b) RSIMPLS.

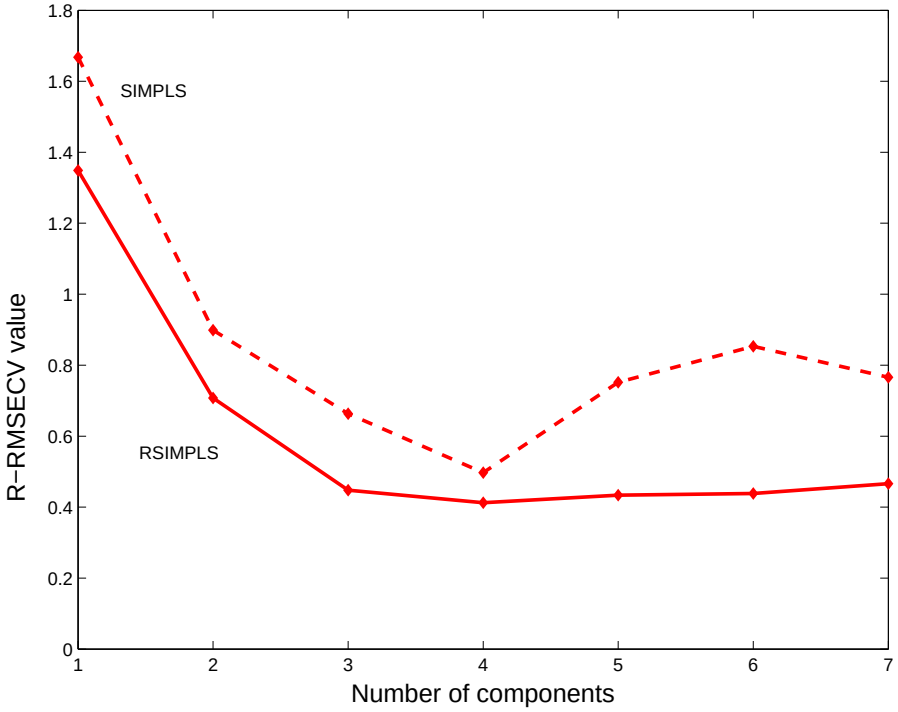
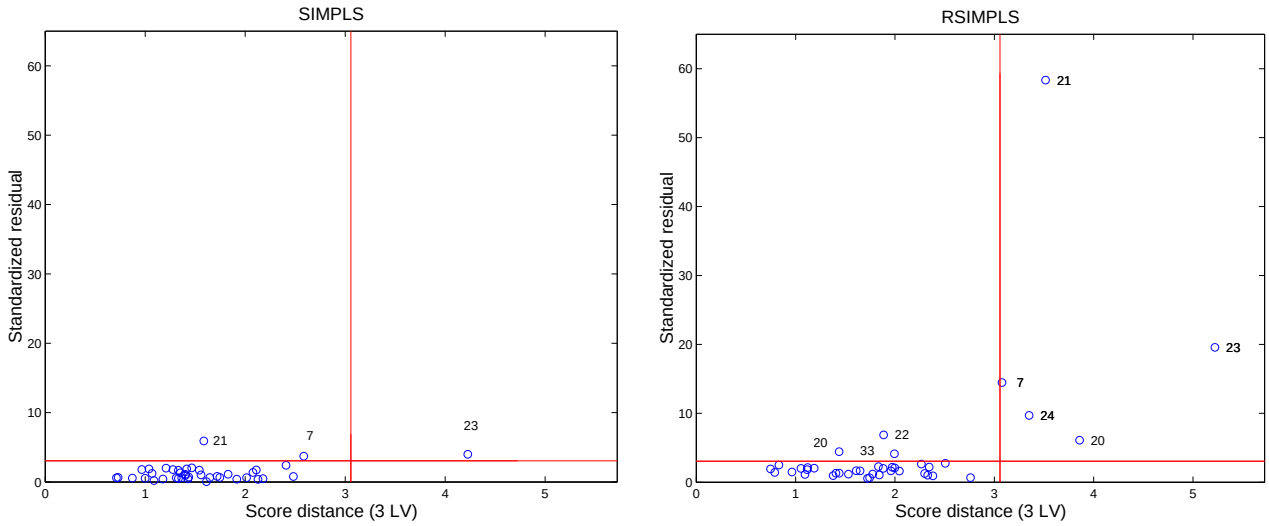
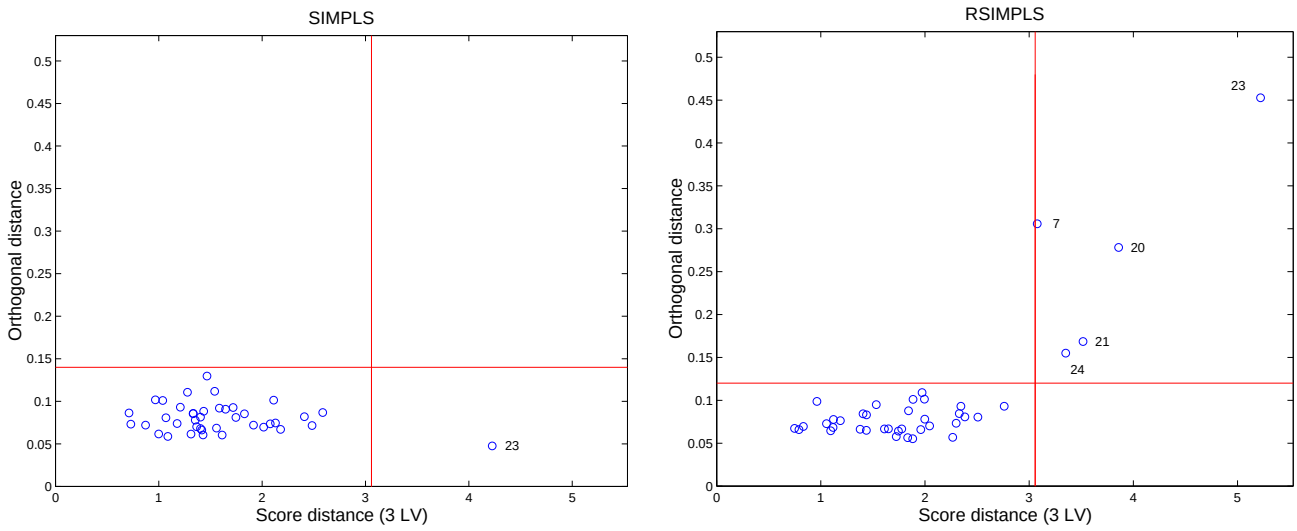



