# Robust multivariate methods in Chemometrics

Peter Filzmoser[1] Sven Serneels[2] Ricardo Maronna[3]
Pierre J. Van Espen[4]
[1] Institut für Statistik und Wahrscheinlichkeitstheorie,
Technical University of Vienna, Vienna, Austria
[2] Statistics and Chemometrics Group, Shell Global Solutions,
Shell Research and Technology Centre, Amsterdam, The Netherlands
[3] Department of Mathematics, National University of La Plata, Argentina
[4] Department of Chemistry, University of Antwerp, Belgium

**Abstract**

This chapter presents an introduction to robust statistics with applications of a chemometric nature. Following a description of the basic ideas and concepts behind robust statistics, including how robust estimators can be conceived, the chapter builds up to the construction (and use) of robust alternatives for some methods for multivariate analysis frequently used in chemometrics, such as principal component analysis and partial least squares. The chapter then provides an insight into how these robust methods can be used or extended to classification. To conclude, the issue of validation of the results is being addressed: it is shown how uncertainty statements associated with robust estimates, can be obtained.

Keywords: robust statistics, robustness, location, scale, regression, M estimators, principal component analysis, partial least squares, linear discriminant analysis, D-PLS, validation, bootstrap, prediction interval.

# Contents

# List of Abbreviations

| | |
|---|---|
| CV | cross-validation |
| D-PLS | discrimination partial least squares |
| EIF | empirical influence function |
| EPXMA | electron probe x-ray micro-analysis |
| IF | influence function |
| IRPLS | iteratively re-weighted partial least squares |
| IRWLS | iteratively re-weighted least squares |
| LAD | least absolute deviation |
| LDA | linear discriminant analysis |
| LIBRA | Library for Robust Analysis |
| LMS | least median of squares |
| LS | least squares |
| LTS | least trimmed squares |
| MAD | median absolute deviation |
| MATLAB | Matrix Laboratory |
| MCD | minimum covariance determinant |
| MSE | mean squared error |
| OLS | ordinary least squares |
| PARAFAC | parallel factor analysis |
| PC | principal component |
| PCA | principal component analysis |
| PCR | principal component regression |
| PLAD | partial least absolute deviations |
| PLS | (uni- or multivariate) partial least squares |
| PLS2 | multivariate partial least squares |
| PM | partial M |
| PP | projection pursuit |
| PP-PLS | projection pursuit partial least squares |
| PRM | partial robust M |
| QDA | quadratic discriminant analysis |
| RAPCA | reflection-based algorithm for principal component analysis |
| RCR | robust continuum regression |
| RMSE | root mean squared error |
| RMSECV | root mean squared error of cross validation |
| RMSEP | root mean squared error of prediction |
| ROBPCA | robust PCA (one specific method by Hubert *et al.*[32]) |
| RSIMPLS | robust SIMPLS (one specific method by Hubert *et al*[30]) |
| SPC | spherical principal components |
| SVD | singular value decomposition |
| TOMCAT | Toolbox for Multivariate Calibration Techniques |
| tri-PLS | trilinear partial least squares |

# 1 Introduction

## 1.1 The concept of robustness

Many statistical methods are based on distributional assumptions. Especially the normal distribution takes on a primordial role: well-known optimality properties of the most frequently applied estimators do only hold at the normal model. For instance, the least squares estimator for regression is known to be the maximum likelihood estimator at the normal model. Nearly all regression methods which are common in chemometrics, are to some extent derived from least squares. This implies that the normal distribution assumes a key position in multivariate chemometric methods.

In practice data do never exactly follow the normal distribution. In most cases the normality assumption is satisfactory and methods based on it will produce reliable results. However, sometimes the normality approximation to the data is rather poor or even completely wrong. Data may intrinsically follow a different distribution than the normal (e.g. think of counting statistics such as X-ray counts which are Poisson distributed). The data may also show bi- or multimodality because the individual cases have been drawn from different populations. Here one can think of a data set containing cases which are known to appertain to different groups, but for which a joint calibration model is desired. E.g. different types of wines need to be analysed for their ester concentration. From the offset it is known that samples of different years and soils have different properties and thus belong to different populations. Nevertheless a model which predicts the ester concentration reliably independently of their origin or year may be required. For such a model probably a regression technique will be used, albeit it is clear that the data were not generated by a single normal distribution. Alternatively, the data may have been generated by the same model but have been influenced by different processes. Samples may have been generated in a similar manner but have undergone exposition to different effects (temperature, light, etc.), changing their behaviour, such that the assumption of a single distribution becomes invalid.

The normality assumption may also be violated by an entirely different process. Outliers may occur which have atypical properties compared to the majority of the data. Outliers can be generated in several ways: they can be objects which intrinsically have different properties or they can be artifacts produced by the data generation process. A typical example in chemometrics would be that some cases have been measured with a different light source or detector such that the spectra cannot be included in a single model with the regular cases. When outliers are present in the data it does not make sense to model the data distribution including the outliers. The true model according to which the non-outlying data points have been generated will differ significantly from a model estimated by data containing outliers.

All the above situations (multimodality, outliers) are examples of situations where the data do not follow a normal model. Whereas they are all examples of nonnormality, robust methods have explicitly been developed for the last mentioned situation. Robust estimators are estimators derived for a given model, including slight deviations from this model. More precisely, if the main group of data points is assumed to come from a distribution $G$, then a robust estimator for such data is designed for the distribution $G_\varepsilon = (1-\varepsilon)G + \varepsilon H$, where $H$

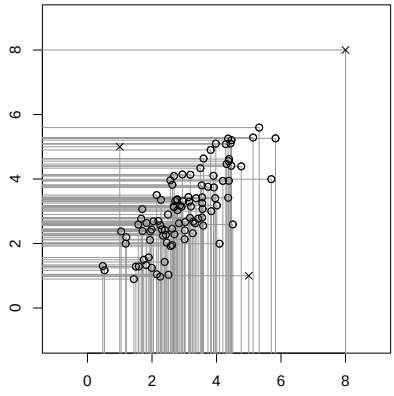



Figure 1: Bivariate normally distributed data cloud ($\circ$) and three outliers at different positions ($\times$) as well as the projection of each point onto both axes.

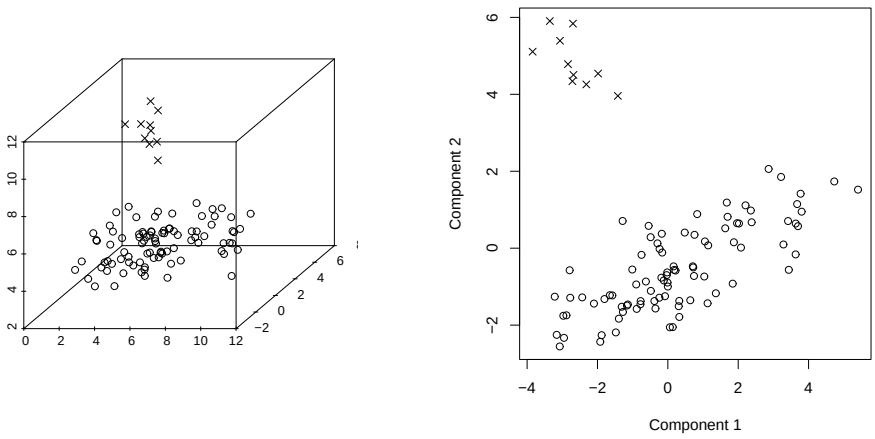
is another distribution and $\varepsilon \in [0, 1)$. Because robust estimators are usually especially designed for such $\varepsilon$ contaminated distributions, they should resist any type of moderate deviation from $G$. This implies that robust estimators can in practice also perform well at distributions which are close to $G$. For instance, if $G$ is the normal distribution, then robust estimators are designed for a normal contaminated with outliers coming from a given outlier generating distribution $H$, but they may perform well as well for heavier tailed "close to normal" distributions such as the Cauchy and Student's $t$ distributions.

## 1.2 Visualising multivariate data for outlier identification

Detection of either multimodality or outliers is straightforward for univariate and bivariate data. For multivariate data clouds it will be difficult or even impossible to visualise the data and graphically detect the outliers. In particular chemometric calibration and classification problems are usually of a high dimensional nature: e.g. in spectrophotometry, the spectra are commonly measured at $p > 1000$ variables. Visual inspection by simply plotting the data is practically impossible as one would need to inspect plots of all possible pairs of two variables. Even then the outliers can be of a multivariate nature such that they will not be detected by inspecting only two dimensions. Consider the reduction of a bivariate problem to one variable. In Figure 1 a bivariate distribution is plotted. Three outliers are added; it can be seen that a projection of the data onto either one of both axes does only reveal one of the three outliers. It is straightforward to imagine that a similar effect occurs when one projects multivariate data on two dimensions.

## 1.3 Masking effect

Instead of simply projecting the data onto a pair of its variables, classical statistical methods can be used for data reduction. A straightforward approach is



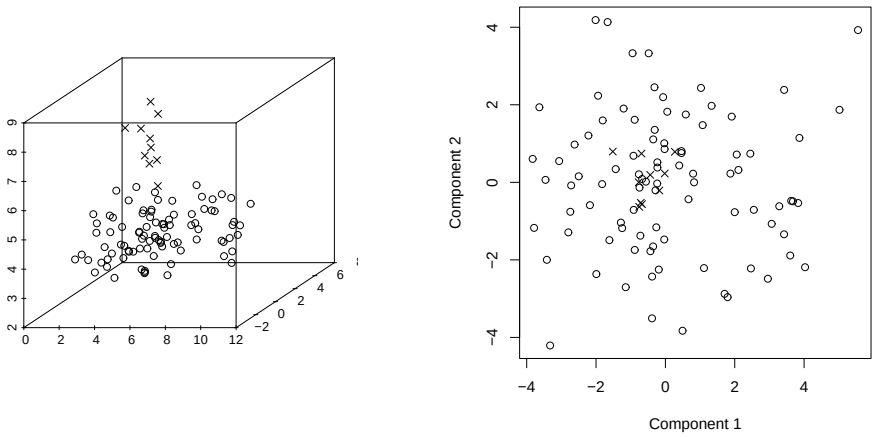

(a) (b)

Figure 2: Trivariate normally distributed data cloud (○) and a group of outliers (×); plotted are (a) the data and (b) a scatterplot of PC1 vs. PC2

summarise the data into principal components and then making pairwise plots of these principal components. Alternatively, if a dependent variable exists as well, a data reduction technique can be used which takes into account the relation to the predictand, such summarising the data into latent variables by partial least squares or canonical correlation analysis. However, methods like principal component analysis or partial least squares are classical (i.e. nonrobust) estimators, which implies that the outliers do also have an effect on the estimates of the latent variables obtained by those methods. The estimated latent variables can be biased in such a way that in fact no outliers are detected. The effect that due to the outliers' presence one is not able to detect them is referred to as the *masking effect*. In Figure 2 we show a trivariate data distribution to which a cluster (ten percent) of outliers has been added. In the left subplot the data are shown as a three dimensional plot; the right subplot shows a biplot of the first two principal components (PCs) of this data set. In Figure 3 similar plots are shown for the same data cloud where the outliers have been added at a different position. From Figures 2 and 3 we see that it depends on the position in space where the outliers are situated whether principal components reveal them or not. A good robust data reduction method (discussed more into detail later in this chapter) would reveal the outliers in both configurations, whereas it would also still yield reasonable results if the data are not contaminated by any type of outliers.

