Robust Methods for High-Dimensional Data

Chapter · January 2020

DOI: 10.1016/B978-0-12-409547-2.14883-8

1 author:

Mia Hubert
KU Leuven
124 PUBLICATIONS **9,201** CITATIONS

SEE PROFILE

Robust methods for high-dimensional data

In: Reference Module in Chemistry, Molecular Sciences and Chemical Engineering, Elsevier, 2020.

Mia Hubert

Department of Mathematics, KU Leuven, Celestijnenlaan 200B, BE-3001 Leuven, Belgium, Mia.Hubert@kuleuven.be

Abstract

When dealing with real data, it often occurs that some observations deviate from the majority of the samples. Since classical methods are sensitive to these outliers, robust methods have been developed that are less sensitive to them. In this chapter we give an overview of robust methods that are designed for high-dimensional data, such as Principal Component Analysis, Principal Component Regression and Partial Least Squares Regression. Also robust classification of high-dimensional data is discussed, as well as robust methods for the analysis of multi-way data.

Keywords: Outlier detection; Outliers; Cellwise outliers; Robust Estimation; Robustness; Principal Component Analysis; Sparse Principal Component Analysis; Principal Component Regression; Partial Least Squares Regression; Classification; Multi-way data.

1 Introduction

Experimental data often contain outliers of one type or another. These observations deviate from the usual assumptions, and/or from the pattern suggested by the majority of the data. Sometimes the outliers are due to recording or copying mistakes, or to a change or failure of the instruments. Often the outlying observations are not incorrect but they were made under exceptional circumstances, or they belong to another population (e.g. it may have been the concentration of a different compound), and consequently they do not fit the model well. It is very important to be able to detect these outliers. For instance, they can pinpoint a change in the production process or in the experimental conditions.

It is however not at all easy to find outlying cases, especially in multivariate and high-dimensional data sets which might consist of many cases and/or variables. A univariate screening of the variables is in most cases not sufficient. Consider e.g. the bivariate data set of Figure 1. It shows the concentration of inorganic phosphorus and organic phosphorus in soil [1]. Apparantly, there are no outlying values in X_1 (inorganic phosphorus) nor in X_2 (organic phosphorus). Only when we take into account the joint distribution of the variables (and hence the covariance structure), the outlying samples can be seen.

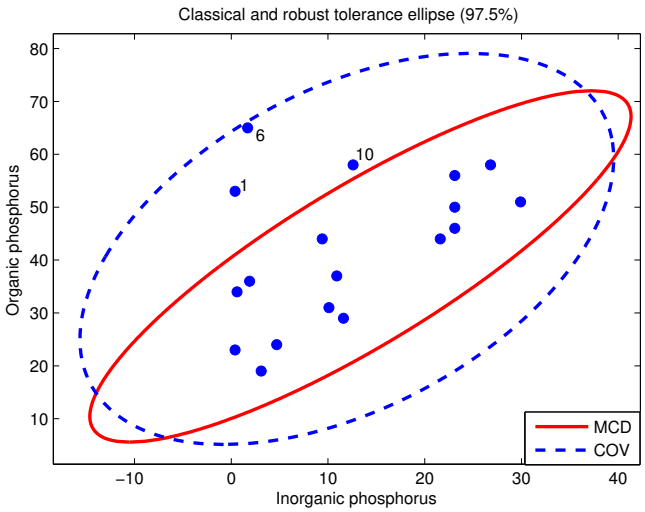


Figure 1: Classical (COV) and robust (MCD) tolerance ellipse of the phosphorus data set.

Unfortunately most standard statistical methods are very sensitive to outliers. In Figure 1 we have plotted the classical tolerance ellipse, defined as the set of two-dimensional points $\boldsymbol{x}$ whose *Mahalanobis distance*

$$\mathrm{MD}(\boldsymbol{x}) = \sqrt{(\boldsymbol{x} - \bar{\boldsymbol{x}})^T \boldsymbol{S}_x^{-1} (\boldsymbol{x} - \bar{\boldsymbol{x}})} \tag{1}$$

equals $\sqrt{\chi^2_{2,0.975}}$, the square root of the 0.975 quantile of the chi-square distribution with two degrees of freedom. (Note that observations are denoted by column vectors.) This Mahalanobis distance is based on the classical measures of location and scatter, being the empirical mean $\bar{\boldsymbol{x}} = \frac{1}{n}\sum_{i=1}^n \boldsymbol{x}_i$, and the empirical covariance matrix $\boldsymbol{S}_x = \sum_{i=1}^n (\boldsymbol{x}_i - \bar{\boldsymbol{x}})(\boldsymbol{x}_i - \bar{\boldsymbol{x}})^T/(n-1)$. If the data are normally distributed, about 97.5% of the observations should fall inside this ellipse. Observations outside this ellipse are suspected to be outliers. We see however that all data points lie inside the classical tolerance ellipse. This is called the *masking* effect: classical methods can be affected by outliers so strongly that the resulting fitted model does not allow to detect the deviating observations. Additionally, some good data points might even appear to be outliers, which is known as *swamping*. To avoid these effects, the goal of *robust statistics* is to find a fit which is similar to the fit we would have found without the outliers. We can then identify the outliers by their large residuals from that robust fit. Robust parametric methods assume that the majority of the samples follows a statistical model, such as a multivariate normal distribution in covariance estimation, a linear model with normal errors in regression etc.

Robust methods for low-dimensional data (location, scatter, correlation, regression) are detailed in the chapter 'Robust multivariate statistical methods' of this book. In this chapter we describe robust procedures for high-dimensional data. We start with several approaches for robust Principal Component Analyis (PCA). We then present chemometrical methods for robust calibration (PCR, PLS) as well as robustness in multi-way data. Further, we

discuss robust classification and we present some results on the robustness of Support Vector Machines, which are kernel methods for non-linear classification and regression. Finally, we review software availability. Throughout, we will illustrate many methods on real data, and apply outlier detection tools.

2 Principal component analysis

2.1 Classical PCA

In the multivariate location and scatter setting we assume that the data are stored in an $n \times p$ data matrix $\boldsymbol{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)^T$ with $\boldsymbol{x}_i = (x_{i1}, \ldots, x_{ip})^T$ the ith observation. Hence n stands for the number of objects and p for the number of variables. High-dimensional data often have more variables than observations, hence it is allowed that $n < p$. When appropriate, we add the dimensions of a matrix as subscripts, so $\boldsymbol{X} = \boldsymbol{X}_{n,p}$.

Principal component analysis is a popular statistical method which tries to explain the covariance structure of data by means of a small number of components. These components are linear combinations of the original variables, and often allow for an interpretation and a better understanding of the different sources of variation. Because PCA is concerned with data reduction, it is widely used for the analysis of high-dimensional data which are frequently encountered in chemometrics. PCA is then often the first step of the data analysis, followed by classification, clustering, or other multivariate techniques, see e.g. [2].

More precisely, we apply PCA when we assume that the data can be well represented in a lower-dimensional subspace. That is, we assume that

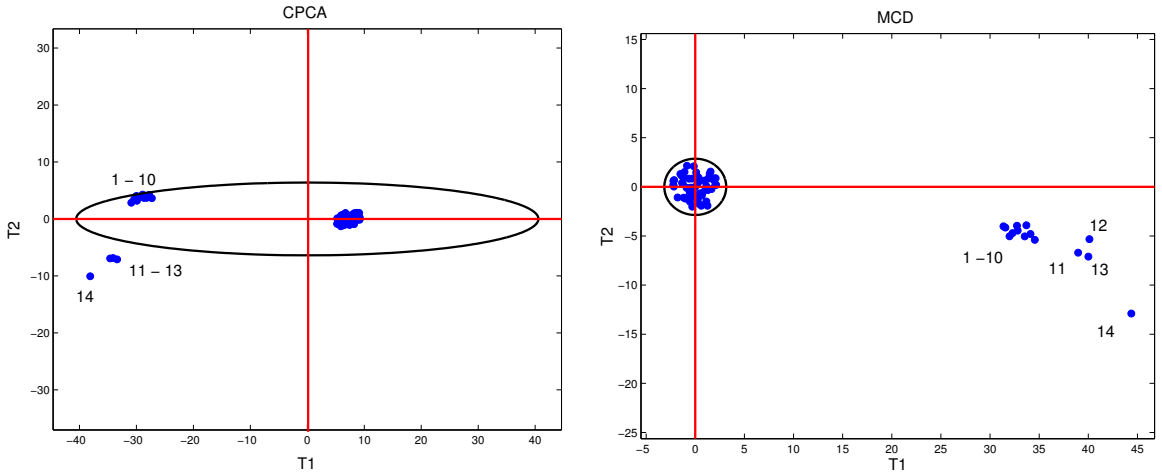
$$\boldsymbol{x}_i = \boldsymbol{\mu}_x + \boldsymbol{\mathcal{P}}_{p,k}\boldsymbol{t}_i + \boldsymbol{\varepsilon}_i \tag{2}$$

with $\boldsymbol{\mu}_x$ the p-variate column vector of location, $\boldsymbol{\mathcal{P}}_{p,k}$ the $p \times k$ loadings matrix whose columns span the PCA subspace, $\boldsymbol{t}_i$ the k-variate score vector, and $\boldsymbol{\varepsilon}_i$ the error. The reduced dimension k can vary from 1 to p but we assume that k is low.

In the classical approach, the first component corresponds to the direction in which the projected observations have the largest variance. The second component is then orthogonal to the first one and again maximizes the variance of the projected data points. Continuing in this way produces all the principal components, which correspond to the eigenvectors of the empirical covariance matrix $\boldsymbol{S}_x$. Unfortunately, both the classical variance (which is being maximized) and the classical covariance matrix (which is being decomposed) are very sensitive to anomalous observations. Consequently, the first components are often pulled towards outlying points, and they may not capture the variation of the regular observations. Therefore, data reduction based on classical PCA (CPCA) becomes unreliable if outliers are present in the data.

To illustrate this, let us consider a small artificial dataset in $p = 4$ dimensions. The Hawkins-Bradu-Kass dataset [1] consists of $n = 75$ observations in which two groups of outliers were created, labelled 1-10 and 11-14. The first two eigenvalues explain already 98%

of the total variation, so we select $k = 2$. The CPCA scores plot is depicted in Figure 2(a). In this figure we can clearly distinguish the two groups of outliers, but we see several other undesirable effects. We first observe that, although the scores have zero mean, the regular data points lie far from zero. This stems from the fact that the mean of the data points is a bad estimate of the true center of the data in the presence of outliers. It is clearly shifted towards the outlying group, and consequently the origin even falls outside the cloud of the regular data points. On the plot we have also superimposed the 97.5% tolerance ellipse. We see that the outliers 1-10 are within the tolerance ellipse, and thus do not stand out based on their Mahalanobis distance. The ellipse has stretched itself to accommodate these outliers.



(a) (b)

Figure 2: Score plot and 97.5% tolerance ellipse of the Hawkins-Bradu-Kass data obtained with (a) CPCA; (b) MCD.

2.2 Robust PCA based on a robust covariance matrix

The goal of robust PCA methods is to obtain principal components that are not influenced much by outliers. A first group of methods is obtained by replacing the classical covariance matrix by a robust covariance estimator. In [3] and [4] it is proposed to use M-estimators of scatter for this purpose, but these cannot resist many outliers.

To cope with a potentially large amount of outliers, one can rely on a high-breakdown estimator of location and scatter, such as the Minimum Covariance Determinant (MCD) method [5]. The MCD [6] looks for those h observations in the data set (where the number $n/2 < h < n$ is given by the user) whose classical covariance matrix has the lowest possible determinant. The MCD estimate of location $\hat{\boldsymbol{\mu}}_0$ is then the average of these h points, whereas the MCD estimate of scatter $\hat{\boldsymbol{\Sigma}}_0$ is their covariance matrix, multiplied with a consistency factor. Based on the raw MCD estimates, a reweighing step can be added which increases the finite-sample efficiency considerably. In general, we can give each $\boldsymbol{x}_i$ some weight w_i, for instance by putting $w_i = 1$ if $(\boldsymbol{x}_i - \hat{\boldsymbol{\mu}}_0)^T \hat{\boldsymbol{\Sigma}}_0^{-1} (\boldsymbol{x}_i - \hat{\boldsymbol{\mu}}_0) \leqslant \chi^2_{p,0.975}$ and $w_i = 0$ otherwise. The

resulting reweighed mean and covariance matrix are then defined as

$$\hat{\boldsymbol{\mu}}_R(\boldsymbol{X}) = \Big(\sum_{i=1}^{n} w_i \boldsymbol{x}_i\Big) / \Big(\sum_{i=1}^{n} w_i\Big) \tag{3}$$

$$\hat{\boldsymbol{\Sigma}}_R(\boldsymbol{X}) = \Big(\sum_{i=1}^{n} w_i (\boldsymbol{x}_i - \hat{\boldsymbol{\mu}}_R)(\boldsymbol{x}_i - \hat{\boldsymbol{\mu}}_R)^T\Big) / \Big(\sum_{i=1}^{n} w_i - 1\Big). \tag{4}$$

The robustness of an estimator can be measured by means of its *breakdown value* [7]. For a location estimator, the breakdown value is the smallest proportion of observations in the data set that need to be replaced to carry the estimate arbitrarily far away. Similarly, the breakdown value of a covariance matrix estimator is defined as the smallest fraction of outliers that can either take the largest eigenvalue to infinity or the smallest eigenvalue to zero. The MCD estimators of location and scatter have a breakdown value $(n - h + 1)/n$, hence the number h determines the robustness of the estimator. The highest breakdown value (50%) is attained when $h = [(n+p+1)/2]$. When a large proportion of contamination is presumed, h should thus be chosen close to $0.5n$. Otherwise an intermediate value for h, such as $0.75n$, is recommended to obtain a higher finite-sample efficiency.

The MCD location and scatter estimates are *affine equivariant*, which means that they behave properly under affine transformations (such as rotations and rescaling) of the data. That is, if the $\boldsymbol{x}_i$ yield the MCD estimates $\hat{\boldsymbol{\mu}}_R$ and $\hat{\boldsymbol{\Sigma}}_R$, then the MCD estimates of the transformated data $\boldsymbol{A}\boldsymbol{x}_i + \boldsymbol{v}$ are equal to $\boldsymbol{A}\hat{\boldsymbol{\mu}}_R + \boldsymbol{v}$ and $\boldsymbol{A}\hat{\boldsymbol{\Sigma}}_R\boldsymbol{A}^T$ for all nonsingular $p \times p$ matrices $\boldsymbol{A}$ and vectors $\boldsymbol{v} \in \mathbb{R}^p$. More properties of the MCD and its computation are discussed in [8] and in the chapter 'Robust multivariate statistical methods' in this book.