Figure 6: The R-RMSECV curve for the Biscuit dough data set.




(a) (b)

Figure 7: Regression diagnostic plot for the Biscuit dough data set with: (a) SIMPLS; (b) RSIMPLS.




(a) (b)

Figure 8: Score diagnostic plot for the Biscuit dough data set with: (a) SIMPLS; (b) RSIMPLS.

Table 1: The mean CPU-time in seconds over five runs of RSIMCD and RSIMPLS for several set-ups.

| | | $q = 1$ | | | $q = 5$ | | |
|---|---|---|---|---|---|---|---|
| $n$ | $p$ | $k$ | RSIMCD | RSIMPLS | $k$ | RSIMCD | RSIMPLS |
| 50 | 100 | 5 | 10.59 | 6.42 | 5 | 13.64 | 7.45 |
| | | 10 | 14.87 | 7.90 | 10 | 18.22 | 9.03 |
| | 500 | 5 | 10.60 | 6.47 | 5 | 13.81 | 7.68 |
| | | 15 | 14.80 | 8.03 | 15 | 18.62 | 9.41 |
| 100 | 5 | 1 | 7.64 | 5.65 | 5 | 14.38 | 8.08 |
| | | 5 | 11.00 | 6.77 | 10 | 14.43 | 8.07 |
| | 500 | 5 | 11.64 | 7.39 | 5 | 15.11 | 8.80 |
| | | 15 | 16.01 | 9.00 | 15 | 19.93 | 10.57 |

Table 2: Summary of the simulation setup.

| Table | $q$ | $n$ | $p$ | $k$ | $\Sigma_t$ | $\Sigma_e$ |
|---|---|---|---|---|---|---|
| 3 | 1 | 100 | 5 | 2 | diag(4,2) | 1 |
| 4 | 1 | 50 | 100 | 5 | diag(7,5,3.5,2.5,1) | 1 |
| 5 | 5 | 100 | 10 | 3 | diag(4,2,1) | $I_q$ |
| 5 | 5 | 50 | 100 | 5 | diag(7,5,3.5,2.5,1) | $I_q$ |

Table 3: Simulation results for low-dimensional regressors ($p = 5$) and one response variable ($q = 1$).