## 1.4 Swamping effect

Apart from the masking effect, outliers can also cause typical points to be identified as outliers. This is easily understood by taking regression as an example. Figure 4 shows a typical regression problem with normally distributed data to which a group of outliers is added. The least squares regression tries to fit all data points, i.e. also the outliers, and is thus attracted towards the outliers. Normally one would assign outliers to a least squares fit as those points which




(a) (b)

Figure 3: Trivariate normally distributed data cloud (○) and a group of outliers (×); plotted are (a) the data and (b) a scatterplot of PC1 vs. PC2

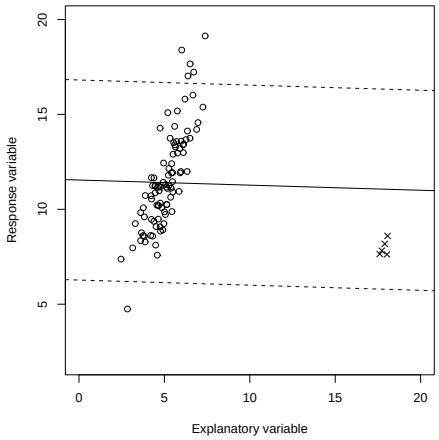



Figure 4: A typical regression problem with normally distributed errors; a cloud of outliers has been added. The solid line shows the least squares fit to the data. The dotted lines show the limits beyond which points are considered to be outlying.

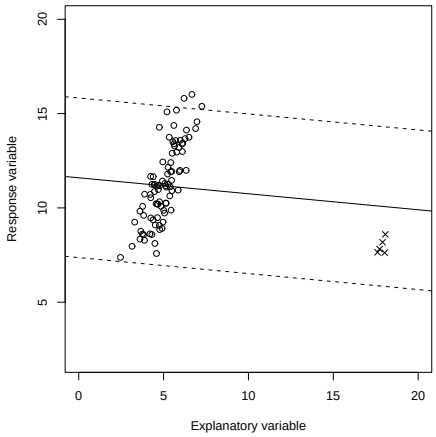



Figure 5: The regression problem from Figure 4 after removal of the outliers detected there.

have the highest residuals to the model. In Figure 4 the regression line is shown together with two dashed lines indicating a residual distance of two standard errors. Potential outliers will fall outside these bands. In the example shown, the points having the largest residuals, which are thus detected "outliers" are in this case not the true outliers, but typical points which are far away from the regression line just because the latter badly fits the majority of the data. The effect that regular data points are identified as outliers is denominated the *swamping effect*.

The above figures show two main effects of outliers. At first the outliers themselves are hard to detect (remember that in a multivariate setting, a plot which shows the entire data distribution cannot be constructed). Secondly, the outliers cause the regression line to be severely distorted. Predictions of future responses made according to this line will be unreliable. At this point one would intuitively proceed by detection of the outliers after which the regression analysis is performed on the remaining points. But exactly due to the masking and swamping effects the wrong data points may be identified as outliers such that eventually the regression analysis on the assumed "clean" data remains a jeopardy. In Figure 5 we show what happens if one would proceed by omitting the identified outliers in Figure 4. Some of the good data points have been removed; the true outliers are still present. The slope of the regression line is still erroneous due to the outliers and the latter are still not detected.

## 1.5 Majority fit

The goal of robust methods is to estimate parameters under conditions of slight deviations to the model. Once the parameters are estimated in a robust way it is possible to identify outliers which are considered to be deviating data points according to the underlying statistical model. It should be observed that outlier detection always involves some subjectivity, while parameter estimation does not. Gross outliers can be detected and henceforth omitted from the analysis.

Apart from these, various other deviations from the underlying model may be present such as a small groups of points which are known to have slightly different properties but cannot be omitted. In this case it is not possible to delete the deviating data points prior to a classical analysis. Nonetheless the classical analysis may probably accord too great an importance to the few differing points. A well chosen robust estimator will provide a reliable fit for the whole range spanned by the data points without being influenced by deviating points, regardless the type of deviation. Even gross outliers may be present in the data without influencing the robust fit.

Most robust methods may be described as classical methods where the data are weighted, with weights depending on the data. The majority of the data will receive a (quasi) uniform weight while the more atypical individual cases are, the lower the weight they will get. Summarising, a robust fit can be considered to be a majority fit, where the fraction of data which makes up the majority depends on the points in space of the individual cases.

### 1.6 Is robustness a synonym to wasting information?

Many early robust estimators were based on trimming (e.g. the well known least trimmed squares estimator for regression where a pre-determined percentage of the largest squared residuals is trimmed and thus not considered for finding the regression parameters). On the other hand, trimming is not done for *any* data points, but especially for the largest and smallest values. If any data points would be trimmed, the precision (or efficiency) of the resulting estimator would be much lower than that where the extremes are trimmed. This fact underlines already that even trimming does not "waste" information. Ideally, robust estimation techniques should only discard data points which are extremely distinct from the bulk of data and are thus very likely to be gross outliers. All other data points should to some extent be taken into account. The amount of information taken from each data point is then regulated by weights between 0 and 1 given to them. This procedure will in general lead to estimators with higher efficiency.

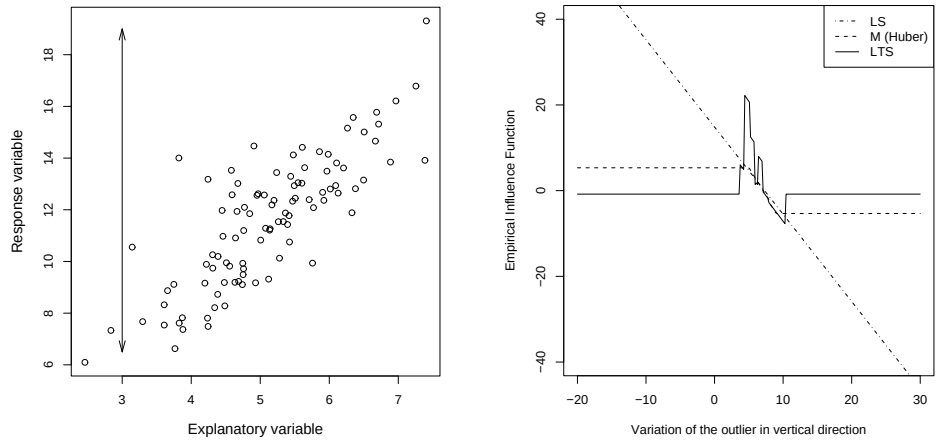
## 2 Designing robust multivariate estimators

### 2.1 Which properties should a robust estimator have?

Robust estimators should be resistant to a sizeable proportion of outliers or deviation from assumptions. They should also still yield reasonable results if these ideal assumptions are valid. In this section some tools are introduced to assess an estimator's robustness properties: the influence function, the maxbias curve and the statistical efficiency.

#### 2.1.1 Empirical influence function and influence function

One of the basic ideas of robustness is that a robust estimator should not be influenced by a limited amount of contamination, regardless where this contamination is situated. A simple way to check the behaviour of an estimator under small contamination is to vary a single data point. As an example, Figure 6 shows the effect of an observation which is varied in space to different regression




(a) (b)

Figure 6: Empirical influence functions for regression estimators. Subplots show (a) the data with varying positions of the outlier and (b) the empirical influence functions of the slope parameter estimated by least squares (LS), M regression and least trimmed squares (LTS).

estimators. The data were taken from a normal distribution. The position of one data point was changed as is shown in Figure 6a. For each position of the data point, we are interested in the change of the slope parameter of the following regression estimators: least squares regression and two robust regression estimators: Huber M regression[27] and LTS regression estimators (see later in this chapter). The result is known as the empirical influence function (EIF), but in order to make the results of the different estimators comparable, we compute the difference of the slopes for the contaminated and uncontaminated data, and divide by the amount $1/n$ of contamination. The right subplot of Figure 6 shows the results. It can be seen that the least squares estimator has an unbounded EIF. This means that a single gross outlier can have an arbitrarily large effect on the estimator. Both robust estimators possess a bounded EIF. However, not only the bound but also the shape of the EIF is of importance in order to understand how a robust estimator deals with contamination. Ideally the EIF should be smooth: it should not show local spikes or should not be a step function. In practice, the effect of placing a data point at one location and then shifting it to a very close position should be very small. One observes that indeed the M estimator has a smooth EIF and is thus virtually insensitive to local shifts in the data. On the contrary, the response of the least trimmed squares estimator to small data perturbations is far from smooth (in the literature it is said to be prone to a high *local shift sensitivity*).

The concept of the EIF can be formalised by the so-called influence function (IF). The IF measures the influence an infinitesimal amount of contamination has on an estimator with respect to its position in space.[24] More precisely, the influence function of an estimator $T$ at a given distribution $G$ is defined as:

$$\mathrm{IF}(\mathbf{z}, T, G) = \lim_{\varepsilon \downarrow 0} \frac{T\left[(1-\varepsilon)G + \varepsilon\delta_{\mathbf{z}}\right] - T(G)}{\varepsilon}, \tag{1}$$

where $\varepsilon$ is the fraction of contamination and $\delta_{\mathbf{z}}$ is a probability measure which puts all the mass at $\mathbf{z}$. The point $\mathbf{z}$ can be any point in the $p$ dimensional space but in practice it will often be a measured data point. Evaluating the influence function at the points of a data set reveals how each data point changes the estimator's behaviour. The influence of an outlier in the data set on the estimator can be measured by evaluating the influence function at the outlier. For nonrobust estimators evaluation of the influence function at the outlier will yield significantly different results compared to evaluating the IF at the typical data, whereas for a robust estimator the effect will be limited.

### 2.1.2 Maxbias curve

Hitherto we have considered the influence of a limited amount of contamination at varying positions in space. An interesting question is what happens if instead of the position in space one changes the proportion of contamination. What one expects is that a robust estimator can withstand a certain fraction of contamination. The mathematical tool to examine to which extent an estimator is distorted with respect to the fraction of contamination in the data is the maxbias curve. The maxbias curve measures the bias an estimator has with respect to the percentage of the worst possible type of contamination. Let $Z$ be the original data set and $\check{Z}$ be a data set in which $m$ out of $n$ observations have been replaced with arbitrary values and let $\| \cdot \|$ denote the Euclidean norm, then the maxbias curve for an estimator $T$ is defined as:

$$\text{maxbias}(m, T, Z) = \sup_{\check{Z}} \| T(\check{Z}) - T(Z) \| . \tag{2}$$

It is known that for some estimates of regression the worst possible type of outliers is found at points where $y$, $x$ and the fraction $y/x$ increase to infinity. In what follows a numerical example is shown which does not reflect the exact maxbias curve but illustrates what happens if bad (but not the worst type) of outliers are added to data for the regression problem discussed in the previous section. For the data set presented in Figure 6, we have added vertical outliers in the following manner: points were added in a range $(\mu_x + a, \mu_y + b)$ about the mean, where $a \in [0, 10]$ and $b \in [10^4, 10^5]$. We then computed the bias of the slope compared to the known regression slope. In Figure 7 we see that the bias of the least squares estimator tends to infinity if a single observation is replaced by bad outliers. The other robust estimators exhibit a moderate bias up till a certain point where they also break down. To conclude we note that if the outliers' position in $x$ would have been put further away from the data cloud too (leverage points), then we would observe a faster breakdown for the robust M regression method displayed here, as this method is known to be only resistant to outliers in $y$.