Let us first reconsider the phosphorus data of Figure 1. Contrary to the classical mean and covariance matrix, a robust method yields a tolerance ellipse which captures the covariance structure of the majority of the data points. Starting from $\hat{\boldsymbol{\mu}}_R$ and $\hat{\boldsymbol{\Sigma}}_R$, we plot the points $\boldsymbol{x}$ whose *robust distance*

$$\mathrm{RD}(\boldsymbol{x}) = \sqrt{(\boldsymbol{x} - \hat{\boldsymbol{\mu}}_R)^T \hat{\boldsymbol{\Sigma}}_R^{-1} (\boldsymbol{x} - \hat{\boldsymbol{\mu}}_R)} \tag{5}$$

is equal to $\sqrt{\chi^2_{2,0.975}}$. In Figure 1, this robust tolerance ellipse is much narrower than the classical one. Observations 1, 6 and 10 now lie outside the ellipse, and are flagged as suspicious observations. The eigenvectors of $\hat{\boldsymbol{\Sigma}}_R$ serve as robust principal components, whereas $\hat{\boldsymbol{\mu}}_R$ acts as a robust center. The PCs then correspond with the principal axes of the tolerance ellipse. Here, the classical and robust principal components do not differ that much, but the second robust eigenvalue is clearly much smaller than the one based on CPCA.

Let us next reconsider the Hawkins-Bradu-Kass data in $p = 4$ dimensions. Robust PCA using the MCD estimator yields the score plot in Figure 2(b). We now see that the center is correctly estimated in the middle of the regular observations. The 97.5% tolerance ellipse nicely encloses these points and excludes all 14 outliers.

Instead of using the MCD, Salibian et al. [9] proposed using S- or MM-estimators of scatter and developed a fast robust bootstrap procedure for inference and to assess the stability of the PCA solution.

Unfortunately the use of these affine equivariant covariance estimators is limited to small to moderate dimensions. To see why, consider e.g. the MCD estimator. It can only be computed if $p < h$, otherwise the covariance matrix of any h-subset has zero determinant. Since $h < n$, p can never be larger than n.

Note that classical PCA is not affine equivariant because it is sensitive to a rescaling of the variables. But it is still translation and orthogonally equivariant, which means that the center and the principal components transform appropriately under rotations, reflections and translations of the data. More formally, it allows transformations $\boldsymbol{A}\boldsymbol{x}_i + \boldsymbol{v}$ for any orthogonal matrix $\boldsymbol{A}$ (that satisfies $\boldsymbol{A}^{-1} = \boldsymbol{A}^T$). Any robust PCA method thus only has to be orthogonally equivariant.

2.3 Robust PCA based on projection pursuit

A second approach to robust PCA uses *Projection Pursuit* (PP) techniques. These methods maximize a robust measure of spread to obtain consecutive directions on which the data points are projected. The original idea [10] uses M-estimators of scale and is quite computationally involved. Faster algorithms are presented in [11, 12] and [13] where the method is called RAPCA (*reflection algorithm* for PCA), see also [14]. All these algorithms start by reducing the data space to the affine subspace spanned by the n observations. This is done quickly and accurately by a singular value decomposition (SVD) of $\boldsymbol{X}_{n,p}$. Let $\tilde{\boldsymbol{X}}$ denote the mean-centered data matrix. Standard SVD computes the eigenvectors of $\tilde{\boldsymbol{X}}^T\tilde{\boldsymbol{X}}$ which is a matrix of size $p \times p$. As the dimension p can be in the hundreds or thousands, this is computationally expensive. The computational speed can be increased by computing the eigenvectors $\boldsymbol{v}$ of $\tilde{\boldsymbol{X}}\tilde{\boldsymbol{X}}^T$ which is an $n \times n$ matrix. The transformed $\tilde{\boldsymbol{X}}^T\boldsymbol{v}$ vectors then yield the eigenvectors of $\tilde{\boldsymbol{X}}^T\tilde{\boldsymbol{X}}$, whereas the eigenvalues remain the same. This is known as the kernel version of the eigenvalue decomposition [15]. Note that this singular value decomposition is just an affine transformation of the data. It is not used to retain only the first eigenvectors of the covariance matrix of $\boldsymbol{X}$. This would imply that classical PCA is performed, which is of course not robust. Here, the data are merely represented in their own dimensionality $r = \text{rank}(\tilde{\boldsymbol{X}}) \leqslant n - 1$. This step is useful as soon as $p > r$. When $p \gg n$ we obtain a huge reduction in size. For spectral data, e.g. $n = 50, p = 1000$, this reduces the 1000-dimensional original data set to one in only 49 dimensions.

The main step of the projection pursuit algorithms is then to search for the direction in which the projected observations have the largest robust scale. To measure the univariate scale the Q_n estimator [16] is used, which attains a breakdown value of 50%. For univariate data $\{x_1, \ldots, x_n\}$ the Q_n estimator is defined as

$$Q_n = 2.2219\, c_n \left\{|x_i - x_j|; i < j\right\}_{(k)} \tag{6}$$

with $k = \binom{h}{2} \approx \binom{n}{2}/4$ and $h = [\frac{n}{2}] + 1$. The notation (k) stands for the kth order statistic out of the $\binom{n}{2} = \frac{n(n-1)}{2}$ possible differences $|x_i - x_j|$ and $[z]$ for the largest integer smaller or equal to z. This scale estimator is thus essentially the first quartile of all pairwise differences

between two data points. Then constant 2.2219 ensures that the estimator is asymptotically consistent at the normal distribution, whereas c_n is a small-sample correction factor which makes Q_n an unbiased estimator at finite samples (note that c_n only depends on the sample size n, and that $c_n \to 1$ for increasing n).

Comparisons using other scale estimators are presented in [11] and [17]. To make the algorithm computationally feasible, the collection of directions to be investigated need to be restricted. In [11, 13] all directions that pass through the L_1-median and a data point are considered. The L_1-median is a highly robust (50% breakdown value) and orthogonally equivariant location estimator, also known as the *spatial median*. It is defined as the point $\boldsymbol{\theta}$ which minimizes the sum of the distances to all observations, i.e.

$$\text{minimize} \sum_{i=1}^{n} \|\boldsymbol{x}_i - \boldsymbol{\theta}\|.$$

The GRID algorithm [12] uses a grid search to find the optimal direction, and is more efficient.

Once the first eigenvector $\boldsymbol{v}_1$ is found, the method can then be applied in the orthogonal complement to search for the second eigenvector and so on. It is not required to compute all eigenvectors, which would be very time-consuming for high p, but the computations can be stopped as soon as the required number of components has been found.

Note that a PCA analysis often starts by prestandardizing the data in order to obtain variables that all have the same spread. Otherwise, the variables with a large variance compared to the others will dominate the first principal components. Standardizing by the mean and the standard deviation of each variable yields a PCA analysis based on the correlation matrix instead of the covariance matrix. One can also standardize each variable j in a robust way, e.g. by first subtracting its median $\text{med}(x_{1j}, \ldots, x_{nj})$ and then dividing by its robust scale estimate $Q_n(x_{1j}, \ldots, x_{nj})$.

2.4 Robust PCA based on projection pursuit and the MCD

Another approach to robust PCA has been proposed in [18] and is called ROBPCA. This method combines ideas of both projection pursuit and robust covariance estimation. The projection pursuit part is used for the initial dimension reduction, whereas the MCD estimator is applied to this lower-dimensional data space. Simulations have shown that this combined approach yields more accurate estimates than the raw projection pursuit algorithm RAPCA. The complete description of the ROBPCA method is quite involved, so here we will only outline the main stages of the algorithm.

First, as in the PP algorithms, the data are preprocessed by reducing their data space to the affine subspace spanned by the n observations. As a result, the data are represented using at most $n - 1 = \text{rank}(\tilde{\boldsymbol{X}}_{n,p})$ variables without loss of information.

In the second step of the ROBPCA algorithm, the outlyingness of each data point is

computed as

$$\text{outl}(\boldsymbol{x}_i) = \max_{\boldsymbol{v} \in B} \frac{|\boldsymbol{v}'\boldsymbol{x}_i - \hat{\mu}_{\text{MCD}}(\boldsymbol{v}'\boldsymbol{x}_j)|}{\hat{\sigma}_{\text{MCD}}(\boldsymbol{v}'\boldsymbol{x}_j)} \quad , \tag{7}$$

where $\hat{\mu}_{\text{MCD}}(\boldsymbol{v}'\boldsymbol{x}_j)$ and $\hat{\sigma}_{\text{MCD}}(\boldsymbol{v}'\boldsymbol{x}_j)$ are the univariate MCD location and scale estimators of $\{\boldsymbol{v}'\boldsymbol{x}_1, \ldots, \boldsymbol{v}'\boldsymbol{x}_n\}$. The set B contains 250 directions through two data points (or all of them if there are fewer than 250).

Next, classical PCA is performed on the $n/2 < h < n$ data points with smallest outlyingness, thereby retaining k principal components. This analysis yields a center $\hat{\boldsymbol{\mu}}_x$, a loading matrix $\boldsymbol{P}_{p,k}$ with orthogonal columns and scores $\boldsymbol{t}_i = \boldsymbol{P}_{k,p}^T(\boldsymbol{x}_i - \hat{\boldsymbol{\mu}}_x)$ for each data point. These k-dimensional scores represent the coordinates of the projections of the centered $\boldsymbol{x}_i$ onto the current PCA-subspace.

In the next step, the *orthogonal distance* is considered for each data point. This orthogonal distance measures the distance (or residual) between an observation $\boldsymbol{x}_i$ and its projection $\hat{\boldsymbol{x}}_i$ in the k-dimensional PCA subspace:

$$\hat{\boldsymbol{x}}_i = \hat{\boldsymbol{\mu}}_x + \boldsymbol{P}_{p,k}\boldsymbol{t}_i \tag{8}$$

$$\text{OD}_i = \|\boldsymbol{x}_i - \hat{\boldsymbol{x}}_i\|. \tag{9}$$

Next, a reweighting step is added. Whereas initially PCA is performed on h observations, classical PCA is now applied onto all observations whose orthogonal distance is not too large (see [19] for details about the cutoff value), yielding a new k-dimensional subspace. Depending on the choice of h and the number of outliers, this reweighted subspace might be based on (many) more cases than h.

The last stage of ROBPCA then consists of projecting all the data points onto this subspace and of computing their center and shape by means of the reweighed MCD estimator. The eigenvectors of this scatter matrix then determine the final robust principal components $\boldsymbol{P}_{p,k}$. The MCD location estimate serves as the final robust center $\hat{\boldsymbol{\mu}}_x$. This allows to compute the final scores $\boldsymbol{t}_i = \boldsymbol{P}_{k,p}^T(\boldsymbol{x}_i - \hat{\boldsymbol{\mu}}_x)$.

Let $\boldsymbol{L}_{k,k}$ denote the diagonal matrix which contains the k eigenvalues l_j of the MCD scatter matrix, sorted from largest to smallest. Thus $l_1 \geqslant l_2 \geqslant \ldots \geqslant l_k$. The *score distance* of the ith sample measures the robust distance of its projection to the center of all the projected observations. Hence, it is measured within the PCA subspace, where due to the knowledge of the eigenvalues, we have information about the covariance structure of the scores. Consequently, the score distance is defined as in (5):

$$\text{SD}_i = \sqrt{\boldsymbol{t}_i^T \boldsymbol{L}^{-1} \boldsymbol{t}_i} = \sqrt{\sum_{j=1}^{k} (t_{ij}^2 / l_j)}. \tag{10}$$

Moreover, the k robust principal components generate a $p \times p$ robust scatter matrix $\hat{\boldsymbol{\Sigma}}_x$ of rank k given by

$$\hat{\boldsymbol{\Sigma}}_x = \boldsymbol{P}_{p,k}\boldsymbol{L}_{k,k}\boldsymbol{P}_{k,p}^T. \tag{11}$$

Note that all results (the scores $\boldsymbol{t}_i$, the scores distances, the orthogonal distances and the scatter matrix $\hat{\boldsymbol{\Sigma}}_x$) depend on the number of components k. But to simplify the notations, we do not explicitly add a subscript k.

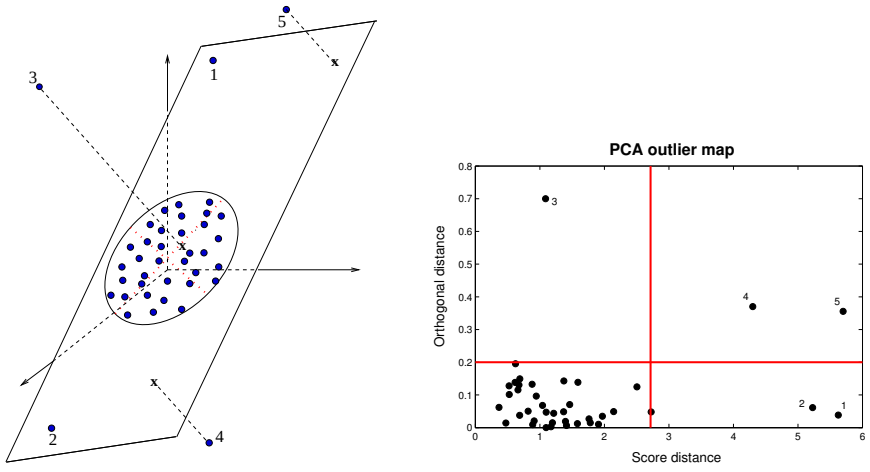
Also note that the k-dimensional subspace spanned by the ROBPCA eigenvectors depends on k through the first reweighting step. Hence, this subspace will in general not be nested into the $(k+1)$-dimensional subspace that is found by applying ROBPCA with $k+1$ components. Consequently, also the resulting principal components are not subsets of each other. This implies that the ROBPCA method is not very fast when the results are required for a set of $k = \{1, \ldots, k_{max}\}$ values. A modified algorithm ROBPCA$_{kmax}$ [19] circumvents this problem, but might be less precise if k_{max} is chosen too large.

Other proposals for robust PCA include the robust Least Trimmed Squares (LTS) subspace estimator and its generalizations, introduced and discussed in [1] and [20, 21]. The idea behind these approaches consists in minimizing a robust scale of the orthogonal distances, similar to the LTS estimator and S-estimators in regression (see also Section 3.1). The spatial sign covariance matrix and its generalizations give rise to robust PCA methods which are very fast to compute, see [22, 23]. More advanced methods for functional data are proposed in [24, 25].

2.5 Outlier map

The result of a PCA analysis can be represented by means of an outlier map, as described in [18]. This figure highlights the outliers and classifies them into several types. In the context of PCA, an outlier either lies far from the subspace spanned by the k eigenvectors, and/or the projected observation lies far from the bulk of the data within this subspace. This can be expressed by means of the orthogonal and the score distances. These two distances define four types of observations, as illustrated in Figure 3(a). *Regular observations* have a small orthogonal and a small score distance. *Bad leverage points*, such as observations 4 and 5, have a large orthogonal distance and a large score distance. They typically have a large influence on classical PCA, as the eigenvectors will be tilted towards them. When points have a large score distance but a small orthogonal distance, we call them *good leverage points*. Observations 1 and 2 in Figure 3(a) can be classified into this category. Finally, *orthogonal outliers* have a large orthogonal distance, but a small score distance, as for example case 3. They cannot be distinguished from the regular observations once they are projected onto the PCA subspace, but they lie far from this subspace.

The outlier map in Figure 3(b) displays the OD_i versus the SD_i. In this plot, lines are drawn to distinguish the observations with a small and a large OD, and with a small and a large SD. For the latter distances, the cutoff value $c = \sqrt{\chi^2_{k,0.975}}$ is used. For the orthogonal distances the approach of [26] is followed. The squared orthogonal distances can be approximated by a scaled χ^2 distribution which in its turn can be approximated by a normal distribution using the Wilson-Hilferty transformation. The mean and variance of this normal distribution are then estimated by applying the univariate MCD to the $\mathrm{OD}_i^{2/3}$.