| | mean(angle) | MSE($\hat{\boldsymbol{\beta}}$) | MSE($\hat{\beta}_0$) | MSE($\hat{\sigma}_e^2$) |
|---|---|---|---|---|
| Algorithm | No contamination | | | |
| SIMPLS | 0.054 | 0.404 | 1.324 | 13.073 |
| RSIMCD | 0.082 | 0.734 | 1.649 | 4.196 |
| RSIMPLS | 0.080 | 0.701 | 1.541 | 6.560 |
| | 10% bad leverage points | | | |
| SIMPLS | 1.132 | 60.236 | 19.075 | 7696 |
| RSIMCD | 0.076 | 0.645 | 1.645 | 8.400 |
| RSIMPLS | 0.077 | 0.684 | 1.823 | 4.888 |
| | 10% vertical outliers | | | |
| SIMPLS | 0.124 | 2.072 | 100 | 8329 |
| RSIMCD | 0.073 | 0.613 | 1.661 | 7.802 |
| RSIMPLS | 0.075 | 0.654 | 1.776 | 4.815 |
| | 10% orthogonal outliers | | | |
| SIMPLS | 0.258 | 5.616 | 1.988 | 67.455 |
| RSIMCD | 0.079 | 0.766 | 2.278 | 10.022 |
| RSIMPLS | 0.078 | 0.735 | 2.063 | 5.111 |

Table 4: Simulation results for high-dimensional regressors ($p = 100$) and one response variable ($q = 1$).

| | mean(angle) | MSE($\hat{\boldsymbol{\beta}}$) | MSE($\hat{\beta}_0$) | MSE($\hat{\sigma}_e^2$) |
|---|---|---|---|---|
| Algorithm | No contamination | | | |
| SIMPLS | 0.565 | 2.291 | 2.990 | 7.173 |
| RSIMCD | 0.429 | 1.127 | 4.595 | 13.679 |
| RSIMPLS | 0.424 | 1.088 | 4.084 | 11.347 |
| | 10% bad leverage points | | | |
| SIMPLS | 0.968 | 12.516 | 7.920 | 336 |
| RSIMCD | 0.420 | 1.081 | 4.651 | 15.290 |
| RSIMPLS | 0.417 | 1.052 | 4.612 | 10.707 |
| | 10% vertical outliers | | | |
| SIMPLS | 1.020 | 14.782 | 56.320 | 165 |
| RSIMCD | 0.509 | 1.645 | 5.814 | 18.693 |
| RSIMPLS | 0.504 | 1.578 | 6.208 | 13.360 |
| | 10% orthogonal outliers | | | |
| SIMPLS | 0.413 | 1.085 | 2.846 | 3.436 |
| RSIMCD | 0.417 | 1.060 | 4.464 | 17.374 |
| RSIMPLS | 0.414 | 1.039 | 4.333 | 11.877 |

Table 5: Simulation results for low- and high-dimensional regressors ($p = 10$ or $p = 100$) and five response variables ($q = 5$).

| | $n = 100, p = 10$ | | | $n = 50, p = 100$ | | |
|---|---|---|---|---|---|---|
| | MSE($\hat{\boldsymbol{\mathcal{B}}}$) | MSE($\hat{\boldsymbol{\beta}}_0$) | MSE($\hat{\Sigma}_e$) | MSE($\hat{\boldsymbol{\mathcal{B}}}$) | MSE($\hat{\boldsymbol{\beta}}_0$) | MSE($\hat{\Sigma}_e$) |
| Algorithm | No contamination | | | | | |
| SIMPLS | 0.599 | 1.544 | 14.827 | 0.248 | 1.647 | 3.125 |
| RSIMCD | 0.940 | 1.773 | 12.120 | 0.475 | 3.607 | 4.343 |
| RSIMPLS | 0.965 | 1.843 | 11.348 | 0.468 | 3.120 | 3.849 |
| | 10% bad leverage points | | | | | |
| SIMPLS | 18.504 | 5.836 | 1328 | 1.271 | 3.945 | 178 |
| RSIMCD | 0.910 | 1.874 | 12.710 | 0.435 | 3.023 | 4.053 |
| RSIMPLS | 0.933 | 1.978 | 11.795 | 0.431 | 3.418 | 3.598 |
| | 10% vertical outliers | | | | | |
| SIMPLS | 3.389 | 103 | 7796 | 7.049 | 52.598 | 730 |
| RSIMCD | 0.921 | 1.889 | 12.735 | 0.441 | 2.931 | 4.144 |
| RSIMPLS | 0.945 | 1.973 | 11.843 | 0.437 | 3.252 | 3.682 |
| | 10% orthogonal outliers | | | | | |
| SIMPLS | 4.230 | 2.166 | 34.781 | 0.397 | 2.314 | 13.502 |
| RSIMCD | 1.071 | 2.632 | 18.278 | 0.430 | 2.932 | 4.156 |
| RSIMPLS | 0.984 | 2.192 | 13.551 | 0.427 | 3.184 | 3.622 |