### 2.1.3 Breakdown point

From the bias curves one observes that for each estimator, there exists a point where the bias tends to infinity with $\varepsilon$. This point is referred to as the *breakdown point*. Loosely, the breakdown point indicates which percentage of the data may be replaced with outliers before the estimator yields aberrant results. Based on

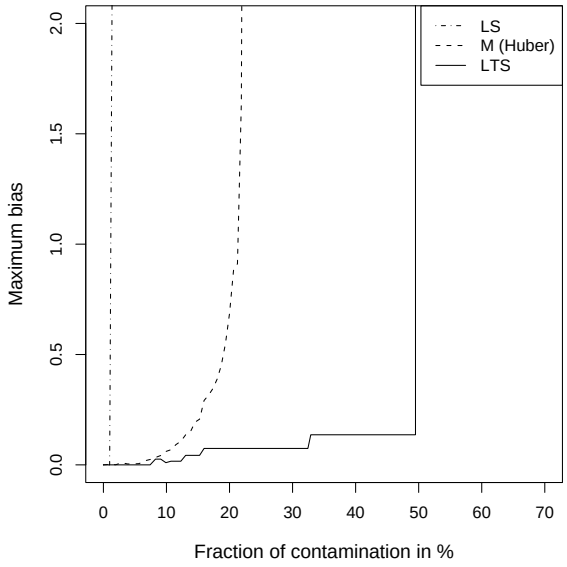



Figure 7: The regression problem from Figure 4 after removal of the outliers detected there.

the maxbias curve, for finite samples the breakdown point is given by:

$$\varepsilon_n^*(T, Z) = \min\left\{\frac{m}{n}; \text{maxbias}(m, T, Z) = \infty\right\}. \tag{3}$$

For $n \to \infty$ one obtains the *asymptotic breakdown point*, denoted $\varepsilon^*$. For least squares regression it holds that $\varepsilon^* = 0$. The maximal possible value of the asymptotic breakdown point equals 1. However, estimators satisfying equivariance conditions[1] have a maximal asymptotic breakdown point of 0.5, which means that the typical points must out-number the outliers in order to produce meaningful results. One of the goals in designing robust estimators is obtaining a high breakdown point. Howbeit, bounded influence and high breakdown should not result in a drastic decrease in efficiency.

### 2.1.4 Statistical efficiency

An important property to any statistical estimator is the variance. It is well known that many parametric estimators have optimality properties at their underlying model. For instance, the maximum likelihood estimator for the linear regression model with normally distributed error terms is the least squares estimator. The least squares estimator is also the minimum variance unbiased estimator for the linear model (the Gauß-Markov theorem). This implies that predictions made by any other regression estimator for data which follow the linear model with normally distributed error terms, will have a higher uncertainty

[1]Equivariance conditions are deemed reasonable for most estimating problems, e.g. location, covariance and regression; for the definition of affine equivariance see Section 3.7.1 and for the definition of scale equivariance see Section 3.3.

than the least squares predictions. So also robust estimators for regression are prone to an increase in variance compared to least squares. This statement may be generalised to other settings than regression; it can be stated that robust estimators always have a higher variance than classical parametric estimators if evaluated at the underlying model of the parametric estimator. They are said to be less *efficient* than parametric estimators. So as to design robust estimators it is important not only to investigate the robustness properties but also the efficiency properties. One could even conjecture that for chemometric robust estimators, efficiency is a more important property than robustness in the breakdown sense. Data sets for multivariate calibration hardly ever contain 50% of outliers. Moreover, a few outliers very far away from the data cloud would readily be detected by inspecting the data. What is occurring more frequently is a data set which slightly deviates from normality (the most frequently assumed underlying distribution) without any gross outliers being present. For such data a robust estimator will outperform the classical estimator because the latter is only optimal at the exact normal model, given it is efficient, such that the effect of increase in variance of the robust estimator does not compensate for the classical estimator's loss in precision due to deviation from normality.

# 3 Robust regression

Regression assumes a key position in chemometrics. Apart from being applied as a method in its own right, it is also a part of more complex estimators such as partial least squares or three-way methods. One way to develop robust alternatives to these methods is by replacing the classical regressions by robust ones. In that case the properties of the entire method derive from the type of robust regression used. In this section we present an overview of some of the most useful robust estimators for regression.

The data of a regression situation are the $n \times p$ matrix $\mathbf{X}$ of predictor variables with elements $x_{ij}$ and the $n-$vector $\mathbf{y}$ with elements $y_i$. For regression with intercept we assume that the first column of the data matrix is a column of ones. Call $\mathbf{x}_i = (x_{i1}, \ldots, x_{ip})^T$ the column vector containing the elements of the $i$-th row of $\mathbf{X}$. The linear regression model is then given by

$$y_i = \mathbf{x}_i^T \boldsymbol{\beta} + e_i, \;\; i = 1, ..., n, \tag{4}$$

where the unknown regression parameter $\boldsymbol{\beta}$ is a $p$-vector and $e_i$ denotes the error terms, which are assumed to be i.i.d. random variables.

For a given estimator $\hat{\boldsymbol{\beta}}$ call $r_i = r_i\left(\hat{\boldsymbol{\beta}}\right) = y_i - \mathbf{x}_i^T \hat{\boldsymbol{\beta}}$ the $i$-th residual. Most regression estimators are based on a minimisation of the size of the residuals. The classical least squares (LS) estimator is defined as

$$\hat{\boldsymbol{\beta}}_{LS} = \arg\min_{\boldsymbol{\beta}} \sum_{i=1}^{n} r_i\left(\boldsymbol{\beta}\right)^2. \tag{5}$$

An ancient alternative to LS is the $L_1$ estimator defined as

$$\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \sum_{i=1}^{n} \left| r_i\left(\boldsymbol{\beta}\right) \right|. \tag{6}$$

## 3.1 M-estimators

The LS estimator is not robust, in the sense that atypical observations may uncontrollably affect the outcome. The reason is that a large residual would dominate the sum (5). One way out of this difficulty is to define a more general family of estimators. Note that (5) and (6) can be written as:

$$\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \sum_{i=1}^{n} \rho(r_i(\boldsymbol{\beta})), \tag{7}$$

where $\rho(r) = r^2$ for LS and $\rho(r) = |r|$ for $L_1$. By taking other $\rho$ functions different estimators are obtained. In fact, equation (7) is the definition of a whole class of estimators commonly referred to as *M-estimators*.

Note that if $\hat{\boldsymbol{\beta}}$ is the LS estimate, then transforming $\mathbf{y}$ to $t\mathbf{y}$ with $t \in \Re$ transforms $\hat{\boldsymbol{\beta}}$ to $t\hat{\boldsymbol{\beta}}$. This property, called *regression equivariance*, is not shared by the estimator (7), except when $\rho(z) = |z|^a$ for some $a$. To make estimator (7) scale equivariant, we define in general a regression M-estimator by

$$\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \sum_{i=1}^{n} \rho\left(\frac{r_i(\boldsymbol{\beta})}{\hat{\sigma}}\right), \tag{8}$$

where $\hat{\sigma}$ is a robust scale estimator of the residuals, that can be estimated either previously or simultaneously with the regression parameters.

The function $\rho$ must be chosen adequately. Recall that we want an estimate to be (a) robust in the sense of being insensitive to outliers, and (b) efficient in the sense of being similar to LS when there are no outliers. For (a) to hold, $\rho(r)$ must increase more slowly than $r^2$ for large $r$, and for (b), $\rho(r)$ must be approximately quadratic for small $r$.

Differentiating (8) with respect to $\boldsymbol{\beta}$ we get that the estimate fulfils the system of *M-estimating equations*

$$\sum_{i=1}^{n} \psi\left(\frac{r_i(\boldsymbol{\beta})}{\hat{\sigma}}\right)\mathbf{x}_i = 0 \tag{9}$$

where $\psi = \rho'$. For LS, $\psi(r) = r$, and (9) are the well-known normal equations. We may then in general interpret (9) as a robustified version of the normal equations, where the residuals are curbed. For $L_1$ we have $\psi(r) = \operatorname{sign}(r)$. In general, solutions of (9) are *local* minima of (8), which may or may not coincide with the global minimum.

Put $W(r) = \psi(r)/r$. Then (9) may be rewritten as

$$\sum_{i=1}^{n} w_i\left(y_i - \mathbf{x}_i^T\boldsymbol{\beta}\right)\mathbf{x}_i = 0 \tag{10}$$

with $w_i = W(r_i(\boldsymbol{\beta})/\hat{\sigma})$. Then (10) is a weighted version of the normal equations, and hence the estimator can be seen as weighted LS, with the weights depending on the data . For LS, $W$ is constant. For an estimator to be robust, observations with large residuals should receive a small weight, which implies that $W(r)$ has to decrease to zero fast enough for large $r$.

If $\psi$ is an increasing function, the estimate is called *monotonic*. A family of monotonic estimators which contains LS and $L_1$ as extreme cases is the *Huber* family, with $\rho'$ given by

$$\psi_{\mathrm{H},k}(r) = \begin{cases} r & \text{for} \quad |r| \leq k \\ k \operatorname{sign}(r) & \text{otherwise} \end{cases} \tag{11}$$

The extreme cases $k \to \infty$ and $k \to 0$ correspond to LS and $L_1$, respectively.

Monotonic estimates have the computational advantage that (9) gives the *global* minima of (8). But they may lack robustness if $\mathbf{X}$ contains atypical rows (the so-called *leverage points*). The intuitive reason is that if some $\mathbf{x}_i$ is "large", then the $i$-th term will dominate the sum in (10), which would be unfortunate if $(\mathbf{x}_i, y_i)$ is atypical (a "bad leverage point"). For this reason it is better to use M-estimators given by (8) with a *bounded* $\rho$. An example is the *bisquare* family, with

$$\rho_{\mathrm{B},k}(r) = \begin{cases} \left(\frac{r}{k}\right)^2 \left(3 - 3\left(\frac{r}{k}\right)^2 + \left(\frac{r}{k}\right)^4\right) & \text{for} \quad |r| \leq k \\ 1 & \text{else} \end{cases} . \tag{12}$$

Bounded $\rho$s present computational difficulties which will be discussed in the next Sections.

When $k \to \infty$ in (11) or (12), the corresponding estimate tends to LS and hence becomes more efficient and at the same time less robust. Thus $k$ is a tuning parameter the choice of which is a compromise between efficiency and robustness. The usual practice is to choose $k$ to attain a given efficiency, such as 0.90.

## 3.2 Computing M-estimators

Equation (10) suggests an iterative procedure to obtain local minima. Assume we have an initial value $\hat{\boldsymbol{\beta}}_0$. Call $\hat{\boldsymbol{\beta}}_m$ the approximation at iteration $m$. Then given $\hat{\boldsymbol{\beta}}_m$, compute the residuals $r_i = r_i\left(\hat{\boldsymbol{\beta}}_m\right)$ and then the weights $w_i = W(r_i/\hat{\sigma})$, and solve (10) to obtain $\hat{\boldsymbol{\beta}}_{m+1}$. The procedure is called *iterative reweighted least squares* (IRWLS), and converges if $W(z)$ is a decreasing function of $|z|$ (Maronna et al., 2006).[41]

If $\psi$ is monotonic, the choice of $\hat{\boldsymbol{\beta}}_0$ influences the number of iterations, but not the final outcome. But if $\rho$ is bounded, then $\psi$ tends to zero at infinity, which implies that there may be many local minima, and therefore the choice of $\hat{\boldsymbol{\beta}}_0$ is crucial. Using a non-robust initial estimator like LS may yield "bad" local minima.

A good initial estimator is also necessary to obtain the scale $\hat{\sigma}$. If there are no leverage points one could use $L_1$ as an initial $\hat{\boldsymbol{\beta}}_0$, and compute $\hat{\sigma}$ as a robust scale of the residuals $r_i(\hat{\sigma})$ (e.g. the MAD). But otherwise we need other choices. The initial estimator should not need a previous residual scale. Before initial estimators can be considered, we present some further concepts.