(a) (b)

Figure 3: (a) Different types of outliers when a three-dimensional dataset is projected on a robust two-dimensional PCA-subspace; (b) the corresponding PCA outlier map.

2.6 Selecting the number of principal components

To choose the optimal number of components k_{opt} there exist many criteria. For a detailed overview, see Jolliffe [27]. A popular graphical technique is based on the *scree plot* which shows the eigenvalues in decreasing order. One then selects the index of the last component before the plot flattens. A more formal criterion considers the total variation which is explained by the first k loadings, and requires e.g. that

$$\left(\sum_{j=1}^{k_{\text{opt}}} l_j\right) / \left(\sum_{j=1}^{p} l_j\right) \geqslant 80\%. \tag{12}$$

Note that this criterion cannot be used with ROBPCA as the method does not yield all p eigenvalues. But it can be applied to the eigenvalues of the covariance matrix obtained after the reweighting step.

Another criterion that is based on the predictive ability of PCA is the PREdicted Sum of Squares (PRESS) statistic. To compute the (cross-validated) PRESS value at a certain k, the ith observation is removed from the original data set (for $i = 1, \ldots, n$), the center and the k loadings of the reduced data set are estimated, and then the fitted value of the ith observation is computed following (8) and denoted as $\hat{\boldsymbol{x}}_{-i,k}$. Finally, we set

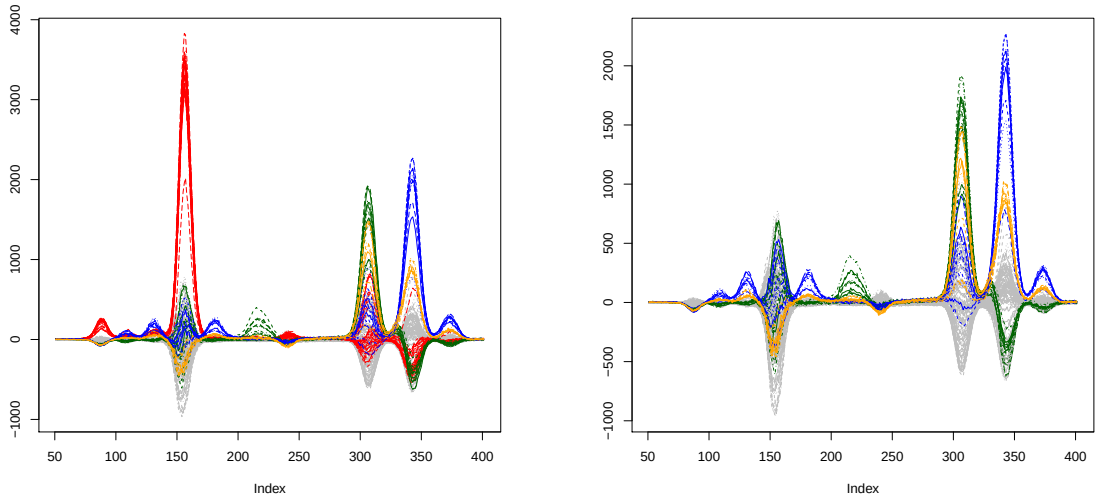
$$\text{PRESS}_k = \sum_{i=1}^{n} \|\boldsymbol{x}_i - \hat{\boldsymbol{x}}_{-i,k}\|^2. \tag{13}$$

The value k for which PRESS_k is small enough is then considered as the optimal number of components k_{opt}. One could also apply formal F-type tests based on successive PRESS values [28, 29].

The PRESS_k statistic is however not suited at contaminated data sets because it also includes the prediction error of the outliers. Even if the fitted values are based on a robust PCA algorithm, their prediction error might increase PRESS_k because they fit the model badly. To obtain a robust PRESS value, the following procedure can be applied. For each PCA model under investigation ($k = 1, \ldots, k_{\max}$), the outliers are marked. These are the observations that exceed the horizontal and/or the vertical cut-off value on the outlier map. Next, all the outliers are collected (over all k) and they are removed together from the sum in (13). Doing so, the robust PRESS_k value is based on the same set of observations for each k. Fast algorithms to compute such a robust PRESS value have been developed [30].

2.7 Example

We illustrate the PCA outlier map on a data set consisting of EPXMA spectra of $n = 180$ archeological glass pieces obtained using a Jeol JSM 6300 scanning electron microscope equipped with an energy-dispersive Si(Li) X-ray detection system (SEM-EDX) [31]. An electron beam current of 1nA and an accelerating voltage of 20 kV were applied, resulting in $p = 750$ variables corresponding with 750 energy values on the accelerator. The first 13 variables are removed as they are almost constant. The data are depicted in Figure 4. Here, we have subtracted the median at each energy value to better visualize the dispersion among the spectra. Only the indexes 50 to 400 are shown, as the others do contain far less information. The colored spectra correspond to groups of observations that are flagged in further analysis. Figure 4(a) contains all spectra, whereas Figure 4(b) only considers the first 142 spectra.



(a) (b)

Figure 4: Median-centered glass dataset showing (a) all 180 spectra; (b) the first 142 spectra.

Four principal components were retained for CPCA and ROBPCA, yielding a variance

explained percentage (12) of more than 99% for both methods.

The resulting outlier maps are shown in Figure 5. In Figure 5(a) we see that CPCA only finds some orthogonal outliers and some boundary cases. On the other hand the ROBPCA plot in Figure 5(b) clearly distinguishes two major groups in the data, as well as some smaller groups of bad and good leverage points. A high-breakdown method such as ROBPCA detects the group with cases 143–180 (colored red) as obvious bad leverage points. Later, it turned out that the window of the detector system had been cleaned before the last 38 spectra were measured. As a result less X-ray radiation was absorbed, resulting in higher X-ray intensities. This can also very well be seen in Figure 4(a) where the red spectra attain huge values around energy value 160. The green, blue and orange colored spectra are outlying in other regions. The blue spectra (57-63) and (73-76) are samples with a large concentration of calcic. The green ones (19-33) have mostly larger measurements at the energy values 215-245. This might indicate a larger concentration of phosphorus.

2.8 Sparse robust PCA

CPCA and all the robust PCA methods considered so far sometimes result in principal components that are difficult to interpret when most of the loadings are neither very small nor very large in absolute value. To increase interpretability, sparse PCA methods were developed to estimate components with many zero loading values.

A sparse version of the robust projection-pursuit techniques from Section 2.3 has been proposed in [32]. The subsequent eigenvectors $\boldsymbol{v}$ again maximize a robust scale of the scores, but now under the constraint that their L_1-norm $\|\boldsymbol{v}\|_1 = \sum_{j=1}^{p} |v_j|$ is smaller than a certain tuning parameter t. The smaller t becomes, the more many of the v_j will shrink towards zero and ultimately will become exactly zero.

In [33] the RObust Sparse PCA method ROSPCA is proposed, based on ROBPCA. Whereas ROBPCA applies classical PCA to the points with smallest outlyingness, ROSPCA applies the sparse classical PCA method ScoTLASS [34]. The selection of the sparsity parameter is based on a BIC-type criterion which combines a term that measures the quality of the fit with a term that penalizes for model complexity.

When we apply ROSPCA to the glass data, we obtain the outlier map in Figure 5(c). It is pretty similar to the ROBPCA results, which indicates that its outlier detection performance is still very high. On the other hand more sparse loadings have been obtained, as the number of nonzero loadings (that is, those v_j with $|v_j| > 10^{-5}$) is 358, 272, 490 and 408 for the four robust PCs.

2.9 Robust PCA with missing values, cellwise and rowwise outliers

All PCA methods described so far cannot handle missing values. If they occur, one could remove the entire row containing the missing element(s), but this could of course potentially

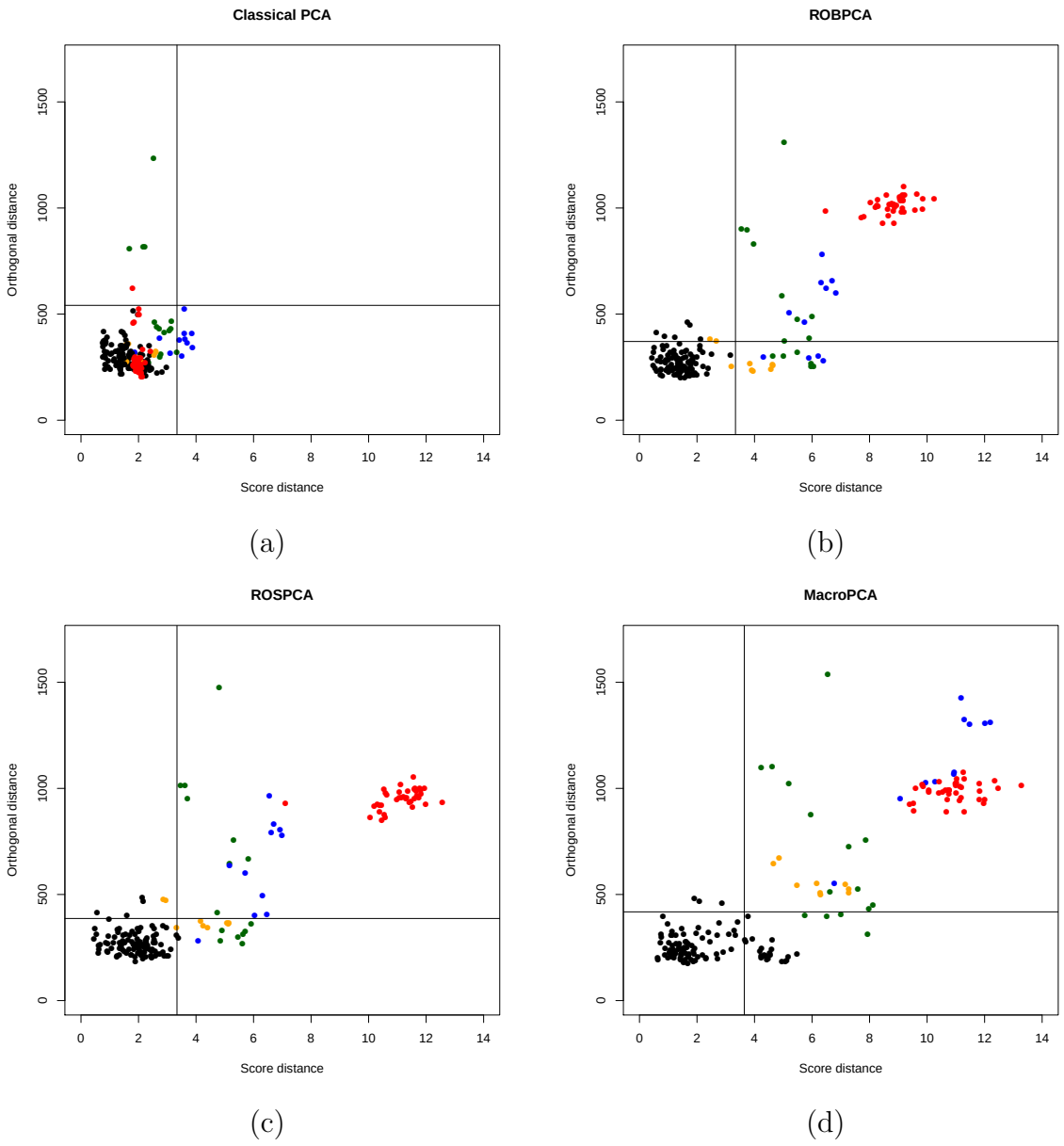


Figure 5: PCA outlier map of the glass dataset based on four principal components, computed with (a) CPCA; (b) ROBPCA; (c) ROSPCA; (d) MacroPCA.

result into a large loss of information, especially at high-dimensional data. For classical PCA, an EM-based iterative method ICPCA has been proposed, see e.g. [35]. It starts by replacing the missing values by initial estimates such as the columnwise means. Then it iteratively fits a CPCA, yielding scores that are transformed back to the original space resulting in new estimates for the missing values, until convergence. This approach has been adapted to ROBPCA, as outlined in [36].

Another issue for robust PCA methods is that they assume that at least half of the rows are completely outlier-free. Otherwise said, the number of outlying cases or *rowwise outliers*

should be at most $n/2$. This assumption might not be satisfied when many cells in the data matrix $\boldsymbol{X}$ are contaminated. These so-called *cellwise outliers* will occur more frequently in high-dimensional than in low-dimensional data.

Recently MacroPCA has been proposed as the first robust PCA method that can handle **M**issing values **A**nd **C**ellwise and **R**owwise **O**utliers simultaneously [37]. It starts by detecting the cellwise outliers by means of the Detect Deviating Cells (DDC) method, introduced in [38]. DDC also provides initial imputations for the outlying cells and the missing values as well as an initial measure of rowwise outlyingness. In the next steps MacroPCA combines ICPCA and ROBPCA to protect against rowwise outliers and to create improved imputations of the outlying cells and missing values. At the end of the algorithm this yields a robust center and a robust loading matrix, derived from the fully imputed data.

To flag the outliers, the NA-imputed data matrix $\mathring{\boldsymbol{X}}$ is considered in which only the missing elements are replaced by their imputed values. The orthogonal distance can then be computed as before:

$$\mathring{\text{OD}}_i = \|\mathring{\boldsymbol{x}}_i - \hat{\mathring{\boldsymbol{x}}}_i\|.$$

The MacroPCA outlier map of the glass data is shown in Figure 5(d). The different groups of outliers are again very visible, mostly as bad leverage points.

MacroPCA also produces a residual map. For each cell, the standardized residual is computed as $r_{ij} = (\mathring{x}_{ij} - \hat{\mathring{x}}_{ij})/s_j$ where s_j is a robust scale of the jth errors $\mathring{x}_{ij} - \hat{\mathring{x}}_{ij}$ for $i = 1, \ldots, n$. Cells with $|r_{ij}| \leqslant \sqrt{\chi^2_{1,0.99}} = 2.57$ are considered regular and colored yellow in the residual map, whereas the missing values are white. Outlying residuals receive a color which ranges from light orange to red when $r_{ij} > 2.57$ and from light purple to dark blue when $r_{ij} < -2.57$. So a dark red cell indicates that its observed value is much higher than its fitted value, while a dark blue cell means the opposite. Note that using by using the 0.99 quantile (instead of the 0.975 quantile used in other examples) we reduce the risk of flagging regular cells as outliers but increase the risk of not flagging a deviating cell. To the right of each row in the map is a circle whose color varies from white to black according to the orthogonal distance $\mathring{\text{OD}}_i$ compared to its cutoff (the horizontal line on the outlier map). Cases with a $\mathring{\text{OD}}_i$ below the cutoff lie close to the PCA subspace and receive a white circle. The others are given darker shades of gray up to black according to their $\mathring{\text{OD}}_i$.

Figure 6 shows the residual map of a subsample of the glass data. Note that the computations are performed on the whole dataset but we only display the results of samples 18-80. The black circles on the right side indicate that many of them have a large OD, which is in accordance with the outlier map in Figure 5(d). We can distinguish three groups of samples with different outlying behavior, which correspond nicely with the different colored observations (green, blue, orange) in the outlier map.

Figure 7 plots the residual map of the whole dataset. Here, the residuals are combined into blocks of 5×5 cells. The color of each block now depends on the most frequent type of outlying cell in it, the resulting color being an average. For example, an orange block indicates that quite a few cells in the block were red and most of the others were yellow.