## 3.3 Robust measures of residual size

Given $\mathbf{r} = (r_1, ..., r_n)$ we shall define a scale $\sigma(\mathbf{r})$ such that $\sigma(t\mathbf{r}) = |t|\sigma(\mathbf{r})$ for $t \in \Re$ (called *scale equivariance*). We shall consider two types of scales.

### 3.3.1 Scales based on ordered values

Call $|r|_{(i)}$ the ordered absolute values of the $r_i$s: $|r|_{(1)} \leq ... \leq |r|_{(n)}$. The simplest scale is a *quantile* of $\mathbf{r}$ :

$$\sigma(\mathbf{r}) = |r|_{(h)} \tag{13}$$

for some $h \in \{1, .., n\}$. For $h = n/2$ we have the median. Other choices will be considered below.

A smoother alternative is to consider a scale more similar to the standard deviation, namely the *trimmed squares scale*

$$\sigma(\mathbf{r}) = \left( \frac{1}{n} \sum_{i=1}^{h} |r|_{(i)}^2 \right)^{1/2}, \tag{14}$$

which for $h = n$ gives the familiar root mean squared error (RMSE).

### 3.3.2 Scale M-estimators

Henceforth a *$\rho$-function* will denote a function $\rho$ such that $\rho(x)$ is a nondecreasing function of $|x|$, $\rho(0) = 0$, and $\rho(x)$ is (strictly) increasing for $x > 0$ such that $\rho(x) < \rho(\infty)$. if $\rho$ is bounded, it is also assumed that $\rho(\infty) = 1$.

An M-estimator of scale (an *M-scale* for short) is defined as the solution $\sigma$ of an equation of the form

$$\frac{1}{n} \sum_{i=1}^{n} \rho\left( \frac{r_i}{\sigma} \right) = \delta \tag{15}$$

where $\rho$ is a $\rho$-function and $\delta \in (0, \rho(\infty))$. The choice $\rho(z) = z^2$ and $\delta = 1$ yields the RMSE. The choice $\rho(z) = \mathrm{I}(|z| > 1)$ and $\delta = 0.5$ yields $\sigma = \mathrm{med}(|r|)$, where "med" denotes the median. We shall be interested in estimates with *bounded* $\rho$.

Equation (15) is nonlinear, but it is easy to solve iteratively. Put

$$W_\sigma(z) = \frac{\rho(z)}{z^2}. \tag{16}$$

Then (15) can be rewritten as

$$\sigma^2 = \frac{1}{n\delta} \sum_{i=1}^{n} w_i r_i^2$$

with $w_i = W_\sigma(r_i/\sigma)$, which displays $\hat{\sigma}$ as a weighted RMSE. Given some starting value $\sigma_0$, an iterative procedure can be implemented as was done for regression M-estimators.

### 3.3.3 Calibrating scales for consistency

If $z \sim \mathrm{N}(0, \sigma^2)$, then the standard deviation of $z$ is $\sigma$ by definition. The median of $|z|$ is instead $0.675\sigma$. Hence if we have a sample $z_1, ..., z_n$, then $\hat{\sigma} = \mathrm{med}(|z_1|, ..., |z_n|)$ will tend to $0.675\sigma$ for large $n$, and therefore $\hat{\sigma}/0.675$ would be an approximately unbiased estimate of $\sigma$ for normal data.

In general, given a scale estimate $\hat{\sigma}$ it is convenient to "normalise" it by dividing it through a constant $c$ so that $\hat{\sigma}/c$ estimates the standard deviation at the normal model. The M-scale (15) with $\rho = \rho_{\mathrm{B},1}$ has $c$=1.65.

## 3.4 Regression estimators based on a robust residual scale

Given $\beta$, let $\mathbf{r}(\boldsymbol{\beta}) = (r_1(\boldsymbol{\beta}), ..., r_n(\boldsymbol{\beta}))$. We shall consider an estimator of the form

$$\hat{\beta} = \arg\min_{\boldsymbol{\beta}} \hat{\sigma}(\mathbf{r}(\boldsymbol{\beta})) \tag{17}$$

where $\hat{\sigma}$ is a robust scale.

### 3.4.1 The LMS and LTS estimators

If $\hat{\sigma}$ is given by (13), we have the *least quantile estimator.* The case $h = n/2$ is the *least median of squares* (LMS) estimate. Actually, to attain maximum breakdown point one must take

$$h = \left[\frac{n+p+1}{2}\right] \tag{18}$$

where $[t]$ is the integer part of $t$. Estimators of this class are very robust in the sense of having a low bias, but their asymptotic efficiency is zero.

If $\hat{\sigma}$ is given by (14), we have the *least trimmed squares* (LTS) estimator. Again, the optimal $h$ is given by (18). Their asymptotic efficiency is about 7%.

### 3.4.2 Regression S estimators

Regression estimators with $\hat{\sigma}$ given by (15) are called *S-estimators.* It can be shown that they satisfy the M-estimating equations (9) with $\psi = \rho'$; and it follows that, given an initial approximation, they can be computed by means of the IRWLS algorithm. The boundedness of $\rho$ is necessary for the robustness of the estimate. But if $\rho$ is bounded, then $\psi$ is not monotonic, which implies that the equations yield only *local* minima of $\hat{\sigma}(\mathbf{r}(\boldsymbol{\beta}))$, and hence a reliable starting approximation is needed. An approach to obtain an initial estimator is given in the next Section.

The efficiency of the S-estimator with $\rho$ the bisquare function is about 29%. In general, it can be shown that the efficiency of S-estimators cannot exceed 33%. Although better than LMS and LTS, S-estimators do not allow the user to choose a desired high efficiency. This goal is attained by the estimates to be described in Section 3.6.

## 3.5 The subsampling algorithm

The approach to find an approximate solution to (17) is to compute a "large" finite set of candidate solutions, and replace the minimisation over $\beta \in R^p$ by minimising $\widehat{\sigma}(\mathbf{r}(\beta))$ over that finite set. To compute the candidate solutions we take subsamples of size $p$

$$\{(\mathbf{x}_i, y_i) : i \in J\}, \quad J \subset \{1, ..., n\}, \ \#(J) = p.$$

For each $J$ find $\boldsymbol{\beta}_J$ that satisfies the exact fit $\mathbf{x}_i^T \boldsymbol{\beta}_J = y_i$ for $i \in J$. Then the problem of minimising $\widehat{\sigma}(\mathbf{r}(\beta))$ for $\beta \in R^p$ is replaced by the finite problem of minimising $\widehat{\sigma}(\mathbf{r}(\beta_J))$ over $J$. Since choosing all $\binom{n}{p}$ subsamples would be

prohibitive unless both $n$ and $p$ are rather small, we choose $N$ of them at random: $\{J_k : k = 1, .., N\}$ and the initial estimate $\widehat{\boldsymbol{\beta}}_{J_{k^*}}$ is defined by

$$k^* = \arg\min \left\{ \widehat{\sigma} \left( \mathbf{r} \left( \boldsymbol{\beta}_{J_k} \right) \right) : k = 1, ..., N \right\}. \tag{19}$$

Suppose the sample contains a proportion $\varepsilon$ of outliers. The probability of an outlier-free subsample is $\alpha = (1-\varepsilon)^p$, and the probability of at least one outlier-free subsample is $1 - (1-\alpha)^N$. If we want this probability to be larger than $1-\gamma$, we must have

$$\ln \gamma \geq N \ln (1-\alpha) \approx -N\alpha$$

and hence

$$N \geq \frac{|\ln \gamma|}{|\ln (1-(1-\varepsilon)^p)|} \approx \frac{|\ln \gamma|}{(1-\varepsilon)^p} \tag{20}$$

for $p$ not too small. Therefore $N$ must grow exponentially with $p$ if robustness is to be ensured.

## 3.6 Regression MM-estimators

We are now ready to define a family of estimators attaining both robustness and controllable efficiency. We shall deal with (8) where $\rho(z)$ is a *bounded* $\rho$-function. We assume that there is a previous estimator $\hat{\boldsymbol{\beta}}_0$ which is robust but possibly inefficient (e.g. an S-estimator). Compute $\hat{\boldsymbol{\beta}}_0$ and the corresponding residuals $r_i$. Compute $\hat{\sigma}$ as an M-scale (15) with $\rho = \rho_{\mathrm{B},1}$. Let $\tilde{\sigma} = \hat{\sigma}/c_0$ with $c_0 = 1.65$. Now let $\rho = \rho_{\mathrm{B},k}$ with $k$ chosen to have a given efficiency $\gamma$. We recommend $\gamma = 0.85$ which implies $k = 3.44$. Then compute $\hat{\boldsymbol{\beta}}$ as a local solution of (8) using the IRWLS starting from $\hat{\boldsymbol{\beta}}_0$. The resulting estimator has the BP of $\hat{\boldsymbol{\beta}}_0$ and the asymptotic efficiency $\gamma$.

It is shown by Maronna et al. (2006)[41] that if $\hat{\boldsymbol{\beta}}_0$ is the bisquare S-estimator, then the resulting MM-estimator with efficiency 0.85 has a contamination bias not much larger than that of $\hat{\boldsymbol{\beta}}_0$.

## 3.7 Robust location and covariance

Multivariate location and covariance play a central role in multivariate statistics because many multivariate methods directly build on these estimates. For example, principal component analysis is carried out on the centred data, and the standard method uses a decomposition of the covariance matrix to find the principal components. Outliers or deviations from a model distribution can lead to very different results, and thus it is necessary to robustly estimate multivariate location and covariance. Many methods have been proposed for this purpose. Before discussing various approaches, we will first think about desired properties of robust location and covariance estimators. Aside from robustness issues, a central property is affine equivariance which will be discussed below.

### 3.7.1 Affine equivariance

It is desirable that location and covariance estimates respond in a mathematically convenient form to certain transformations of the data. For example, if a

constant is added to each data point, the location estimate of the modified data should be equal to the location estimate of the original data plus this constant, but the covariance estimate should remain unchanged. Similarly, if each data point is multiplied by a constant, the new location estimate should be equal to the old one multiplied by the same constant, and the new variances should be the constant squared times the old variances. More general, one can define a transformation that is using a nonsingular $p \times p$ matrix $\mathbf{A}$ and a vector $\mathbf{b}$ of length $p$ to transform the $p$-dimensional observations $\mathbf{x}_1, \ldots, \mathbf{x}_n$ by $\mathbf{A}\mathbf{x}_j + \mathbf{b}$. This transformation performs any desired nonsingular linear transformation of the original data. Thus, if $\mathbf{t}$ denotes a location estimator, it is requested that

$$\mathbf{t}(\mathbf{A}\mathbf{x}_1 + \mathbf{b}, \ldots, \mathbf{A}\mathbf{x}_n + \mathbf{b}) = \mathbf{A} \cdot \mathbf{t}(\mathbf{x}_1, \ldots, \mathbf{x}_n) + \mathbf{b}, \tag{21}$$

and for a covariance estimator $\mathbf{C}$ we require

$$\mathbf{C}(\mathbf{A}\mathbf{x}_1 + \mathbf{b}, \ldots, \mathbf{A}\mathbf{x}_n + \mathbf{b}) = \mathbf{A} \cdot \mathbf{C}(\mathbf{x}_1, \ldots, \mathbf{x}_n) \cdot \mathbf{A}^T. \tag{22}$$

Location and covariance estimators that fulfil (21) and (22) are called *affine equivariant* estimators. These estimators transform properly under changes of the origin, the scale, or under rotations.

Figure 8a shows a bivariate data set where the location (+) was estimated by the arithmetic mean and the covariance by the sample covariance matrix. The latter is visualised by so-called *tolerance ellipses*: In case of normally distributed data the tolerance ellipses would contain a certain percentage of data points around the centre and according to the covariance structure. Here we show the 50% and 90% tolerance ellipses. Figure 8b pictures the data after applying the transformation

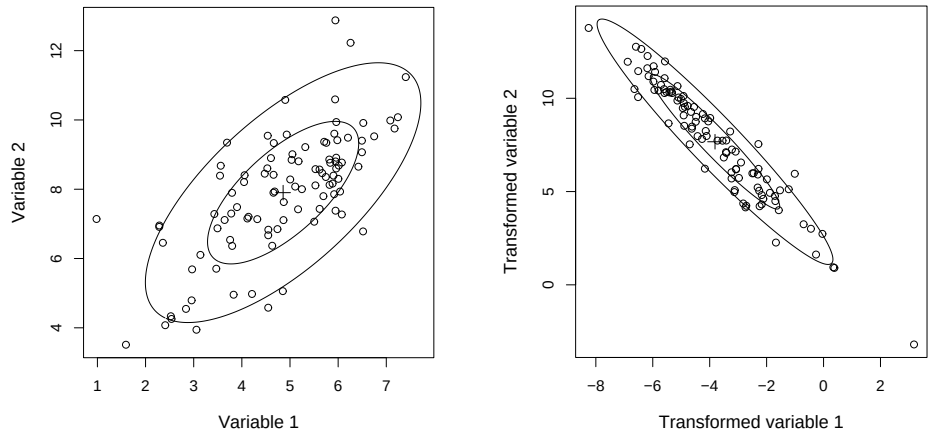
$$\mathbf{A}\mathbf{x}_j + \mathbf{b} = \begin{pmatrix} -2 & 3 \\ 1 & -1 \end{pmatrix} \mathbf{x}_j + \mathbf{b}$$

to each data point. The location and covariance estimates were not recomputed for the transformed data but were transformed according to the equations (21) and (22). It is obvious from the figure that the transformed estimates are the same as if they would have been derived directly from the transformed data. Note that the transformation matrix $\mathbf{A}$ is close to singularity because the spread of the data becomes very small in one direction. The property of affine equivariance is only valid for nonsingular transformation matrices.

Affine equivariance is not only important for estimation of location and covariance but also for multivariate methods like discriminant analysis or canonical correlation analysis. The results of these methods will remain unchanged under linear transformations. This is different for principal component analysis which is *orthogonal equivariant* but not affine equivariant. The results will only be properly transformed under orthogonal transformation matrices $\mathbf{A}$.

### 3.7.2 Asymptotic breakdown point

The breakdown point (for simplicity, we will omit "asymptotic in this section) was already discussed in section 2.1.3 in the context of regression, and it is also used as an important characterisation of robustness for location and covariance estimators. Clearly, the breakdown points of the classical estimators, the arithmetic mean and the sample covariance matrix, are both 0 because even a single




(a) (b)

Figure 8: Bivariate data with estimated location (+) and covariance matrix (visualised by tolerance ellipses); plotted are (a) the original data and (b) the transformed data, together with the transformed estimates.

observation placed at an arbitrary position in space can completely spoil these estimators.

The simplest choice for a very robust location estimator would be the median, computed for each variable. Since the median for univariate data has a breakdown point of 0.5, the coordinate-wise median would also have this breakdown point. However, this estimator is not affine equivariant. For the data used in Figure 8a the median for the variables is $(4.97, 8.14)$ and for the transformed data in Figure 8b it is $(-4.12, 7.98)$. The transformation of the coordinate-wise median of the original data results in $(-3.79, 7.76)$. This difference would in general become larger if less data points were available.

The univariate median is that point which minimises the sum of the distances to all data points. A natural extension of this concept to higher dimensions is called *spatial median* or $L_1$-median. It is defined as that point in the multivariate space which minimises the sum of the Euclidean distances to all data points. The spatial median has good statistical properties: it has a breakdown point of 0.5. However, this multivariate location estimator is only orthogonal equivariant but not affine equivariant. Note that in the context of principal component analysis or partial least squares this would be sufficient because these methods are only orthogonal equivariant.

### 3.7.3 The MCD estimator

An estimator of multivariate location and covariance which is affine equivariant and has high breakdown point is the *Minimum Covariance Determinant* (MCD) estimator. The idea behind this estimator is in fact related to the LTS estimator from Section 3.4.1. Here, one is searching for those $h$ data points for which the determinant of the (classical) covariance matrix is minimal. The location estimator $\mathbf{t}$ is the mean of these $h$ observations, and the covariance estimator $\mathbf{C}$ is given by the covariance matrix with the smallest determinant, but multiplied

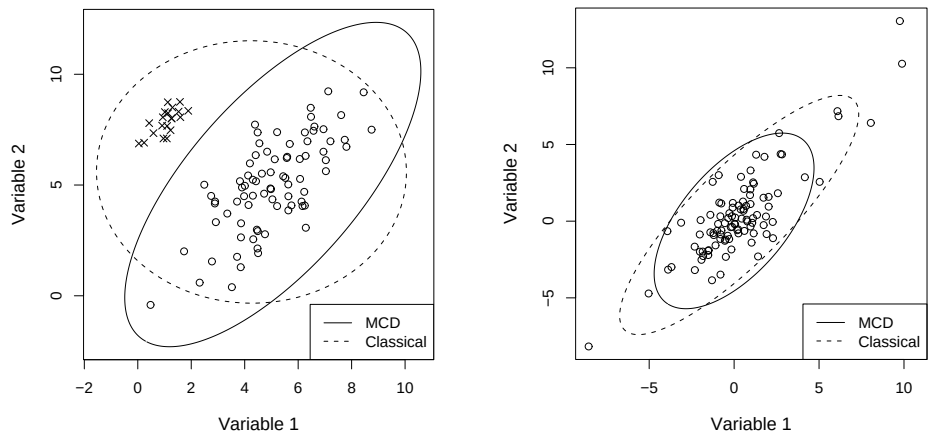



Figure 9: Tolerance ellipses (97.5%) based on the MCD estimator and on the classical sample mean and sample covariance matrix for (a) bivariate normally distributed data with outliers and (b) bivariate $T_2$ distributed data.

by a constant to obtain consistency for normal distribution. The parameter $h$ determines the robustness but also the efficiency of the resulting estimator. The highest possible breakdown point can be achieved if $h \approx n/2$ is taken, but this choice leads to a low efficiency. On the other hand, for higher values of $h$ the efficiency increases but the breakdown point decreases. Therefore, a compromise between efficiency and robustness is considered in practice.

The computation of the MCD estimator is not trivial. While for a low number of samples in low dimension in principle all subsets of $h$ data points can be considered in order to find the subset with smallest determinant of its covariance matrix, this is no longer possible for large $n$ or in higher dimension. For this situation, a fast algorithm has been proposed which finds an approximation of the solution.[47]

It is important to note that the MCD estimator can only be applied to data sets where the number of observations is larger than the number of variables, which is a serious limitation for many applications in chemometrics. The reason is that if $p > n$ then also $p > h$, and the covariance matrix of any $h$ data points will always be singular, leading to a determinant of 0. Thus, each subset of $h$ data points would lead to the smallest possible determinant, resulting in a non-unique solution. In fact, a non-trivial solution can only be obtained if $h$ is smaller than the rank of the data.

Figure 9 shows a comparison between the MCD estimator and classical location and covariance estimation for two simulated data sets. The covariance estimates are visualised by 97.5% tolerance ellipses, the location estimates are the centres of the ellipses. In Figure 9a a bivariate normally distributed data set is used, where 20% of the data points are generated with a different mean and covariance. Note that these deviating data points cannot be identified as outliers by inspecting the projections on the coordinates. Not only the location estimate is influenced by the deviating points, but especially the covariance

structure. The data coming from the outlier distribution are inflating the tolerance ellipse based on the classical estimators while that based on the MCD is much more compact and reflects the structure of the majority of data.

The second example shown in Figure 9b is simulated from a bivariate $T$ distribution with 2 degrees of freedom with a certain covariance structure. Also here the inflation of the classical ellipse due to some very distant points is visible.

We can also compute the correlation coefficient using the classical and robust covariance estimates. In the first example the classical correlation is 0.00 while the MCD gives a correlation of 0.70, which is also the result of the classical correlation for the data without outliers. For the second example we obtain a value of 0.84 for the classical correlation and 0.60 for the robust correlation.

### 3.7.4 Multivariate S estimators

Similar as in the regression context (see Section 3.4.2), it is possible to define S estimators in the context of robust location and covariance estimation.[9,38] The idea is to make the Mahalanobis distances small. The Mahalanobis or multivariate distances are defined as

$$d(\mathbf{x}_i, \mathbf{t}, \mathbf{C}) = (\mathbf{x}_i - \mathbf{t})^T \mathbf{C}^{-1} (\mathbf{x}_i - \mathbf{t}) \quad \text{for } i = 1, \ldots, n$$

for a location estimator $\mathbf{t}$ and a covariance estimator $\mathbf{C}$. Note that $d$ is actually a squared distance. Thus, in contrast to the squared Euclidean distance

$$d(\mathbf{x}_i, \mathbf{t}) = (\mathbf{x}_i - \mathbf{t})^T (\mathbf{x}_i - \mathbf{t}) \quad \text{for } i = 1, \ldots, n$$

the Mahalanobis distance also accounts for the covariance structure of the data. Small Mahalanobis distances can be achieved by using a scale estimator $\sigma$ and minimising $\sigma(d(\mathbf{x}_1, \mathbf{t}, \mathbf{C}), \ldots, d(\mathbf{x}_n, \mathbf{t}, \mathbf{C}))$ under the restriction that the determinant of $\mathbf{C}$ is 1. Davies[9] suggested to take for the scale estimator $s$ an M estimator of scale[24,27] which has been defined in Equation (15).

S estimators are affine equivariant, for differentiable $\rho$ they are asymptotically normal, and for well-chosen $\rho$ and $\delta$ they achieve maximum breakdown point.

### 3.7.5 Multivariate MM estimators

Like in robust regression (see Section 3.6), a drawback of S estimators is that their asymptotic efficiency might be rather low. MM estimators for multivariate location and covariance combine both high breakdown point and high efficiency.[39] The resulting estimators are affine equivariant and have bounded influence function. The solution for the estimators can be found by an iterative algorithm.

### 3.7.6 The Stahel-Donoho estimator

The name of this estimator for multivariate location and covariance origins from independent findings of Stahel[57] and Donoho.[16] The idea is based on downweight outlying observations in the classical estimation of multivariate location and covariance. Outlying observations are observations which are deviating from the multivariate data structure with respect to the majority of data points. Note that multivariate outliers are not necessarily univariate outliers, since they can

be "hidden" in the multivariate space. An example are the outliers in Figure 9a that could not be identified as univariate outliers by inspecting the values on the original coordinates.

Finding multivariate outliers is in some sense strongly related to multivariate location and covariance estimation. Once reliable estimates have been derived, the Mahalanobis distances can be computed, and observations with large values of the Mahalanobis distance can be considered as potential multivariate outliers. All methods discussed so far possessing high breakdown point are potentially suitable for multivariate outlier detection.