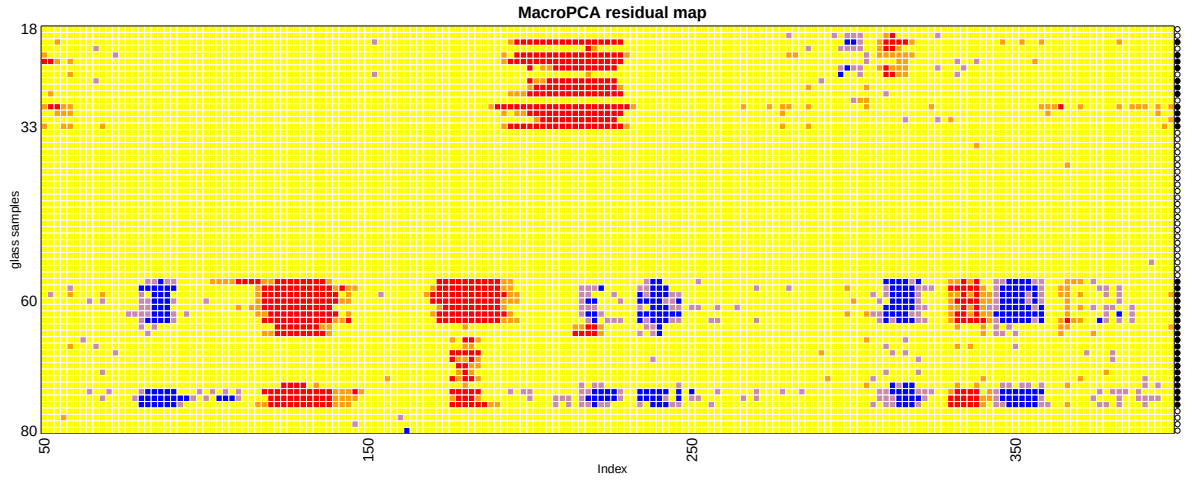


Figure 6: MacroPCA residual map of samples 18-80 from the glass dataset.

The more red cells in the block, the darker red the block will be. Now the last 38 samples clearly stand out.

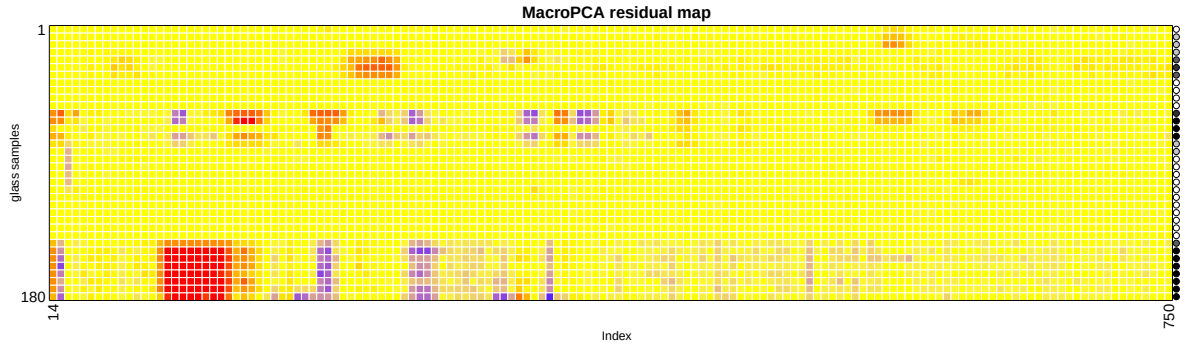


Figure 7: MacroPCA residual map of the complete glass dataset.

3 Linear calibration

3.1 Low-dimensional predictors

The multiple linear regression model assumes that in addition to the p predictor variables X_j, a response variable Y is measured, which can be well modeled as a linear combination of the X_j. More precisely, the model says that for all observations $(\boldsymbol{x}_i, y_i)$ it holds that

$$\begin{aligned} y_i &= \beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip} + \varepsilon_i \\ &= \beta_0 + \boldsymbol{\beta}^T \boldsymbol{x}_i + \varepsilon_i \end{aligned} \tag{14}$$

where the errors ε_i are assumed to be independent and identically distributed with zero mean and constant variance σ^2. The vector $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)^T$ is called the slope, and β_0 the intercept. We denote $\boldsymbol{x}_i = (x_{i1}, \ldots, x_{ip})^T$ and $\boldsymbol{\theta} = (\beta_0, \boldsymbol{\beta}^T)^T = (\beta_0, \beta_1, \ldots, \beta_p)^T$. Applying a regression estimator to the data yields $p+1$ regression coefficients $\hat{\boldsymbol{\theta}} = (\hat{\beta}_0, \ldots, \hat{\beta}_p)^T$. The residual r_i of case i is defined as the difference between the observed response y_i and its estimated value $\hat{y}_i$:

$$r_i(\hat{\boldsymbol{\theta}}) = y_i - \hat{y}_i = y_i - (\hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \ldots + \hat{\beta}_p x_{ip}).$$

The classical least squares (LS) method to estimate $\boldsymbol{\theta}$ minimizes the sum of the squared residuals. When $n \geqslant p+1 = \text{rank}(\boldsymbol{X})$, it equals $\hat{\boldsymbol{\theta}}_{\text{LS}} = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{y}$ (where the design matrix $\boldsymbol{X}$ is enlarged with a column of ones for the intercept term). However, the LS fit is extremely sensitive to regression outliers, which are observations that do not obey the linear pattern formed by the majority of the data. This is illustrated in Figure 8 for simple regression (where there is only one regressor x, or $p = 1$). It contains the Hertzsprung-Russell diagram of 47 stars, of which the logarithm of their light intensity and the logarithm of their surface temperature were measured [1]. The four most outlying observations are giant stars, which clearly deviate from the main sequence stars. The least squares fit in this plot was attracted by the giant stars.

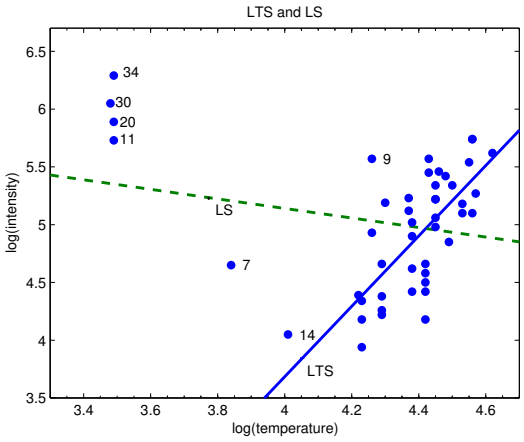
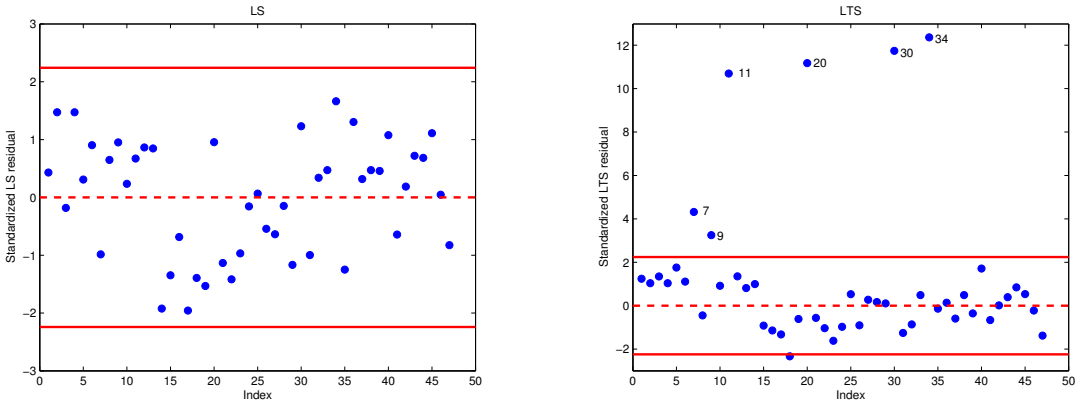


Figure 8: Stars regression data set with classical and robust fit.

An estimate of the variance of the error distribution σ^2 is given by $s^2 = \sum_{i=1}^{n} r_i^2/(n - p - 1)$. One often flags observations for which $|r_i/s|$ exceeds a cut-off like $\sqrt{\chi^2_{1,0.975}} = 2.24$ as regression outliers. In Figure 9(a) this strategy fails: the standardized least squares residuals of all 47 points lie inside the tolerance band between -2.24 and 2.24. The four outliers in Figure 8 have attracted the least squares line so much that they have small residuals r_i from it.

On Figure 8 a robust regression fit is superimposed. The *least trimmed squares estimator* (LTS) proposed in [6] is given by



(a) (b)

Figure 9: Standardized residuals of the stars data set, based on the (a) classical LS; (b) robust LTS estimator.

$$\text{minimize} \sum_{i=1}^{h} (r^2)_{(i)} \tag{15}$$

where $(r^2)_{(1)} \leqslant (r^2)_{(2)} \leqslant \cdots \leqslant (r^2)_{(n)}$ are the ordered squared residuals. (They are first squared, and then ranked.) The value h plays the same role as in the definition of the MCD estimator. For $h \approx n/2$ we find a breakdown value of 50%, whereas for larger h we obtain a breakdown value of $(n-h+1)/n$. A fast algorithm for the LTS estimator (FastLTS) has been developed [39]. The LTS estimator is regression, scale and affine equivariant [1] which essentially means that the estimate transforms correctly under affine transformations of the carriers and/or the response variable.

The scale of the errors σ can be estimated by $\hat{\sigma}^2_{\text{LTS}} = c^2_{h,n} \frac{1}{h} \sum_{i=1}^{h} (r^2)_{(i)}$ where r_i are the residuals from the LTS fit, and $c_{h,n}$ makes $\hat{\sigma}$ consistent and unbiased at Gaussian error distributions [40]. We can then identify regression outliers by their standardized LTS residuals $r_i/\hat{\sigma}_{\text{LTS}}$. This yields Figure 9(b) in which we clearly see the outliers. We can also use the standardized LTS residuals to assign a weight to every observation. The reweighed LS estimator with these LTS weights inherits the nice robustness properties of LTS, but is more efficient and yields all the usual inferential output such as t-statistics, F-statistics, an R^2 statistic, and the corresponding p-values.

Residuals plots become even more important in multiple regression with more than one regressor, as then we can no longer rely on a scatter plot of the data. Figure 9 however only allows us to detect observations that lie far away from the regression fit. It is also interesting to detect aberrant behavior in $\boldsymbol{X}$-space. Therefore a more informative outlier map can be constructed [41], plotting the standardized LTS residuals versus robust distances (5) based on (for example) the MCD estimator which is applied to the X_j-variables only.

Figure 10 shows the outlier map of the stars data. We can distinguish three types of

outliers. *Bad leverage points* lie far away from the regression fit and far away from the other observations in $\boldsymbol{X}$-space, e.g. the four giant stars and star 7. *Vertical outliers* have an outlying residual, but no outlying robust distance, e.g. star 9. Observation 14 is a *good leverage point*: it has an outlying robust distance, but it still follows the linear trend of the main sequence, since its absolute standardized residual does not exceed 2.24.

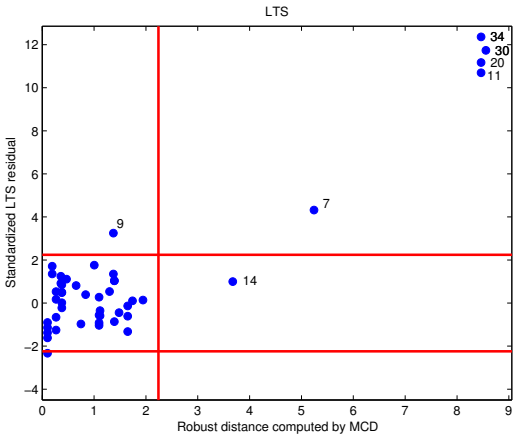


Figure 10: Outlier map for the stars data set.

The multiple regression model (14) can be extended to the case where we have more than one response variable. For p-variate predictors $\boldsymbol{x}_i = (x_{i1}, \ldots, x_{ip})^T$ and q-variate responses $\boldsymbol{y}_i = (y_{i1}, \ldots, y_{iq})^T$ the *multivariate (multiple) regression model* is given by

$$\boldsymbol{y}_i = \boldsymbol{\beta}_0 + \boldsymbol{\mathcal{B}}^T \boldsymbol{x}_i + \boldsymbol{\varepsilon}_i \tag{16}$$

where $\boldsymbol{\mathcal{B}}$ is the $p \times q$ slope matrix, $\boldsymbol{\beta}_0$ is the q-dimensional intercept vector, and the errors $\boldsymbol{\varepsilon}_i = (\varepsilon_{i1}, \ldots, \varepsilon_{iq})^T$ are i.i.d. with zero mean and with $\mathrm{Cov}(\boldsymbol{\varepsilon}) = \boldsymbol{\Sigma}_\varepsilon$ a positive definite matrix of size q. The least squares solution can be written as

$$\hat{\boldsymbol{\mathcal{B}}} = \hat{\boldsymbol{\Sigma}}_x^{-1} \hat{\boldsymbol{\Sigma}}_{xy} \tag{17}$$

$$\hat{\boldsymbol{\beta}}_0 = \hat{\boldsymbol{\mu}}_y - \hat{\boldsymbol{\mathcal{B}}}^T \hat{\boldsymbol{\mu}}_x \tag{18}$$

$$\hat{\boldsymbol{\Sigma}}_\varepsilon = \hat{\boldsymbol{\Sigma}}_y - \hat{\boldsymbol{\mathcal{B}}}^T \hat{\boldsymbol{\Sigma}}_x \hat{\boldsymbol{\mathcal{B}}} \tag{19}$$

where

$$\hat{\boldsymbol{\mu}} = \begin{pmatrix} \hat{\boldsymbol{\mu}}_x \\ \hat{\boldsymbol{\mu}}_y \end{pmatrix} \qquad \text{and} \qquad \hat{\boldsymbol{\Sigma}} = \begin{pmatrix} \hat{\boldsymbol{\Sigma}}_x & \hat{\boldsymbol{\Sigma}}_{xy} \\ \hat{\boldsymbol{\Sigma}}_{yx} & \hat{\boldsymbol{\Sigma}}_y \end{pmatrix} \tag{20}$$

are the empirical mean and covariance matrix of the joint (X, Y)-variables.

In [42] it is proposed to fill in the MCD estimates for the center $\boldsymbol{\mu}$ and the scatter matrix $\boldsymbol{\Sigma}$ of the joint (X, Y)-variables in (20), yielding robust estimates (17) to (19). The resulting estimates are called MCD-regression estimates. They inherit the high breakdown value of

the MCD estimator. To obtain a better efficiency, the reweighed MCD estimates are used in (17)-(19) and followed by a regression reweighing step. For any fit $\hat{\boldsymbol{\theta}} = (\hat{\boldsymbol{\beta}}_0, \hat{\boldsymbol{\mathcal{B}}}^T)^T$, denote the corresponding q-dimensional residuals by $\boldsymbol{r}_i(\hat{\boldsymbol{\theta}}) = \boldsymbol{y}_i - \hat{\boldsymbol{\mathcal{B}}}^T\boldsymbol{x}_i - \hat{\boldsymbol{\beta}}_0$. Then the *residual distance* of the ith case is defined as

$$\mathrm{ResD}_i = \sqrt{\boldsymbol{r}_i^T \hat{\boldsymbol{\Sigma}}_{\boldsymbol{\varepsilon}}^{-1} \boldsymbol{r}_i}. \tag{21}$$

These residual distances can then be used in a reweighing step in order to improve the efficiency.