The Stahel-Donoho estimator first identifies multivariate outliers in a very simple way: Each observation is projected to the one-dimensional space and a measure of outlyingness is computed. Of course, there are infinitely many possible projection directions from multivariate to one dimension, and thus an infinite number of measures of outlyingness for each observation is obtained. Thus, the goal is to identify the supremum over all possible projection directions $\mathbf{a} \in \Re^p$ with $\|\mathbf{a}\| = 1$ of the measure of outlyingness,

$$\text{out}(\mathbf{x}_i, \mathbf{X}) = \sup_{\mathbf{a}} \frac{|\mathbf{x}_i^T \mathbf{a} - m(\mathbf{X}\mathbf{a})|}{s(\mathbf{X}\mathbf{a})} \tag{23}$$

for observation $\mathbf{x}_i$ $(i = 1, \ldots, n)$ of the data set $\mathbf{X}$. Here, $m$ and $s$ and robust univariate location and scatter estimators, respectively, e.g. the median and the MAD. Using an appropriate weight function $w$, each observations receives a weight $w_i = w(\text{out}(\mathbf{x}_i, \mathbf{X}))$, depending on its outlyingness. The location estimator is then defined as

$$\mathbf{t} = \frac{1}{\sum_{i=1}^n w_i} \sum_{i=1}^n w_i \mathbf{x}_i$$

and the covariance estimator as

$$\mathbf{C} = \frac{1}{\sum_{i=1}^n w_i} \sum_{i=1}^n w_i (\mathbf{x}_i - \mathbf{t})(\mathbf{x}_i - \mathbf{t})^T.$$

If high breakdown point estimators are used for $m$ and $s$, and if an appropriate weight function $w$ is chosen, the Stahel-Donoho estimator can achieve the maximum breakdown point.

The disadvantage of this estimator is its high computational cost. Although approximate algorithms have been developed, it will be difficult to deal with high-dimensional data sets that are typical in chemometric applications. On the other hand, unlike the previously discussed robust estimators for multivariate location and covariance, the Stahel-Donoho estimator can handle data sets with more variables than observations, which makes it attractive for chemometrics.

### 3.7.7 Using spatial signs

Also spatial sign covariances can handle data with more variables than observations. They are very fast to compute, have a bounded influence function, can deal with a moderate fraction of outliers in the data, are not relying on the assumption of multivariate normality, but are not affine equivariant. The

spatial sign $\mathbf{S}(\mathbf{x})$ of a multivariate observation $\mathbf{x}$ with respect to the centre $\mathbf{t}$ of the data is defined as

$$\mathbf{S}(\mathbf{x}) = \frac{\mathbf{x} - \mathbf{t}}{\|\mathbf{x} - \mathbf{t}\|}$$

(see[60]). This is a unit vector pointing in the direction $\mathbf{x} - \mathbf{t}$. For the location estimator $\mathbf{t}$ one can take the spatial median which is the solution of minimising $\sum_{i=1}^{n} \|\mathbf{x}_i - \mathbf{t}\|$. The spatial median has maximum breakdown point and is orthogonal equivariant. Another choice for $\mathbf{t}$ could be the co-ordinate-wise median.

At the basis of spatial signs the so-called *spatial sign covariance matrix* can be constructed:

- compute the sample covariance matrix from the spatial signs $\mathbf{S}(\mathbf{x}_1), \ldots, \mathbf{S}(\mathbf{x}_n)$, and find the corresponding eigenvectors $\mathbf{u}_j$, for $j = 1, \ldots, p$, and arrange them as columns in the matrix $\mathbf{U}$,
- project the observations on the $j$-th eigenvector (scores) and estimate robustly the spread (eigenvalues) by using e.g. the MAD,

  $$\lambda_j = \mathrm{MAD}(\mathbf{x}_1^T \mathbf{u}_j, \ldots, \mathbf{x}_n^T \mathbf{u}_j)^2$$

  for $j = 1, \ldots, p$. Arrange them in the diagonal of a squared matrix, i.e. $\mathbf{\Lambda} = \mathrm{diag}(\lambda_1, \ldots, \lambda_p)$,
- The covariance matrix estimate is

  $$\mathbf{C} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T.$$

## 3.8 Projection pursuit

As we have reviewed in the previous sections the basic idea behind the construction of most robust multivariate estimators consists of attributing weights to each data point separately. Weights can be computed in many ways; they can be continuous (e.g. M estimators, S estimators) or binary (estimators based on trimming such as LTS regression). With the computation of weights, the classical estimator is always to some extent a part of the robust estimator process. For example, M regression estimators are computed from iteratively re-weighted data. Within each iterative re-weighting step the classical estimator is computed. For LTS regression a subset of $h$ cases is sought for. Once this subset is found, the classical estimator is computed on the reduced data set.

An entirely different approach to robustifying multivariate methods consists of projection pursuit. Initially, projection pursuit was developed as a data reduction technique for high dimensional data.[21] However, in a short time span application of projection pursuit has spread throughout many areas of statistics, such that in 1985 a review article could already report projection pursuit based approaches to density estimation, regression, estimation of the covariance structure and principal component analysis.[28] Its relative popularity can be explained by the method's versatility: depending on which criterion one uses to evaluate the projections (the *projection index*), a projection pursuit algorithm can yield (approximate) estimates to very different approaches. In practice, if a single projection pursuit algorithm is established, it can be used almost directly to produce a manifold of estimators.

The basic idea of projection pursuit consists of reducing a problem of an intrinsically multivariate nature to many univariate problems by dint of projection. If the data are $p$ variate then a projection pursuit algorithm encompasses the following steps: construct all possible $p$ vectors (*directions*) and evaluate for each of these the projection index. The direction which yields the optimal value for the projection index is the solution. For instance, principal components are components which capture a maximum of variance. So in theory they can be found by computing all possible $p$ vectors (denoted $\boldsymbol{a}$) and then computing the variance of $\mathbf{Xa}$. The vector $\mathbf{a} \in \Re^p$ yielding the maximal value for var($\mathbf{Xa}$) is the first principal component. By evaluating a robust measure of spread (e.g. a scale M estimator, see Section 3.3.2), a robust method for principal component analysis is readily obtained.

Of course, in practice only a finite number of directions can be constructed. Hence, the projection pursuit approach to all multivariate estimation procedures always yields approximate solutions. The quality of the approximation depends on the number of directions evaluated. The obtained solution evidently also depends on the choice of the directions which are scanned. Several algorithms have been proposed in literature. One can choose to construct directions randomly. However, in doing so one disregards the data structure due to which it is hard to ascertain that based on a small set, the obtained solution will be a good approximation to the true solution. A second approach consists of taking the $n$ directions contained in the data set as directions to evaluate and if necessary to augment these directions by random linear combinations of the original directions. This algorithm has been adopted successfully for principal component analysis[4] as well as for continuum regression.[53] A third approach is a so-called *grid algorithm*. The grid algorithm restricts the search for the optimum to a plane. Its consists of the following steps:

1. Compute for each variable $\mathbf{x}_i$ the projection index based on the $n$ data points. This yields $p$ values for the projection index.

2. Sort the variables in descending order according to the value they yield for the projection index.

3. Find the optimum direction in the plane spanned by the first two sorted variables. This yields the first approximation to the optimal direction: $\mathbf{a}^{(1)} = (\gamma_1^{(1)} \ \gamma_2^{(1)} \ \mathbf{0}_{p-2}^T)$.

4. For $j = 3 : p$, find the optimal directions in the plane spanned by the vectors $\mathbf{Xa}^{(j-1)}$ and the $j$th sorted variable $\mathbf{x}_{(j)}$. The next solution is then given by:

$$\mathbf{a}^{(j)} = \left(\prod_{k=1}^{j} \gamma_1^{(k)} \quad \gamma_2^{(1)} \prod_{k=2}^{j} \gamma_1^{(k)} \quad \gamma_2^{(2)} \prod_{k=3}^{j} \gamma_1^{(k)} \quad \cdots \quad \gamma_2^{(j)}, \quad \mathbf{0}_{p-j}^T\right). \tag{24}$$

5. When all variables have passed the previous phase, restart evaluating all entries $a_i$ of $\mathbf{a}$ by searching the optimal direction in the plane spanned by the vectors $\mathbf{Xa}^{(j-1)}$ and $\mathbf{x}_j$. Stop the iteration if $\parallel \mathbf{a}^{(q)} - \mathbf{a}^{(q-1)} \parallel$ is smaller than a certain tolerance limit (e.g. $10^{-5}$).

To find the optimal directions in the plane itself (required above in steps 3 through 6), the following algorithm is used:

1. Consider a limited set of linear combinations of both variables $\gamma_1 \mathbf{x}_1 + \gamma_2 \mathbf{x}_2$ and evaluate the projection index for each of these directions. The directions are chosen on a unit sphere (the side constraint $\gamma_1^2 + \gamma_2^2 = 1$ should be satisfied) at regular intervals. For instance, if ten initial guesses are chosen these directions are at 0, 18, 36, ..., 162 degrees.

2. Project the data onto the initial optimum.

3. Scan the same number of directions in a narrower interval, e.g. for ten directions: -45, -35, ..., 45 degrees. The angle in which the grid search is effectuated is made narrower until convergence is reached.

The grid algorithm has shown to be successful for principal component analysis,[6] where it is more precise than the algorithm based on choosing data points and linear combinations. An implementation of it for continuum regression has also been reported.[19]

Although much research has been carried out on the algorithmic aspect of projection pursuit, a draw-back of the method is still its computational cost. With any of the algorithms above, still a large number of directions need to be constructed for the projection pursuit approximation to be reliable. As computer power is continuously increasing, at the moment the computational cost for "normal" data sets encountered in chemometrics (e.g. size $100 \times 2000$) is not excessively high to construct the estimator once. Howbeit, for many applications of projection pursuit, a single computation of the estimator is not sufficient. For instance, if the estimator needs to be cross-validated, in each cross validation loops the estimator needs to be evaluated. Depending on the application, including a projection pursuit based estimator into a cross-validation routine may still require long computation times.

# 4 Robust alternatives to principal component analysis

Recall that principal component analysis (PCA) proceeds by finding directions in space which maximise or minimise the dispersion, measured by the variance. The classical approach is based on the covariance matrix. The first principal direction is the unit vector $\mathbf{b}_1$ such that $\mathrm{var}\,(\mathbf{X}\mathbf{b}_1) = \max$. The directions $\mathbf{b}_j$ for $j > 1$ are the unit vectors such that $\mathrm{var}\,(\mathbf{X}\mathbf{b}_j) = \max$ under the restriction that $\mathbf{b}_j^T \mathbf{b}_k = 0$ for $k < j$. Call $\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_p$ the eigenvalues of $\mathbf{\Sigma}$ in descending order, and $\mathbf{e}_1, ..., \mathbf{e}_p$ the respective eigenvectors. Then it is shown that $\mathbf{b}_j = \mathbf{e}_j$ for $j = 1, .., p$. Since $\lambda_j = \mathrm{var}\,(\mathbf{X}\mathbf{b}_j)$, the number $q$ of components is usually chosen so that the "proportion of unexplained variance"

$$u_q = \frac{\sum_{j=q+1}^{p} \lambda_j}{\sum_{j=1}^{p} \lambda_j}$$

is sufficiently small (say 10%).

PCA may be viewed geometrically as searching for a $q$-dimensional linear manifold that "best" approximates the data. The point of the manifold closest to $\mathbf{x}_i$ is its "$q$-dimensional reconstruction"

$$\hat{\mathbf{x}}_i = \mathbf{B}\mathbf{B}^T(\mathbf{x}_i - \bar{\mathbf{x}}) + \bar{\mathbf{x}} \tag{25}$$

where $\bar{\mathbf{x}}$ is the data average and $\mathbf{B}$ is the orthogonal $p \times q-$matrix with columns $\mathbf{b}_1, ..., \mathbf{b}_q$.

Outliers in the data may uncontrollably alter the directions $\mathbf{b}_j$ and/or the eigenvectors and hence the choice of $q$. There are many proposals to overcome this difficulty. The simplest is to replace the covariance matrix with a robust dispersion matrix. The eigenvectors of this robust dispersion matrix will result in robust principal components. Croux and Haesbroeck (2000)[3] derived influence functions and asymptotic variances for the robust estimators of eigenvalues and eigenvectors.