We refer to the chapter on 'Robust multivariate statistical methods' for more details and other examples of robust estimators for low-dimensional data.

3.2 High-dimensional predictors

3.2.1 PCR and PLSR

When the number of independent variables p in a regression model is very large or when the regressors are highly correlated (this is known as *multicollinearity*), traditional regression methods such as ordinary least squares tend to fail.

An important example in chemometrics is multivariate calibration whose goal is to predict constituent concentrations of a material (or other properties such as hardness or density or biological activity) based on its spectrum (or e.g. chromatograms or concentrations of metabolites or elements). 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. The more general problem assumes that the number of response variables q is larger than one, which means that several concentrations are to be estimated together. This model 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 intercorrelated with each other. As argued in Martens and Naes [43] 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, multivariate calibration can be very important in order to detect outlying samples which would not be discovered by separate regressions. Here, we will write down the formulas for the general multivariate setting (16) for which $q \geqslant 1$, but they can of course be simplified when $q = 1$.

Principal Component Regression (PCR) and Partial Least Squares Regression (PLSR) are two methods frequently used to build regression models in very high dimensions. Both assume that the linear relation (16) between the X- and Y-variables is a bilinear model

which depends on k-dimensional scores $\boldsymbol{t}_i$:

$$\boldsymbol{x}_i = \boldsymbol{\mu}_x + \boldsymbol{\mathcal{P}}_{p,k}\boldsymbol{t}_i + \boldsymbol{\varepsilon}_i \tag{22}$$

$$\boldsymbol{y}_i = \boldsymbol{\mu}_y + \boldsymbol{\mathcal{A}}_{q,k}\boldsymbol{t}_i + \boldsymbol{\varepsilon}_i'. \tag{23}$$

Typically k is taken much smaller than p. Consequently, both PCR and PLSR first try to estimate the scores $\boldsymbol{t}_i$. Then a traditional regression can be used to regress the response $\boldsymbol{y}$ onto these low-dimensional scores. In order to obtain the scores, one can perform PCA on the independent variables. This is the idea behind PCR. In this case no information about the response variable is used when reducing the dimension. Therefore PLSR is sometimes more appropriate, as this method estimates the scores maximizing a covariance criterion between the independent and dependent variable. In any case, the original versions of both PCR and PLSR strongly rely on CPCA and LS, making them very sensitive to outliers.

3.2.2 Robust PCR

A robust PCR method (RPCR) was proposed in [44]. In the first stage of the algorithm, robust scores $\boldsymbol{t}_i$ are obtained by applying ROBPCA to the X-variables and retaining k components. In the second stage of RPCR, the original response variables $\boldsymbol{y}_i$ are regressed on the $\boldsymbol{t}_i$ using a robust regression method. If there is only one response variable ($q = 1$), the reweighed LTS estimator is applied. If $q > 1$ MCD-regression is performed. Note that 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, it is recommended to use the same value in both steps. Using all robust distances in play (the orthogonal distances (9), the score distances (10) and the residual distances (21)) one can again construct outlier maps to visualize outliers and classify observations as regular observations, PCA outliers or regression outliers.

3.2.3 Robust PLSR

In PLSR the estimation of the scores is a little bit more involved as it also includes information about the response variable. Let $\tilde{\boldsymbol{X}}_{n,p}$ and $\tilde{\boldsymbol{Y}}_{n,q}$ denote the mean-centered data matrices. The normalized PLS weight vectors $\boldsymbol{r}_a$ and $\boldsymbol{q}_a$ (with $\|\boldsymbol{r}_a\| = \|\boldsymbol{q}_a\| = 1$) are then defined as the vectors that maximize

$$\mathrm{cov}(\tilde{\boldsymbol{Y}}\boldsymbol{q}_a, \tilde{\boldsymbol{X}}\boldsymbol{r}_a) = \boldsymbol{q}_a^T \frac{\tilde{\boldsymbol{Y}}^T\tilde{\boldsymbol{X}}}{n-1}\boldsymbol{r}_a = \boldsymbol{q}_a^T \boldsymbol{S}_{yx}\boldsymbol{r}_a \tag{24}$$

for each $a = 1, \ldots, k$, where $\boldsymbol{S}_{yx}^T = \boldsymbol{S}_{xy} = \frac{\tilde{\boldsymbol{X}}^T\tilde{\boldsymbol{Y}}}{n-1}$ is the empirical cross-covariance matrix between the X- and the Y-variables. The elements of the scores $\boldsymbol{t}_i$ are then defined as linear combinations of the mean-centered data: $t_{ia} = \tilde{\boldsymbol{x}}_i^T\boldsymbol{r}_a$, or equivalently $\boldsymbol{T}_{n,k} = \tilde{\boldsymbol{X}}_{n,p}\boldsymbol{R}_{p,k}$ with $\boldsymbol{R}_{p,k} = (\boldsymbol{r}_1, \ldots, \boldsymbol{r}_k)$.

The computation of the PLS weight vectors can be performed using the SIMPLS algorithm [45]. The solution of the maximization problem (24) is found by taking $\boldsymbol{r}_1$ and $\boldsymbol{q}_1$ as

the first left and right singular eigenvectors of $\boldsymbol{S}_{xy}$. The other PLSR weight vectors $\boldsymbol{r}_a$ and $\boldsymbol{q}_a$ for $a = 2, \ldots, k$ are obtained by imposing an orthogonality constraint to the elements of the scores. If we require that $\sum_{i=1}^{n} t_{ia} t_{ib} = 0$ for $a \neq b$, a deflation of the cross-covariance matrix $\boldsymbol{S}_{xy}$ provides the solutions for the other PLSR weight vectors. This deflation is carried out by first calculating the $\boldsymbol{x}$-loading

$$\boldsymbol{p}_a = \boldsymbol{S}_x \boldsymbol{r}_a / (\boldsymbol{r}_a^T \boldsymbol{S}_x \boldsymbol{r}_a) \tag{25}$$

with $\boldsymbol{S}_x$ the empirical covariance matrix of the X-variables. Next an orthonormal base $\{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_a\}$ of $\{\boldsymbol{p}_1, \ldots, \boldsymbol{p}_a\}$ is constructed and $\boldsymbol{S}_{xy}$ is deflated as

$$\boldsymbol{S}_{xy}^a = \boldsymbol{S}_{xy}^{a-1} - \boldsymbol{v}_a (\boldsymbol{v}_a^T \boldsymbol{S}_{xy}^{a-1})$$

with $\boldsymbol{S}_{xy}^1 = \boldsymbol{S}_{xy}$. In general the PLSR weight vectors $\boldsymbol{r}_a$ and $\boldsymbol{q}_a$ are obtained as the left and right singular vector of $\boldsymbol{S}_{xy}^a$.

A robust method RSIMPLS has been developed in [46]. It starts by applying ROBPCA to the joint (X, Y)-variables in order to replace $\boldsymbol{S}_{xy}$ and $\boldsymbol{S}_x$ by robust estimates, and then proceeds analogously to the SIMPLS algorithm. More precisely, to obtain robust scores, ROBPCA is first applied to $\boldsymbol{Z}_{n,m} = (\boldsymbol{X}_{n,p}, \boldsymbol{Y}_{n,q})$ with $m = p + q$. Assume we select k_0 components. This yields a robust estimate $\hat{\boldsymbol{\mu}}_z$ of the center of $\boldsymbol{Z}$, and following (11) an estimate $\hat{\boldsymbol{\Sigma}}_z$ of its shape. These estimates can then be split into blocks, just like (20). The cross-covariance matrix $\boldsymbol{\Sigma}_{xy}$ is then estimated by $\hat{\boldsymbol{\Sigma}}_{xy}$ and the PLS weight vectors $\boldsymbol{r}_a$ are computed as in the SIMPLS algorithm, but now starting from $\hat{\boldsymbol{\Sigma}}_{xy}$ instead of $\boldsymbol{S}_{xy}$. In analogy with (25) the $\boldsymbol{x}$-loadings $\boldsymbol{p}_j$ are defined as $\boldsymbol{p}_j = \hat{\boldsymbol{\Sigma}}_x \boldsymbol{r}_j / (\boldsymbol{r}_j^T \hat{\boldsymbol{\Sigma}}_x \boldsymbol{r}_j)$. Then the deflation of the scatter matrix $\hat{\boldsymbol{\Sigma}}_{xy}^a$ is performed as in SIMPLS. In each step, the robust scores are calculated as $t_{ia} = \breve{\boldsymbol{x}}_i^T \boldsymbol{r}_a = (\boldsymbol{x}_i - \hat{\boldsymbol{\mu}}_x)^T \boldsymbol{r}_a$ where $\breve{\boldsymbol{x}}_i = \boldsymbol{x}_i - \hat{\boldsymbol{\mu}}_x$ are the robustly centered observations.

Next, we need a robust regression of $\boldsymbol{y}_i$ on $\boldsymbol{t}_i$. This could again be done using MCD-regression. A faster approach is also possible [46], by explicitly making use of the prior information given by ROBPCA in the first step of the algorithm.

This RSIMPLS approach yields bounded influence functions for the weight vectors $\boldsymbol{r}_a$ and $\boldsymbol{q}_a$ and for the regression estimates [47]. Also the breakdown value is inherited from the MCD estimator.

Another robustification of PLSR has been proposed in [48]. A reweighing scheme is introduced based on ordinary PLSR, leading to a fast and robust procedure. The algorithm can however only deal with the univariate case ($q = 1$).

3.2.4 Model calibration and validation

An important issue in PCR and PLSR is the selection of the optimal number of scores k_{opt}. A traditional approach consists of minimizing the root mean squared error of cross-validation criterion RMSECV_k, defined as

$$\text{RMSECV}_k = \sqrt{\frac{1}{n} \sum_{i=1}^{n} \|\boldsymbol{y}_i - \hat{\boldsymbol{y}}_{-i,k}\|^2} \tag{26}$$

with $\hat{\boldsymbol{y}}_{-i,k}$ the cross-validated prediction of $\boldsymbol{y}_i$ based on k scores. The goal of the RMSECV_k statistic is twofold. It yields an estimate of the root mean squared prediction error when k components are used in the model, whereas the curve of RMSECV_k for $k = 1, \ldots, k_{\max}$ is a graphical tool to choose the optimal number of scores.

As argued for the PRESS statistic (13) in PCA, also this RMSECV_k statistic is not suited for contaminated data sets because it includes the prediction error of the outliers. A robust RMSECV (R-RMSECV) measure was constructed in [49] by omitting the outliers from the sum in (26). In a naive algorithm this approach would of course be extremely time consuming, since we have to run the entire RPCR or RSIMPLS algorithm n times (each deleting another observation in the cross-validation) for every possible choice of k. In [49] a faster algorithm was proposed, which efficiently reuses results and information from previous runs. The R-RMSECV_k criterion is a robust measure of how well the model predicts the response for *new* observations. If we want to see how well the model fits the *given* observations, we can define a very similar goodness-of-fit criterion. The root mean squared error (RMSE_k) is calculated by replacing $\hat{\boldsymbol{y}}_{-i,k}$ in (26) by the fitted value $\hat{\boldsymbol{y}}_{i,k}$ obtained using all observations including the ith one. As for R-RMSECV_k, a robust R-RMSE_k does not include the outliers to compute the average squared error. Finally, a Robust Component Selection (RCS) statistic is defined [49] by

$$\text{RCS}_k = \sqrt{\gamma\, \text{R-RMSECV}_k^2 + (1-\gamma)\, \text{R-RMSE}_k^2}$$

with a tuning parameter $\gamma \in [0, 1]$. If the user selects a small γ, then the goodness-of-fit becomes the most important term. Choosing γ close to one on the other hand emphasizes the importance of the quality of predictions. If the user has no a priori preference, $\gamma = 0.5$ can be selected in order to give equal weight to both terms. Finally, a plot of RCS_k versus k offers an easy way of visually selecting the most appropriate k.

Another approach is presented in [50]. Here, different robust estimators of the Root Mean Squared Error of Prediction (RMSEP) are proposed, and the optimal number of components is selected by means of a desirability index. Also different ways to robustly preprocess the data are considered in that paper.

3.3 Examples

We first illustrate the robustness of RSIMPLS on the octane data set [51] consisting of NIR absorbance spectra over $p = 226$ wavelengths ranging from 1102nm to 1552nm with measurements every two nm. For each of the $n = 39$ production gasoline samples the octane number y was measured, so $q = 1$. It is known that the octane data set contains six outliers (25, 26, 36–39) to which alcohol was added.

Figure 11 shows the RCS curves for γ equal to 0, 0.5 and 1. Choosing $k = 1$ is clearly a bad idea, since the prediction error is very high in that case. From two components on, the RCS curve becomes rather stable. For $\gamma \geqslant 0.5$, the minimal error is attained at $k = 6$,

which would be a good choice. The difference with $k = 2$ is however not very large, so for the sake of simplicity we decided to retain $k = 2$ components.

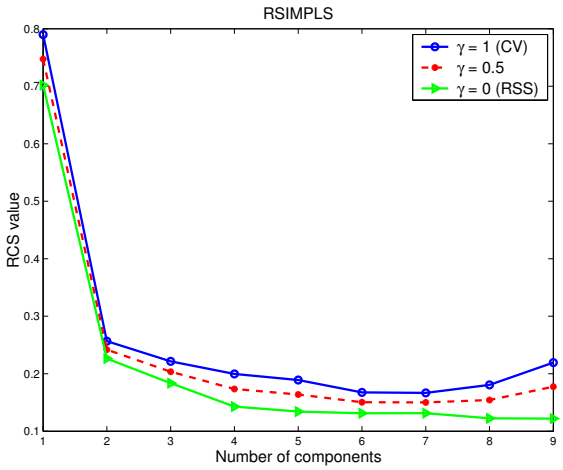


Figure 11: RCS curve for the octane data set.

The resulting outlier maps are shown in Figure 12. The robust PCA outlier map is displayed in Figure 12(b). According to model (22), it displays the score distance $\mathrm{SD}_i = \sqrt{(\boldsymbol{t}_i - \hat{\boldsymbol{\mu}}_t)^T \hat{\boldsymbol{\Sigma}}_t^{-1} (\boldsymbol{t}_i - \hat{\boldsymbol{\mu}}_t)}$ on the horizontal axis, where $\hat{\boldsymbol{\mu}}_t$ and $\hat{\boldsymbol{\Sigma}}_t$ are derived in the regression step of the RSIMPLS algorithm. On the vertical axis it shows the orthogonal distance of the observation to the t-space, so $\mathrm{OD}_i = \|\boldsymbol{x}_i - \hat{\boldsymbol{\mu}}_x - \boldsymbol{P}_{p,k}\boldsymbol{t}_i\|$. We immediately spot the six samples with added alcohol. The SIMPLS outlier map is shown in Figure 12(a). We see that this analysis only detects the outlying spectrum 26, which does not even stick out much above the border line. The robust regression outlier map in Figure 12(d) shows that the outliers are good leverage points, whereas SIMPLS in Figure 12(c) again only reveals case 26.

To illustrate RPCR we analyze the Biscuit Dough data set [52]. It contains 40 NIR spectra of biscuit dough with measurements every two nm, from 1200nm up to 2400nm. After some preprocessing (as described in [13]) we end up with a data set of $n = 40$ observations in $p = 600$ dimensions. The responses are the percentages of four 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 responses, a multivariate regression is performed. The RCS curve is plotted in Figure 13 and suggests to select $k = 2$ components. Note that here $k_{max} = 4$, due to the large number of parameters in the multivariate regression model, see [46] for details.

Next, we can construct outlier maps. ROBPCA yields the PCA outlier map displayed in Figure 14(a). We see that there are no PCA leverage points but there are some orthogonal outliers, the largest being 23, 7 and 20. The result of the regression step is shown in Figure 14(b). It exposes the robust distances of the residuals versus the score distances. RPCR shows that observation 21 has an extremely high residual distance. Other vertical outliers are 23, 7, 20, and 24, whereas there are a few borderline cases. In [44] it is demonstrated that

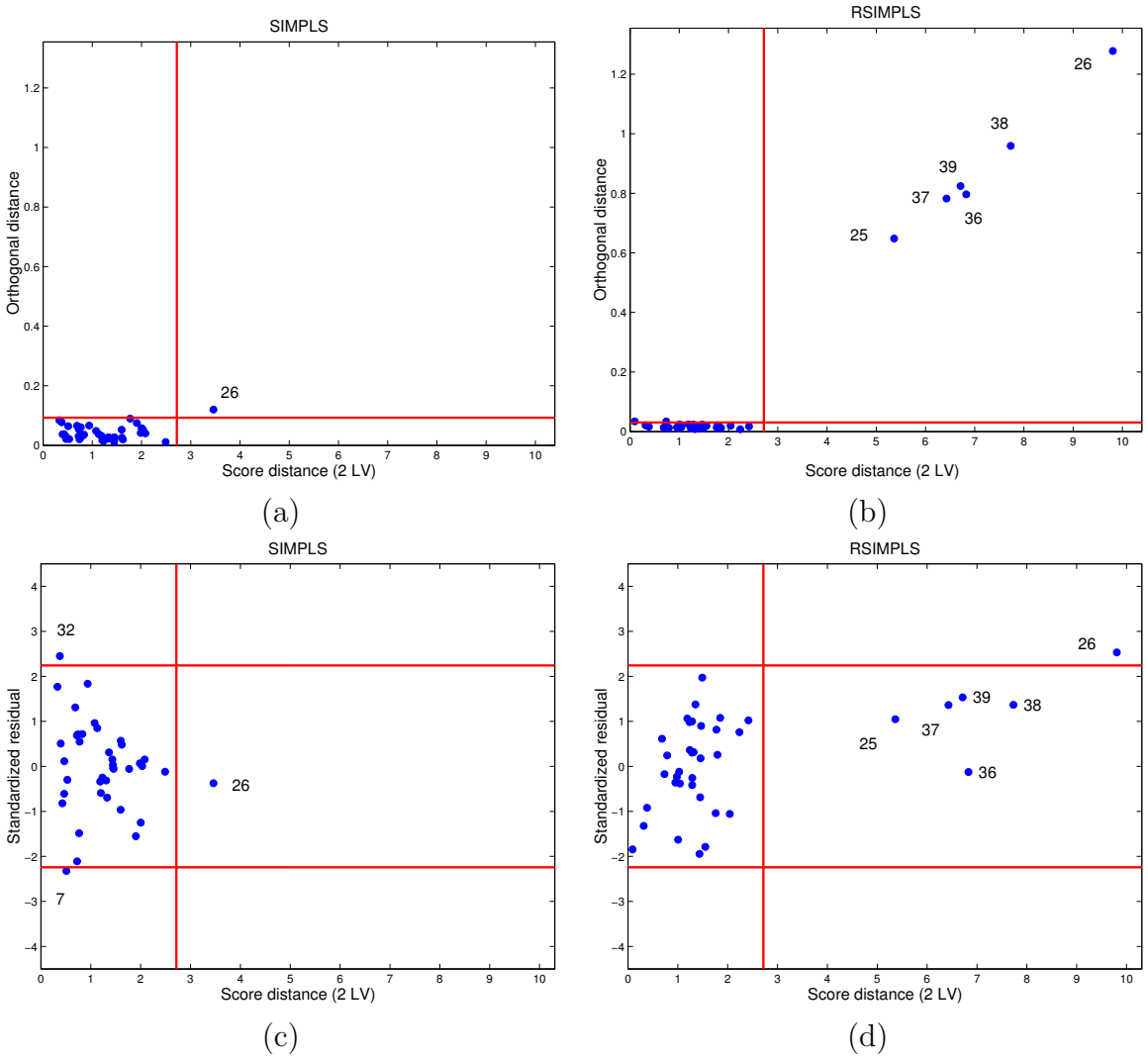


Figure 12: (a) PCA outlier map of the octane data set obtained with SIMPLS; (b) with RSIMPLS; (c) Regression outlier map obtained with SIMPLS; (d) with RSIMPLS.

case 21 never showed up as such a large outlier when performing four univariate calibrations. It is only by using the full covariance structure of the residuals in (21) that this extreme data point is spotted.

3.3.1 Other approaches

In literature, several other approaches for robust PCR and PLSR have been proposed, among which [53, 54, 48]. However, most methods lack some equivariance properties, are less robust, or they can not be applied to multivariate responses ($q > 1$). See e.g. [55] for a nice overview and discussion.

PCR and PLS have been extended to continuum regression [56]. A robust version, based on the projection-pursuit idea, has been proposed in [57], and is applicable when $q = 1$.

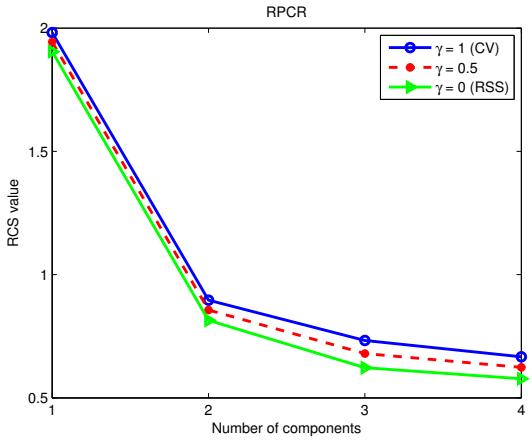
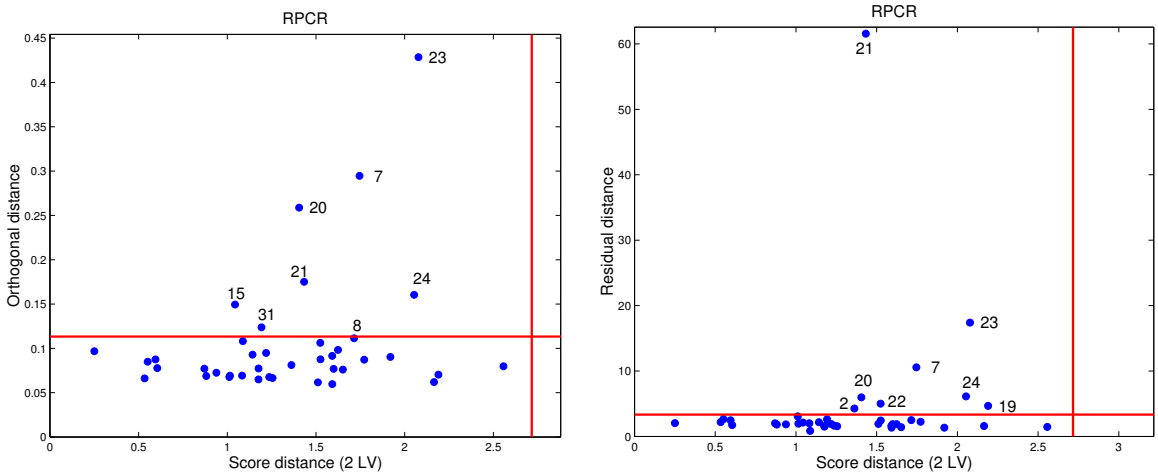


Figure 13: RCS curve for the Biscuit Dough data set.



(a) (b)

Figure 14: (a) PCA outlier map of the biscuit dough data set obtained with RPCR; (b) regression outlier map obtained with RPCR.

4 Classification

4.1 SIMCA and RSIMCA

Classification (also known as discriminant analysis) has many applications in chemometrics, see e.g. [58]. Robust approaches for low-dimensional data based on the MCD and S-estimators have been studied by several authors, see e.g. [59, 60, 61].

For high-dimensional data, a robust version of the SIMCA method based on the ROBPCA estimator, has been proposed in [62]. Suppose that we have l groups with p-dimensional data matrices $\boldsymbol{X}^j$, $j = 1, \ldots, l$. Denote n_j the number of observations in group j. The SIMCA method starts by performing PCA on each group $\boldsymbol{X}^j$ separately. Let k_j denote the number of

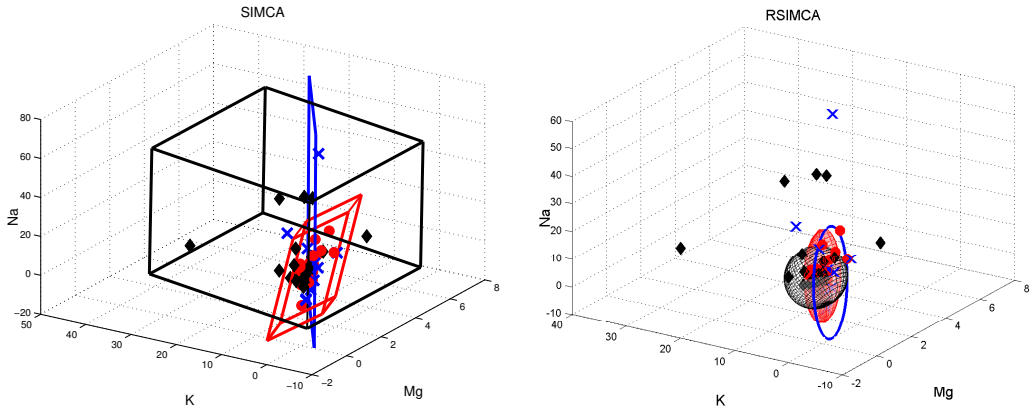
retained principal components in group j. New observations are then classified by means of their distances to the different PCA models. The choice of an appropriate distance however is a difficult task. A first idea is to use the orthogonal distances obtained from the PCA analysis, cfr. (9). Denote $\mathrm{OD}^{(j)}$ the orthogonal distance from a new observation $\boldsymbol{x}$ to the PCA hyperplane for the jth group. Denote OD_i^j the orthogonal distance from the ith observation in group j to the PCA hyperplane for the jth group. Then for j ranging from 1 to l, an F-test is performed with test statistic $(s^{(j)}/s_j)^2$ where

$$(s^{(j)})^2 = \frac{(\mathrm{OD}^{(j)})^2}{p-k_j} \qquad \text{and} \qquad s_j^2 = \frac{\sum_{i=1}^{n_j}(\mathrm{OD}_i^j)^2}{(p-k_j)(n_j-k_j-1)}.$$

If the observed F-value is smaller than the critical value, $\boldsymbol{x}$ is said to belong to group j. Note that an observation can be classified in many different groups, hence the term *Soft* in SIMCA.

This approach based on the orthogonal distances only, turned out not to be completely satisfactory. To fully exploit applying PCA in each group separately, it was suggested to include the score distances (10) as well. They can be used to construct a multidimensional box around the jth PCA model. In Figure 15(a) these boxes are plotted for a three-dimensional example with three groups. A boundary distance $\mathrm{BD}^{(j)}$ is then defined as the distance of a new observation $\boldsymbol{x}$ to the box for the jth group. Assigning $\boldsymbol{x}$ to any of the l classes is then done by means of an F-test based on a linear combination of $(\mathrm{BD}^{(j)})^2$ and $(\mathrm{OD}^{(j)})^2$.

A first step in robustifying SIMCA can be obtained by applying a robust PCA method, such as ROBPCA to each group. The number of components in each group can e.g. be selected by robust cross-validation, as explained in Section 2.6. Also the classification rule needs to be changed, since the SIMCA-boxes are defined in such a way that they contain all the observations. When outliers are present these boxes can thus be highly inflated, as



(a) (b)

Figure 15: (a) Boxes based on classical PCA, which are blown up by some outliers; (b) ellipsoids based on robust PCA.

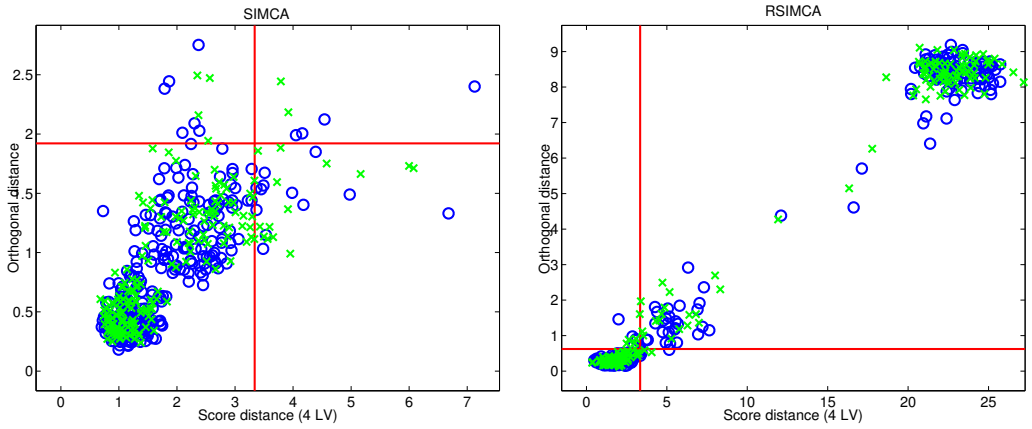
demonstrated in Figure 15(a). Figure 15(b) shows robust ellipsoids of the groups [62]. A

new observation $\boldsymbol{x}$ is classified in group m if

$$\gamma\left(\frac{\mathrm{OD}_j(\boldsymbol{x})}{c_j^v}\right)^2 + (1-\gamma)\left(\frac{\mathrm{SD}_j(\boldsymbol{x})}{c_j^h}\right)^2$$

is smallest for $j = m$, where OD (resp. SD) now denotes the orthogonal (resp. score) distance to a robust PCA model. The numbers c_j^v and c_j^h are carefully chosen normalizers. The tuning parameter $0 \leqslant \gamma \leqslant 1$ is added for two reasons. If the user a priori judges that the OD (resp. the SD) is the most important criterion to build the classifier, the parameter γ can be chosen close to one (resp. zero). Otherwise, γ can be selected such that the misclassification probability is minimized. This probability can be estimated by means of a validation set or cross-validation.