Another approach is to maximise a robust dispersion measure instead of the variance. This idea was proposed by Li and Chen (1985).[36] Croux and Ruiz-Gazen (2005)[5] derived theoretical properties of the estimators for the eigenvectors, eigenvalues and the associated dispersion matrix. They introduced an algorithm for computation which was improved by Croux et al. (2007).[6]

Rather than describing the many procedures proposed in the literature, we give two very simple methods.

The first was proposed by Locantore et al. (1999)[37] and is called *spherical principal components* (SPC). Let $\hat{\boldsymbol{\mu}}$ be a robust location vector. Let

$$\mathbf{y}_i = \frac{\mathbf{x}_i - \hat{\boldsymbol{\mu}}}{\|\mathbf{x}_i - \hat{\boldsymbol{\mu}}\|},$$

(see Section 3.7.7). That is, the $\mathbf{y}_i$s are the $\mathbf{x}_i$s shifted to the unit spherical surface centred at $\hat{\boldsymbol{\mu}}$. Compute the cross-products matrix of the $\mathbf{y}_i$s

$$\mathbf{C} = \sum_{i=1}^{n} \mathbf{y}_i \mathbf{y}_i^T$$

and its eigenvectors $\mathbf{b}_j$ $(j = 1, ..., p)$. Let $\hat{\sigma}(.)$ be a robust dispersion measure like the MAD, and define $\lambda_j = \hat{\sigma}(\mathbf{X}\mathbf{b}_j)^2$. Sort the $\lambda_j$s in descending order (and the respective $\mathbf{b}_j$s accordingly). Then proceed as in the classical case. It is shown that in the case of an elliptic distribution, the $\mathbf{b}_j$s estimate the eigenvectors of the covariance matrix (but not necessarily in the correct order).

A simple way to obtain $\hat{\boldsymbol{\mu}}$ is the coordinatewise median:

$$\hat{\boldsymbol{\mu}} = (\mu_1, .., \mu_p)^T \text{ with } \mu_j = \text{med}_i(x_{ij}). \tag{26}$$

A better one is the space median, which is

$$\hat{\boldsymbol{\mu}} = \arg\min_{\boldsymbol{\mu}} \sum_{i=1}^{n} \|\mathbf{x}_i - \boldsymbol{\mu}\|, \tag{27}$$

see also Section 3.7.7.

It is easy to compute $\hat{\mu}$ iteratively. Start from some initial $\mu_0$ (e.g., the coordinatewise median). At iteration $k$, let $w_i = 1/\|\mathbf{x}_i - \boldsymbol{\mu}_k\|$ $(i = 1, .., n)$ and compute $\mu_{k+1}$ as the mean of the $\mathbf{x}_i$s with weights $w_i$.The procedure converges quickly.

A more efficient robust PCA estimator (Maronna, 2005)[40] is as follows. Given $q$, let $\mathbf{B}_0$ be a $p \times q$-matrix of principal directions and $\boldsymbol{\mu}_0$ a robust location

vector. For instance $\mathbf{B}_0 = [\mathbf{b}_1, ..., \mathbf{b}_q]$ where the $\mathbf{b}_j$s are the first principal directions given by the SPC method, and $\boldsymbol{\mu}_0$ is the vector $\hat{\boldsymbol{\mu}}$ in (26) or (27). At iteration $k+1$, compute the reconstructions $\hat{\mathbf{x}}_i = \mathbf{B}_k \mathbf{B}_k^T (\mathbf{x}_i - \boldsymbol{\mu}_k) + \boldsymbol{\mu}_k$, and the distances $r_i = \|\mathbf{x}_i - \hat{\mathbf{x}}_i\|^2$. Let $\hat{\sigma}$ be an M-scale (15) of $(r_1, ..., r_n)$. Let $w_i = W_\sigma(r_i/\hat{\sigma})$ with $W_\sigma$ defined in (16). Let $\boldsymbol{\mu}_{k+1}$ and $\boldsymbol{\Sigma}_{k+1}$ be the mean and covariance matrix of the $\mathbf{x}_i$s with weights $w_i$, and let $\mathbf{B}_{k+1}$ be the first $q$ eigenvectors of $\boldsymbol{\Sigma}_{k+1}$. And so on. Simulations have shown that this procedure is robust and is more efficient than SPC for normal data.

# 5 Robust alternatives to partial least squares

## 5.1 A brief introduction to PLS

Partial least squares (PLS) is one of the most successful tools in chemometrics. Historically it started as a method to estimate structural relations between several blocks of variables,[63] in which sense it is applied in the fields of marketing and econometrics.[58] However, when a structural PLS model is set up between two groups of variables, it can also be used for prediction and is thus a regression technique. In chemometrics, virtually all applications of PLS fall in the latter category. Hence, in the current section we will limit ourselves to partial least squares for models between two groups of variables, $\mathbf{x}$ and $\mathbf{y}$. It will be assumed that these variables are of dimensions $p$ and $q$, respectively and are assumed to be centred. The data are a set of $n$ samples measured at these variables.

Data from chemometrics are often of a high dimensional nature. Usually $p$ is big and may exceed $n$ (this applies to most spectrophotometrical applications); moreover the $p$ variables in $\mathbf{x}$ may be multicollinear. For such data it is well known that the least squares estimator fails. A viable approach to overcome these problems is first to estimate a new set of uncorrelated latent variables $\mathbf{t}$ and $\mathbf{u}$ from the original variables, between which standard regression can be carried out. Depending on how the latent variables are defined, different regression techniques are obtained.

In virtually all regression methods based on the estimation of latent variables, the latter are defined as linear combinations of the original variables: for all $i \in \{1, \min(n,p)\}$ it holds that $\mathbf{t}_i = \mathbf{X}\mathbf{v}_i$ and $\mathbf{u}_i = \mathbf{Y}\mathbf{w}_i$. In the special case of PLS, the latent variables are defined by the following criterion:

$$(\mathbf{v}_h, \mathbf{w}_h) = \underset{\mathbf{a},\mathbf{b}}{\operatorname{argmax}} \left(\operatorname{cov}\left[\mathbf{X}\mathbf{a}_h, \mathbf{Y}\mathbf{b}_h\right]\right) \tag{28a}$$

under the constraints that

$$\| \mathbf{a}_h \| = 1 \quad \text{and} \quad \| \mathbf{b}_h \| = 1 \tag{28b}$$

and

$$\mathbf{a}_h^T \mathbf{X}^T \mathbf{X} \mathbf{a}_i = 0 \quad \text{for } 1 \leq h < i. \tag{28c}$$

It is clear from the criterion that PLS can be seen as being a compromise between principal component analysis and regression. The criterion can be solved by dint of the Lagrange multiplier method, leading (among other results) to the conclusion that the X weighting vectors $\mathbf{v}_i$ are successive eigenvectors of the matrix $\mathbf{X}^T \mathbf{y} \mathbf{y}^T \mathbf{X}$. For the other entities engaged in the criterion (e.g. $\mathbf{w}$), analogous eigenvector relations can be determined.

Let $\mathbf{U}$ and $\mathbf{T}$ denote the matrices which collect the above vectors $\mathbf{u}_i$ and $\mathbf{t}_i$ in their columns. PLS regression resides in the idea of carrying out a regression of $\mathbf{U}$ on $\mathbf{T}$ (which does not lead to problems since the number of latent variables $k$ is always $k < n$ and they are by definition uncorrelated). Although the regression is set up between the latent variables, it is possible to re-write the formulæ in such a way that a direct relation between $\mathbf{X}$ and $\mathbf{Y}$ is obtained:

$$\hat{\mathbf{Y}} = \mathbf{X}\mathbf{V}\left(\mathbf{V}^T\mathbf{X}^T\mathbf{X}\mathbf{V}\right)^{-1}\mathbf{V}^T\mathbf{X}^T\mathbf{Y}. \tag{29}$$

Here, $\mathbf{V}$ is the matrix with the weighting vectors $\mathbf{v}_i$ in its columns. In order to do prediction, this equation is very practicable. From it, it can be seen as well that it is possible to define a matrix of PLS regression coefficients:

$$\hat{\mathbf{B}} = \mathbf{V}\left(\mathbf{V}^T\mathbf{X}^T\mathbf{X}\mathbf{V}\right)^{-1}\mathbf{V}^T\mathbf{X}^T\mathbf{Y}. \tag{30}$$

PLS uses the classical covariance in its definition (see criterion (28a)). The classical covariance is a nonrobust estimator; as all PLS estimates derive from this classical covariance it can be expected that the whole PLS procedure is nonrobust. Indeed, if one takes into consideration the PLS influence function (first derived by Serneels et al.[49]), it can be seen that the PLS influence functions are unbounded and thus that PLS is nonrobust.

Prior to proceeding to robust PLS, we note that PLS is a specific estimator fitting into a more general framework called *continuum regression*. The continuum embraces the whole range of regression methods between ordinary least squares (OLS) and principal component regression (PCR), PLS being half-way. However, as OLS, PCR and PLS are the only three methods from the continuum regression framework for which the maximization can be solved analytically, PLS comes in many cases out as the best compromise between modelling predictor variance (PCR), modelling relation to the predictand (OLS) and computational simplicity.

## 5.2 Robustifying PLS

Several approaches are possible to obtain a robust alternative to PLS. By analogy to PCA, it is possible to do projection pursuit or to use a robust estimator for covariance. By analogy to regression, it is also possible to make partial versions (i.e. latent variables based versions) of robust regression estimators such as least absolute deviation (LAD) or robust M regression. In what follows, the key principles of the existing robust PLS methods, will be outlined.

### 5.2.1 Projection pursuit

The power of projection pursuit (PP) is that it reduces an essentially multivariate problem to many ones of a bivariate nature: it suffices to replace the classical covariance in criterion (28a) by a robust estimator for covariance. By such an order of proceeding one obtains robust estimates for the weighting vectors. The scores, however, again contain the outliers (due to multiplication with $\mathbf{X}$). Thus, in order to obtain robust regression coefficients, a robust regression estimator needs to be used to perform the regression between the latent variables.

A robust PLS regression estimator has hitherto only been published for univariate PLS regression (i.e. for the case where $q = 1$), as a part of the robust

continuum regression (RCR) framework.[53] Robust PLS is obtained there by setting the continuum parameter $\delta$ to 0.5. The projection index proposed by the authors is the trimmed covariance. In the final step, a robust M regression is performed to obtain the regression coefficients.

Robust PLS as a part of the RCR framework can deal with high dimensional data and is robust both with respect to vertical outliers and leverage points. Its theoretical robustness properties have hitherto not been investigated. By analogy to projection pursuit PCA[5] one can expect that the estimated weighting vectors will inherit the robustness properties of the projection index used. In this case this implies that the breakdown point can be expected to equal the percentage of trimming used. One can also expect the influence function to be bounded but nonsmooth, as is typical for estimators based on trimming. For the regression coefficients, however, this can be expected not to carry through as they also depend on the final robust M regression step. Further theoretical developments will shed more light on its properties.