Also in this setting, outlier maps are helpful graphical tools to gain more insight in the data. We illustrate RSIMCA on the fruit data set [61]. It contains the spectra of three different cultivars of a melon. The cultivars (named D, M and HA) have sizes 490, 106 and 500, and all spectra are measured in 256 wavelengths. The RSIMCA classification rules were derived on a training set, consisting of a random subset of 60% of the samples. The other samples were assigned to the validation set. Figure 16(b) shows the outlier map of cultivar HA. It plots the (SD_i,OD_i) for all 500 observations, using circles for the training data and crosses for the validation data. We immediately detect a large group of outliers. As it turns out, these outliers were caused by a change in the illumination system. Figure 16(a) depicts the corresponding outlier map using the classical SIMCA results. In this case the outliers remain undetected, another example of the masking effect that non-robust methods can suffer from.



(a) (b)

Figure 16: Fruit data, cultivar HA. (a) Outlier map for classical SIMCA; (b) outlier map for RSIMCA.

A different classifier for multivariate, high-dimensional and functional data is presented in [63]. The approach is based on a kind of robust distance between data points and groups.

Then each data point is mapped to the vector of its distances to all groups, followed by k-nearest neighbor (kNN) classification of the transformed data points.

4.2 Support Vector Machines

For non-linear modeling, Support Vector Machines (SVM) are very powerful. Excellent introductory material on this subject can be found in [64, 65, 66]. In [67] a short summary is given. Support Vector Machines are defined by means of a kernel function. The most common ones are linear kernels for linear classification, polynomial kernels for polynomial decision boundaries (e.g. a parabola), and Gaussian kernels for general nonlinear, semiparametric decision boundaries. In recent years some interesting results were obtained with respect to their robustness. In [68] the influence function of a broad class of kernel classifiers was investigated. It was proven that the robustness properties of SVM classification strongly depend on the choice of the kernel. For an unbounded kernel, e.g. a linear kernel, the resulting SVM methods are not robust and suffer the same problems as traditional linear classifiers (such as the classical SIMCA method). However, when a bounded kernel is used, such as the Gaussian kernel, the resulting non-linear SVM classification handles outliers quite well. Also Support Vector Data Description [69] can be used to detect outliers.

SVMs are also used to handle non-linear regression in high dimensions. Within the chemometrical literature, the Pearson Universal Kernel was introduced in [70] and good results were reported. The robustness of these kernel-based regression methods was investigated in [71]. As in classification, the kernel plays an important role. A linear kernel leads to non-robust methods whereas a bounded kernel (e.g. the Gaussian kernel) leads to quite robust methods with respect to outliers in $\boldsymbol{x}$-space. In order to reduce the effect of vertical outliers, one should choose a loss function with a bounded derivative. For Least Squares SVMs [72] robustness can be achieved by some reweighting steps [73, 74].

5 Multi-way analysis

Many experiments give rise to more complex data structures, where different sets of variables are measured at the same time. It is shown in e.g. [75] that preserving the nature of the data set by arranging it in a higher-dimensional tensor, instead of forcing it into a matrix, leads to a better understanding and more precise models. Such complex data structures are called *multi-way* data sets. Different techniques exist to analyze these multi-way arrays, among which PARAFAC and Tucker3 are the most popular ones. As they aim at constructing scores and loadings to express the data in a more comprehensive way, PARAFAC and Tucker3 can be seen as generalizations of PCA to higher-order tensors [76].

Three-way data $\underline{\boldsymbol{X}}$ contain I observations that are measured for J and K variables. PARAFAC decomposes the data into trilinear components. The structural model can be

described as

$$x_{ijk} = \sum_{f=1}^{F} a_{if} b_{jf} c_{kf} + e_{ijk} \tag{27}$$

where a_{if}, b_{jf} and c_{kf} are parameters describing the importance of the samples/variables to each component. The residual e_{ijk} contains the variation not captured by the PARAFAC model. In terms of the unfolded matrix $\boldsymbol{X}^{I\times JK}$, this model can also be written as

$$\boldsymbol{X}^{I\times JK} = \boldsymbol{A}(\boldsymbol{C} \odot \boldsymbol{B})^T + \boldsymbol{E} \tag{28}$$

with $\boldsymbol{A}$ an $(I \times F)$-matrix of scores, $\boldsymbol{B}$ a $(J \times F)$-matrix of B-loadings, and $\boldsymbol{C}$ a $(K \times F)$-matrix of C-loadings. The number F stands for the number of factors to include in the model, $\boldsymbol{E}$ is the error term and $\odot$ is the Kathri-Rao product, which is defined by $\boldsymbol{C} \odot \boldsymbol{B} = [vec(\boldsymbol{b}_1\boldsymbol{c}_1^T), \ldots, vec(\boldsymbol{b}_F\boldsymbol{c}_F^T)]$. The *vec* operator yields the vector obtained by unfolding a matrix column-wise to one column. The PARAFAC model is for example used to decompose fluorescence data into tri-linear components according to the number of fluorophores (F) present in the samples. The observed value x_{ijk} then corresponds to the intensity of sample i at emission wavelength j and excitation wavelength k [77, 78, 79].

The scores and loadings of the PARAFAC model are estimated by minimizing the objective function

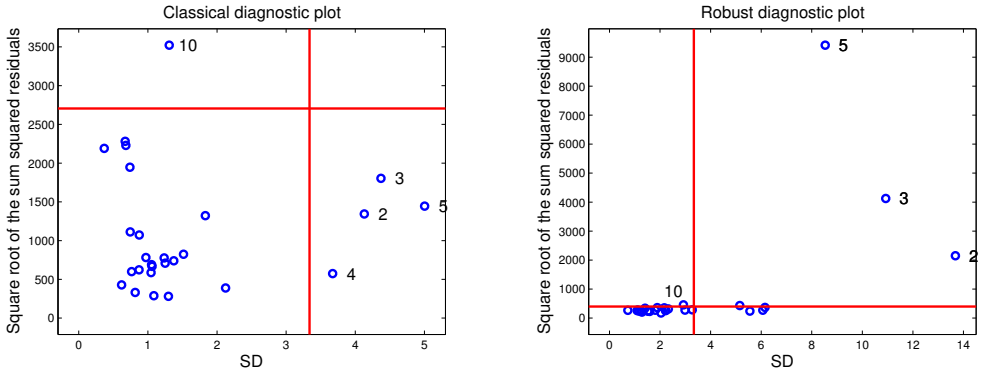
$$\sum_{i=1}^{I}\sum_{j=1}^{J}\sum_{k=1}^{K}(x_{ijk} - \hat{x}_{ijk})^2 = \|\boldsymbol{X} - \hat{\boldsymbol{X}}\|_F^2 = \|\boldsymbol{X} - \hat{\boldsymbol{A}}(\hat{\boldsymbol{C}} \odot \hat{\boldsymbol{B}})^T\|_F^2. \tag{29}$$

An algorithm based on alternating Least Squares (ALS) is typically used for this purpose. This means that given initial estimates for $\boldsymbol{B}$ and $\boldsymbol{C}$, $\boldsymbol{A}$ is estimated conditionally on $\boldsymbol{B}$ and $\boldsymbol{C}$ by minimizing (29). If we define $\boldsymbol{Z} = \boldsymbol{C} \odot \boldsymbol{B}$ the optimization problem can be reduced to minimizing $\|\boldsymbol{X} - \boldsymbol{A}\boldsymbol{Z}^T\|_F^2$, which gives rise to the classical least squares regression problem. It is obvious that the model will be highly influenced by outliers, as a least squares minimization procedure is used.

To cope with outlying samples, a robust counterpart for PARAFAC has been constructed [80]. The procedure starts by looking for $\frac{I}{2} < h < I$ points that minimize the objective function (29). The value of h plays the same role as in the MCD estimator and the LTS estimator. To find this optimal h-subset, ROBPCA is applied to the unfolded data $\boldsymbol{X}^{I\times JK}$ and the h points with the smallest residuals from the robust subspace are taken as initial h-subset. After performing the classical PARAFAC algorithm on these h points, the h-subset is updated by taking the h observations with smallest residuals. This whole procedure is iterated until the relative change in fit becomes small. Finally, a reweighting step is included to increase the accuracy of the method.

We illustrate this robust PARAFAC method on the three-way fluorescence Dorrit data [81, 82]. The data set contains 27 excitation-emission (EEM) landscapes and is preprocessed as in [80] leading to a data array of size $27 \times 116 \times 18$. Four fluorophores are mixed together for different sets of concentrations, so $F = 4$.

To find the outlying EEM-landscapes, we apply the robust PARAFAC method and construct outlier maps (defined analogously to PCA). The classical and robust outlier maps are depicted in Figure 17. In the classical outlier map, observation 10 is marked as a residual outlier whereas samples 2, 3, 4 and 5 are flagged as good leverage points. The robust analysis yields a very different conclusion. It shows that that samples 2, 3 and 5 are bad leverage points. In order to find out which of these results we should trust, the emission and excita-



(a) (b)

Figure 17: Outlier maps of the Dorrit data set based on (a) classical PARAFAC and (b) robust PARAFAC.

tion loadings are plotted in Figure 18. The classical results in Figure 18(a–b) are corrupted by the outliers, as neither the emission loadings nor the excitation loadings correspond to the expected profiles of the known chemical compounds (see e.g. [81]). Moreover, we can compute the angle (in radians) between the estimated and the reference subspaces for the B- and C-loadings. This yields 1.34 and 0.44, which confirms that the classical estimates are far off. On the other hand, based on visual inspection of Figure 18(c-d) and the angles 0.06 for the B-loadings and 0.18 for the C-loadings, we can conclude that the robust algorithm has succeeded in estimating the underlying structure of the data much more accurately. When the data contain missing elements, an EM-based iterative adaptation is presented in [83].

The Tucker3 model [84] adds an $L \times M \times N$ core matrix $\underline{\boldsymbol{Z}}$ in (27):

$$x_{ijk} = \sum_{l=1}^{L} \sum_{m=1}^{M} \sum_{n=1}^{N} a_{il} b_{jm} c_{kn} z_{lmn} + e_{ijk}.$$

An algorithm, based on the MCD estimator, has been proposed to construct a robust Tucker3 algorithm [85].

Note that all methods described in this chapter assume that there is an outlier-free majority of samples available. If all samples are somehow contaminated, this methodology will not work and algorithms for element-wise contamination are needed, see e.g. [86, 38, 87]. A typical example of elementwise contamination is scattering in fluorescence data, which affects all or a vast majority of the samples. A robust procedure to identify this scattering is proposed in [88] and uses the ROBPCA method on sliced data matrices.

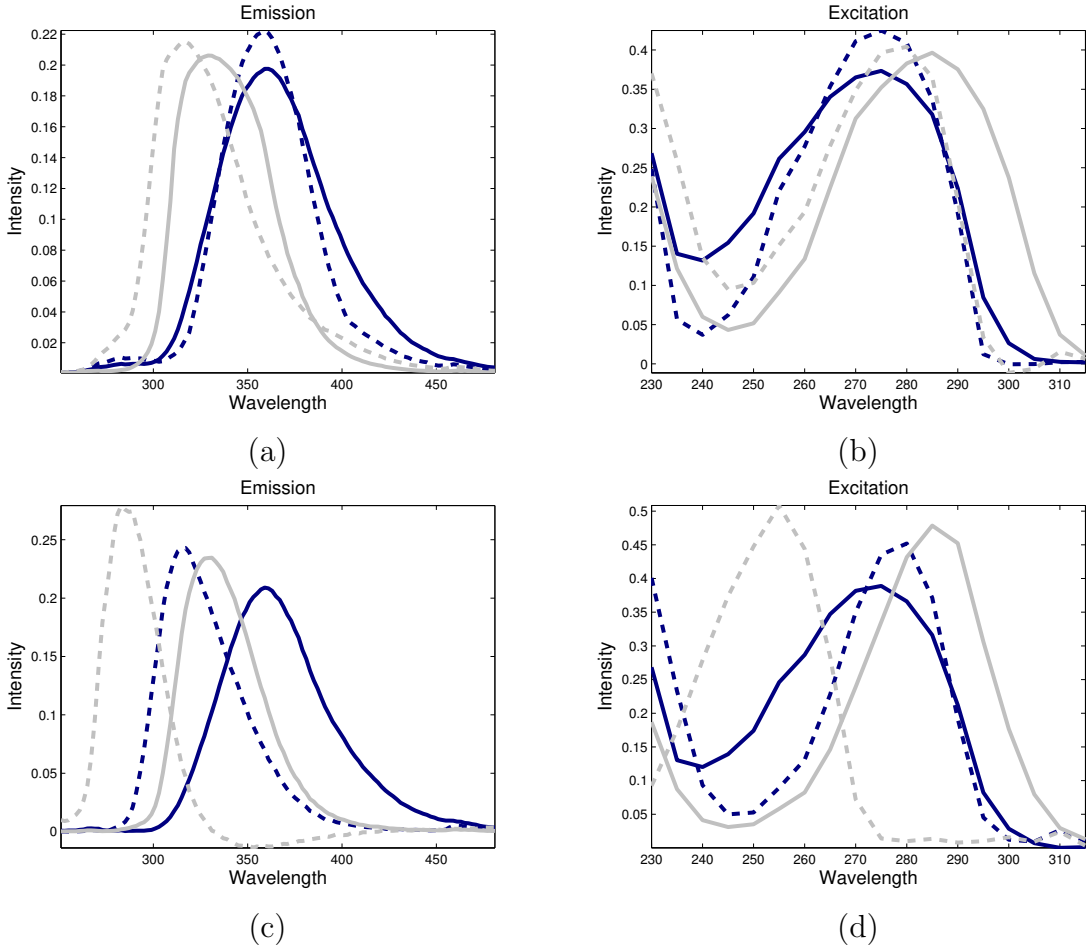


Figure 18: Emission (left) and excitation (right) loadings for the Dorrit data set, using the classical (top) and robust (bottom) PARAFAC algorithms.

6 Software availability

Many robust functions described in this chapter are part of "LIBRA, Library for Robust Analysis", which can be downloaded from `http://wis.kuleuven.be/statdatascience/robust`. The user-friendly syntax of the functions is described in [89, 90]. Besides multivariate calibration methods, LIBRA also contains robust functions for univariate location, scale and skewness, and for classification and clustering. On the former website functions for robust multiway analysis are provided as well. Matlab functions for MCD, LTS, MCD-regression, ROBPCA, RPCR and RSIMPLS are also part of the PLS_Toolbox (`http://www.eigenvector.com`). Also the Matlab library TOMCAT [91] contains a few robust calibration methods.

The free software R contains many implementations of robust methods, e.g. in the packages `robustbase, rrcov, rrcovHD, rospca, mrfDepth, cellWise`.
See `http://wis.kuleuven.be/statdatascience/robust/software` for an overview.

Support Vector Machines are implemented in numerous software packages, see e.g. the

R package `libsvm`. For Least Squares SVM applications, a Matlab toolbox is available at `http://www.esat.kuleuven.be/sista/lssvmlab`.

Acknowledgements

The author gratefully acknowledges financial support by project C16/15/068 of Internal Funds KU Leuven, and support from the CRoNoS project, with reference CRoNoS COST Action IC1408.

References

[1] P. J. Rousseeuw, A. M. Leroy, Robust Regression and Outlier Detection, Wiley-Interscience, New York, 1987.

[2] M. Hubert, S. Engelen, Robust PCA and classification in biosciences, Bioinformatics 20 (2004) 1728–1736.

[3] R. A. Maronna, Robust M-estimators of multivariate location and scatter, The Annals of Statistics 4 (1976) 51–67.

[4] N. A. Campbell, Robust procedures in multivariate analysis I: Robust covariance estimation, Applied Statistics 29 (1980) 231–237.

[5] C. Croux, G. Haesbroeck, Principal components analysis based on robust estimators of the covariance or correlation matrix: influence functions and efficiencies, Biometrika 87 (2000) 603–618.

[6] P. J. Rousseeuw, Least median of squares regression, Journal of the American Statistical Association 79 (1984) 871–880.

[7] D. L. Donoho, P. J. Huber, The notion of breakdown point, in: P. Bickel, K. Doksum, J. L. Hodges (Eds.), A Festschrift for Erich Lehmann, Wadsworth, Belmont, 1983, pp. 157–184.

[8] M. Hubert, M. Debruyne, P. J. Rousseeuw, Minimum Covariance Determinant and extensions, Wiley Interdisciplinary Reviews: Computational Statistics 10 (3) (2018) e1421.

[9] M. Salibian-Barrera, S. Van Aelst, G. Willems, PCA based on multivariate MM-estimators with fast and robust bootstrap, Journal of the American Statistical Association 101 (2006) 1198–1211.

[10] G. Li, Z. Chen, Projection-pursuit approach to robust dispersion matrices and principal components: primary theory and Monte Carlo, Journal of the American Statistical Association 80 (1985) 759–766.

[11] C. Croux, A. Ruiz-Gazen, High breakdown estimators for principal components: the projection-pursuit approach revisited, Journal of Multivariate Analysis 95 (2005) 206–226.

[12] C. Croux, P. Filzmoser, M. Oliviera, Algorithms for projection-pursuit robust principal component analysis, Chemometrics and Intelligent Laboratory Systems 87 (2007) 218–225.

[13] M. Hubert, P. J. Rousseeuw, S. Verboven, A fast robust method for principal components with applications to chemometrics, Chemometrics and Intelligent Laboratory Systems 60 (2002) 101–111.

[14] I. Stanimirova, B. Walczak, D. Massart, V. Simenov, A comparison between two robust PCA algorithms, Chemometrics and Intelligent Laboratory Systems 71 (2004) 83–95.

[15] W. Wu, D. Massart, S. de Jong, The kernel PCA algorithms for wide data. Part I: Theory and algorithms, Chemometrics and Intelligent Laboratory Systems 36 (1997) 165–172.

[16] P. J. Rousseeuw, C. Croux, Alternatives to the median absolute deviation, Journal of the American Statistical Association 88 (1993) 1273–1283.

[17] H. Cui, X. He, K. Ng, Asymptotic distributions of principal components based on robust dispersions, Biometrika 90 (2003) 953–966.

[18] M. Hubert, P. J. Rousseeuw, K. Vanden Branden, ROBPCA: a new approach to robust principal component analysis, Technometrics 47 (2005) 64–79.

[19] S. Engelen, M. Hubert, K. Vanden Branden, A comparison of three procedures for robust PCA in high dimensions, Austrian Journal of Statistics 34 (2005) 117–126.

[20] R. A. Maronna, Principal components and orthogonal regression based on robust scales, Technometrics 47 (2005) 264–273.

[21] H. Cevallos-Valdiviezo, S. Van Aelst, Fast computation of robust subspace estimators, Computational Statistics & Data Analysis 134 (2019) 171–185.

[22] N. Locantore, J. Marron, D. Simpson, N. Tripoli, J. Zhang, K. Cohen, Robust principal component analysis for functional data, Test 8 (1999) 1–73.

[23] J. Raymaekers, P. J. Rousseeuw, A generalized spatial sign covariance matrix, Journal of Multivariate Analysis 171 (2019) 94 – 111.

[24] J. L. Bali, G. Boente, D. E. Tyler, J.-L. Wang, Robust functional principal components: A projection-pursuit approach, The Annals of Statistics 39 (6) (2011) 2852–2882.

[25] G. Boente, M. Salibian-Barrera, S-estimators for functional principal component analysis, Journal of the American Statistical Association 110 (511) (2015) 1100–1111.

[26] G. E. P. Box, Some theorems on quadratic forms applied in the study of analysis of variance problems: Effect of inequality of variance in one-way classification, The Annals of Mathematical Statistics 25 (1954) 33–51.

[27] I. Jolliffe, Principal Component Analysis, Springer, New York, 1986.

[28] S. Wold, Cross-validatory estimation of the number of components in factor and principal components models, Technometrics 20 (1978) 397–405.

[29] H. Eastment, W. Krzanowski, Cross-validatory choice of the number of components from a principal components analysis, Technometrics 24 (1982) 73–77.

[30] M. Hubert, S. Engelen, Fast cross-validation for high-breakdown resampling algorithms for PCA, Computational Statistics & Data Analysis 51 (2007) 5013–5024.

[31] P. Lemberge, I. De Raedt, K. Janssens, F. Wei, P. J. Van Espen, Quantitative Z-analysis of 16th–17th century archaelogical glass vessels using PLS regression of EPXMA and μ-XRF data, Journal of Chemometrics 14 (2000) 751–763.

[32] C. Croux, P. Filzmoser, H. Fritz, Robust sparse principal component analysis, Technometrics 55 (2013) 202–214.

[33] M. Hubert, T. Reynkens, E. Schmitt, T. Verdonck, Sparse PCA for high-dimensional data with outliers, Technometrics 58 (4) (2016) 424–434.

[34] I. T. Jolliffe, N. T. Trendafilov, M. Uddin, A modified principal component technique based on the LASSO, Journal of Computational and Graphical Statistics 12 (2003) 531–547.

[35] B. Walczak, D. Massart, Tutorial: Dealing with missing data, part I, Chemometrics and Intelligent Laboratory Systems 58 (2001) 15–27.

[36] S. Serneels, T. Verdonck, Principal component analysis for data containing outliers and missing elements, Computational Statistics & Data Analysis 52 (2008) 1712–1727.

[37] M. Hubert, P. J. Rousseeuw, W. Van den Bossche, MacroPCA: An all-in-one PCA method allowing for missing values as well as cellwise and rowwise outliers, Technometrics 61 (4) (2019) 459–473.

[38] P. J. Rousseeuw, W. Van den Bossche, Detecting deviating data cells, Technometrics 60 (2018) 135–145.

[39] P. J. Rousseeuw, K. Van Driessen, Computing LTS regression for large data sets, Data Mining and Knowledge Discovery 12 (2006) 29–45.

[40] P. J. Rousseeuw, M. Hubert, Recent developments in PROGRESS, in: L1-Statistical Procedures and Related Topics, Institute of Mathematical Statistics Lecture Notes-Monograph Series Vol 31, Hayward, California, 1997, pp. 201–214.

[41] P. J. Rousseeuw, B. C. van Zomeren, Unmasking multivariate outliers and leverage points, Journal of the American Statistical Association 85 (1990) 633–651.

[42] P. J. Rousseeuw, S. Van Aelst, K. Van Driessen, J. Agulló, Robust multivariate regression, Technometrics 46 (2004) 293–305.

[43] H. Martens, T. Naes, Multivariate calibration, Wiley, Chichester, UK, 1998.

[44] M. Hubert, S. Verboven, A robust PCR method for high-dimensional regressors, Journal of Chemometrics 17 (2003) 438–452.

[45] S. de Jong, SIMPLS: an alternative approach to partial least squares regression, Chemometrics and Intelligent Laboratory Systems 18 (1993) 251–263.

[46] M. Hubert, K. Vanden Branden, Robust methods for Partial Least Squares Regression, Journal of Chemometrics 17 (2003) 537–549.

[47] K. Vanden Branden, M. Hubert, Robustness properties of a robust PLS regression method, Analytica Chimica Acta 515 (2004) 229–241.

[48] S. Serneels, C. Croux, P. Filzmoser, P. J. Van Espen, Partial robust M-regression, Chemometrics and Intelligent Laboratory Systems 79 (2005) 55–64.

[49] S. Engelen, M. Hubert, Fast model selection for robust calibration, Analytica Chemica Acta 544 (2005) 219–228.

[50] S. Verboven, M. Hubert, P. Goos, Robust preprocessing and model selection for spectral data, Journal of Chemometrics 26 (2012) 282–289.

[51] K. H. Esbensen, S. Schönkopf, T. Midtgaard, Multivariate Analysis in Practice, Camo, Trondheim, 1994.

[52] B. G. Osborne, T. Fearn, A. R. Miller, S. Douglas, Application of near infrared reflectance spectroscopy to the compositional analysis of biscuits and biscuit dough, Journal of Scientific Food Agriculture 35 (1984) 99–105.

[53] B. Walczak, D. Massart, Robust principal component regression as a detection tool for outliers, Chemometrics and Intelligent Laboratory Systems 27 (1995) 41–54.

[54] R. J. Pell, Multiple outlier detection for multivariate calibration using robust statistical techniques, Chemometrics and Intelligent Laboratory Systems 52 (2000) 87–104.

[55] S. Møller, J. von Frese, R. Bro, Robust methods for multivariate data analysis, Journal of Chemometrics 19 (2005) 549–563.

[56] M. Stone, R. Brooks, Continuum regression: cross-validated sequentially constructed prediction embracing ordinary least squares, partial least squares and principal components regression (with discussion), Journal of the Royal Statistical Association, B 52 (1990) 237–269.

[57] S. Serneels, P. Filzmoser, C. Croux, P. J. Van Espen, Robust continuum regression, Chemometrics and Intelligent Laboratory Systems 76 (2005) 197–204.

[58] F. Marini, R. Bucci, A. Magri, A. Magri, Authentication of italian *cdo* wines by class-modeling techniques, Chemometrics and Intelligent Laboratory Systems 84 (2006) 164–171.

[59] X. He, W. Fung, High breakdown estimation for multiple populations with applications to discriminant analysis, Journal of Multivariate Analysis 72 (2000) 151–162.

[60] C. Croux, C. Dehon, Robust linear discriminant analysis using S-estimators, The Canadian Journal of Statistics 29 (2001) 473–492.

[61] M. Hubert, K. Van Driessen, Fast and robust discriminant analysis, Computational Statistics & Data Analysis 45 (2004) 301–320.

[62] K. Vanden Branden, M. Hubert, Robust classification in high dimensions based on the SIMCA method, Chemometrics and Intelligent Laboratory Systems 79 (2005) 10–21.

[63] M. Hubert, P. J. Rousseeuw, P. Segaert, Multivariate and functional classification using depth and distance, Advances in Data Analysis and Classification 11 (2017) 445–466.

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

[65] N. Cristianini, J. Shawe-Taylor, An Introduction to Support Vector Machines, Cambridge University Press, Cambridge, 2000.

[66] B. Schölkopf, A. Smola, Learning with Kernels, MIT Press, Cambridge, MA, 2002.

[67] P. J. Rousseeuw, M. Debruyne, S. Engelen, M. Hubert, Robustness and outlier detection in chemometrics, Critical Reviews in Analytical Chemistry 36 (2006) 221–242.

[68] A. Christmann, I. Steinwart, On robust properties of convex risk minimization methods for pattern recognition, Journal of Machine Learning Research 5 (2004) 1007–1034.

[69] D. M. J. Tax, R. P. W. Duin, Support vector data description, Machine Learning 54 (2004) 45–66.

[70] B. Üstün, W. J. Melsen, L. M. C. Buydens, Facilitating the application of support vector regression by using a universal Pearson VII function based kernel, Chemometrics and Intelligent Laboratory Systems 81 (2006) 29–40.

[71] A. Christmann, I. Steinwart, Consistency and robustness of kernel based regression, Bernoulli 13 (2007) 799–819.

[72] J. A. K. Suykens, T. Van Gestel, J. De Brabanter, B. De Moor, J. Vandewalle, Least Squares Support Vector Machines, World Scientific, Singapore, 2002.

[73] J. A. K. Suykens, J. De Brabanter, L. Lukas, J. Vandewalle, Weighted least squares support vector machines : Robustness and sparse approximation, Neurocomputing 48 (2002) 85–105.

[74] M. Debruyne, A. Christmann, M. Hubert, J. Suykens, Robustness of reweighted least squares kernel based regression, Journal of Multivariate Analysis 101 (2010) 447–463.

[75] R. Bro, Multi-way analysis in the food industry, Ph.D. thesis, Royal Veterinary and Agricultural university, Denmark (1998).

[76] A. Smilde, R. Bro, P. Geladi, Multi-way Analysis with Applications in the Chemical Sciences, Wiley, England, 2004.

[77] C. M. Andersen, R. Bro, Practical aspects of PARAFAC modelling of fluorescence excitation-emission data, Journal of Chemometrics 17 (2003) 200–215.

[78] R. Bro, Exploratory study of sugar production using fluorescence spectroscopy and multi-way analysis, Chemometrics and Intelligent Laboratory Systems 46 (1999) 133–147.

[79] R. D. Jiji, G. Andersson, K. Booksh, Application of PARAFAC for calibration with excitation-emission matrix fluorescence spectra of three classes of environmental pollutants, Journal of Chemometrics 14 (2000) 171–185.

[80] S. Engelen, M. Hubert, Detecting outlying samples in a parallel factor analysis model, Analytica Chimica Acta 705 (2011) 155–165.

[81] D. Baunsgaard, Factors affecting 3-way modelling (PARAFAC) of fluorescence landscapes, Ph.D. thesis, Royal Veterinary and Agricultural University, Department of Dairy and Food technology, Frederiksberg, Denmark (1999).

[82] J. Riu, R. Bro, Jack-knife technique for outlier detection and estimation of standard errors in PARAFAC models, Chemometrics and Intelligent Laboratory Systems 65 (2003) 35–49.

[83] M. Hubert, J. Van Kerckhoven, T. Verdonck, Robust PARAFAC for incomplete data, Journal of Chemometrics 26 (2012) 290–298.

[84] L. Tucker, Some mathematical notes on three-mode factor analysis, Psychometrika 31 (1966) 279–311.

[85] V. Pravdova, B. Walczak, D. L. Massart, A robust version of the Tucker3 model, Chemometrics and Intelligent Laboratory Systems 59 (2001) 75–88.

[86] C. Agostinelli, A. Leung, V. J. Yohai, R. H. Zamar, Robust estimation of multivariate location and scatter in the presence of cellwise and casewise contamination, Test 24 (3) (2015) 441–461.

[87] P. J. Rousseeuw, M. Hubert, Anomaly detection by robust statistics, WIREs Data Mining and Knowledge Discovery 8 (2) (2018) e1326–n/a.

[88] S. Engelen, S. Frosch Møller, M. Hubert, Automatically identifying scatter in fluorescence data using robust techniques, Chemometrics and Intelligent Laboratory Systems 86 (2007) 35–51.

[89] S. Verboven, M. Hubert, LIBRA: a Matlab library for robust analysis, Chemometrics and Intelligent Laboratory Systems 75 (2005) 127–136.

[90] S. Verboven, M. Hubert, MATLAB library LIBRA, Wiley Interdisciplinary Reviews: Computational Statistics 2 (2010) 509–515.

[91] M. Daszykowski, S. Serneels, K. Kaczmarek, P. Van Espen, C. Croux, B. Walczak, TOMCAT: a MATLAB toolbox for multivariate calibration techniques, Chemometrics and Intelligent Laboratory Systems 85 (2007) 269–277.