A final disadvantage is that RCR may be computationally slow, if it is needed to insert it in a cross-validation procedure. In that case it may be advisable to use a low number of PP directions during cross validation, and to compute a more precise estimate for calibration. Both RCR and the cross-validation procedure are publicly available as a part of the TOMCAT toolbox.[8]

### 5.2.2 Robust covariance estimation

It can be shown that all PLS estimators derive from two basic population entities: the shape of $\mathbf{X}$ and the covariance between $\mathbf{X}$ and $\mathbf{y}$. In fact, if one considers the augmented data $\mathbf{Z} = (\mathbf{X}, \mathbf{y})$, then these properties are summarised in the shape of $\mathbf{Z}$. It is well known that the covariance matrix of $\mathbf{Z}$ takes on the partitioned form

$$\mathbf{\Sigma_Z} = \begin{pmatrix} \mathbf{\Sigma_X} & \mathbf{\Sigma_{Xy}} \\ \mathbf{\Sigma_{Xy}}^T & \sigma_\mathbf{y}^2 \end{pmatrix}. \tag{31}$$

Hence, as all PLS estimators derive from $\mathbf{\Sigma_X}$ and $\mathbf{\Sigma_{Xy}}$, it suffices to use robust covariance estimates for these (or, more practicably, for $\mathbf{\Sigma_Z}$) in order to obtain a robust PLS procedure.

The approach of plugging in a robust covariance estimator has been explored. Both the affine equivariant robust and semi nonparametric covariance matrix estimators have been examined in the context of PLS.

**Robust PLS based on the Stahel-Donoho estimator** Gil and Romera[23] propose to plug in the Stahel-Donoho or Minimum Volume Ellipsoid estimators for covariance (see Section 3.7), but they prefer the Stahel-Donoho estimator based on previous results (theoretical and simulation) by Maronna and Yohai.[42]

The robustness properties of the whole PLS procedure are not known. However, it has been shown that just like for PLS itself, all PLS influence functions derive from the influence functions of both covariance estimates involved. For the robust procedure this is also true such that the influence function will be behaving similar to the influence function of the robust covariance estimator plugged in. In this case one may expect the influence function to be analogous to the influence function of the Stahel-Donoho estimator. Statistical efficiency and breakdown have not yet been investigated for this method.

A major drawback to the method is that it is only fit for data for which $n > p$, hence precluding almost any application to spectrophotometry. A full algorithm is not publicly available.

**Robust PLS based on the sign covariance matrix** Another approach is to use a sign covariance matrix,[60] see Section 3.7.7. The sign covariance matrix is a semi nonparametric covariance matrix estimate which is an attempt to generalise the bivariate correlation estimators such as the Spearman and Kendall correlation to a multivariate estimator. The sign covariance matrix is based on the concept of the spatial sign, and is the simplest of six proposals of sign covariance matrices made by Visuri et al.[60] They finally prefer a covariance matrix based on the Oja median instead of the spatial sign because the former is affine equivariant. Howbeit, as the PLS method is not affine equivariant as such, this property is not a prerequisite for the construction of a robust PLS method. Hence, it is possible to use the spatial sign covariance matrix instead, without loss of good properties. In fact, using the spatial sign covariance matrix has two advantages over the method based on Oja medians:

1. the spatial sign covariance matrix has a very simple mathematical definition, which implies that the mathematical treatment, but above all, the computational algorithm, becomes very simple;

2. in contrast to the method based on Oja medians, the spatial sign covariance matrix has a bounded influence function, such that the resulting PLS procedure will be robust.

Recollect that the spatial sign covariance matrix consists of a transformation of the data to their spatial signs, followed by a computation of the (classical) covariance matrix. As all PLS estimators derive from the covariance estimator, the same carries through to the whole PLS procedure, i.e. a robust PLS based on the spatial sign covariance matrix is equivalent to the following steps:

1. Transform each observation in the data to its spatial sign, i.e. replace each row $\underline{x}_i$ from $\boldsymbol{X}$ by

$$\text{sgn}(\underline{x}_i) = \begin{cases} \underline{x}_i / \parallel \underline{x}_i \parallel & \text{if } \underline{x}_i \neq \underline{0}, \\ \underline{0} & \text{if } \underline{x}_i = \underline{0}, \end{cases} \tag{32}$$

   where underlined characters denote row vectors;

2. carry out PLS on the transformed data.

Thanks to this property robust PLS based on a spatial sign covariance matrix becomes extremely efficient in the computational sense: in terms of computational efficiency, it is the fastest existing robust alternative to PLS. Because the method consists of a transformation of the data prior to normal PLS, the projection to the spatial sign can be seen as a form of data preprocessing. Hence, the method also carries the name of *spatial sign preprocessing* ([55]).

The influence function of the spatial sign transformation has been determined.[55] In fact, it is analogous to the influence function of PLS: a sequential set of influence functions each of which derive from the influence function of

the spatial sign covariance matrix. The influence functions are bounded and smooth.

The breakdown and efficiency properties have not been theoretically investigated but simulation results are available, which will be discussed in Section 5.3.

The major drawback of the method is that its robustness properties can not be chose by the user: the method does not depend on any tuneable parameter.

### 5.2.3 Robust PLS by robust PCA

It has been heeded in Section 5.1 that the computation of the PLS estimators comes down to eigenvector and eigenvalue computations. Since principal component analysis corresponds to an eigenanalysis of the covariance matrix; PLS can thus be seen as PCA applied to covariance matrices of a special type such as $\mathbf{X}^T\mathbf{Y}\mathbf{Y}^T\mathbf{X}$.

A possible means to create a robust version of PLS resides thus in using a robust PCA method for these purposes. This approach has been followed by Hubert and Vanden Branden (2003), who have proposed the RSIMPLS method.[30] A full mathematical treatment would be too exhaustive for this summary, but the method can vaguely be described as using the ROBPCA method[32] to compute the PLS components and adapting the algorithm such that further steps are consistent with the SIMPLS algorithm.[12]

The method is entirely robust, both with respect to vertical outliers and leverage points. The algorithm is fairly fast in terms of computation such that high dimensional data are tractable with RSIMPLS. An implementation is publicly available as shareware as a part of the LIBRA toolbox (a MATLAB Library for Robust Analysis).[59]

The robustness properties of the method are not fully known. The influence function of the ancillary ROBPCA method is known and is shown to be bounded but nonsmooth (reflecting the two estimation stages in the ROBPCA algorithm).[11] The influence function of a very closely related method for robust PLS has been established by the same authors,[11] showing analogous behaviour. They do not describe the true influence functions of RSIMPLS for reasons of mathematical tractability but no surprises seem to be expected: one can assume the RSIMPLS influence function to be as well bounded but nonsmooth. The method has a tuneable parameter which presumably determines the breakdown point of the estimator. Simulations (see next section) corroborate these assumptions.

### 5.2.4 Robust PLS as a partial version of robust regression

The name partial least squares suggests that PLS be a partial version of the least squares regression estimator. One way to interpret this statement is by seeing the word *partial* in the sense of being *restricted to the space spanned by the latent variables* $\mathbf{t}_i$. Indeed, PLS regression consists of an estimation stage of the latent variables followed by a regression between these latent variables (or of $\mathbf{y}$ on $\mathbf{T}$ if univariate).

A straightforward approach to robustify PLS thus consists of constructing a partial version of a robust regression estimator. Two such estimators have been proposed: the partial least absolute deviation (PLAD) estimator[15] and

the partial robust M (PRM) regression estimator.[52] The latter is more interesting since it leads to a computationally simple method. It is well known that computation of M estimators for regression can be completed with the use of an iteratively reweighted least squares algorithm.[13] Hence, to compute the partial robust M regression estimator, it suffices to perform an iteratively re-weighted partial least squares algorithm, analogous to the earlier IRPLS method,[7] but by using robust starting values and by using weights which depend on distances both in the residual and score spaces.[52] This makes the method robust with respect to both vertical outliers and leverage points.

The theoretical robustness properties of the PRM regression method are not yet known. The method has a tuneable parameter and in fact, could easily be generalised to different weighting schemes than the one proposed by the authors. Simulations on the original proposal indicate that the method combines a good efficiency to a high robustness. In terms of computation the method is fairly fast and can handle high dimensional data sets, but is still outperformed by robust PLS based on the sign covariance matrix (spatial sign preprocessed PLS).[55]

A drawback to the method is that models are not nested, i.e. a three-component PRM model on a given dataset is not identical to the first three components of a 4 component model on the same data.

## 5.3 Simulation results

As for most robust alternatives to PLS, the theoretical criteria to assess its robustness properties such as the influence function, the MaxBias curve and the statistical efficiency are unknown, one has to rely on simulation results in order to compare the different methods. Important simulation results have first appeared in.[55] The results reported there are summarised in this section. These results concern the methods of partial robust M regression,[52] RSIMPLS[32] and robust PLS based on a sign covariance matrix (or spatial sign preprocessed PLS),[55] as these are the methods which are most suited to chemometric applications. PP-PLS (or RCR at $\delta = .5$) is left out because of computational complexity, whereas the Gil-Romera method[23] is not considered because it is not fit for data where the number of variables exceeds the number of cases.

### 5.3.1 Statistical efficiency

The basic setup of a simulation for statistical efficiency is the following: generate a data matrix according to the latent variables regression model such that the complexity is known (i.e. one knows from beforehand the optimal number of latent variables $k$ to use in the estimation phase):

$$\mathbf{X} = \mathbf{T}_k \mathbf{P}_k^T + \mathbf{E}. \tag{33}$$

If leverage points are to be included, a certain amount of rows from the simulated data matrix, corresponding to the desired fraction of contamination, is replaced by bad outliers. Based on the simulated predictor matrix, the dependent variable (predictand) is simulated according to the regression model

$$\mathbf{y}_m = \mathbf{X}\boldsymbol{\beta} + \varepsilon_m, \tag{34}$$

where the vector of regression coefficients $\boldsymbol{\beta}$ is known and $\varepsilon$ is a random error term taken from a considered distribution. The distributions under consideration can in theory be of any possible type, but usually three types are chosen: the normal distribution, a distribution different from the normal with wider tails and finally a distribution which is likely to generate many points far away from the centre (outliers). A large number of predictands is to be simulated in this manner (depending on the computational cost, this large number may equal $m = 200,\ m = 1000,\ m = 10000, \ldots$).

Evaluating the efficiency of a given method can now be done as follows: for each of the constructed pair of data matrices $\boldsymbol{X}, \boldsymbol{y}_m$, one estimates the vector of regression coefficients $\hat{\boldsymbol{\beta}}_m$ by dint of the methods one wants to evaluate, and then compares that estimate to the true vector of regression coefficients which is known from the simulation setup. Using a mean squared error (MSE) ($\text{MSE} = 1/m \sum_{i=1}^{m} (\hat{\boldsymbol{\beta}}_m - \boldsymbol{\beta})^T (\hat{\boldsymbol{\beta}}_m - \boldsymbol{\beta})$) the former comparison can be formalised. It is often reported that data dimensions influence the estimators' performance; hence several data structures ought to be considered: at least a distinction should be made between the cases $n < p$ and $p < n$, but more dimensionalities can be taken into account. Also, it should be investigated if the optimal complexity $k$ influences the results.

In Reference[52] a comparison is made between the methods PLS, PM, PRM and RSIMPLS (PM is a simpler version of PRM where no weighting in the score space is done). Both vertical outliers and leverage points are considered. The error distributions considered are the normal, Laplace and $t_5$ (generating a moderate fraction outliers) and the Cauchy and Slash distributions (heavily tailed). The data structures range from one to three latent variables and the data dimensions vary from $n/p = 4$ to $n/p = 0.1$. In Reference[55] PLS, PRM, Spatial sign pre-processed PLS and RSIMPLS are compared with the same error terms, but with data dimensions now ranging to $n/p = 0.01$. Leverage outliers are in this publication not considered.

The general trends to be observed from these simulations are the following. At first, as can be expected, the classical partial least squares estimator performs best at uncontaminated data. At normal models, robust estimators are known to be less efficient than classical estimators: they suffer from an increase in variance. It is clear that in all data structures considered the robust estimators do indeed show an increased variance compared to PLS. However, there is a significant difference among the robust estimators: the PRM estimator hardly loses efficiency with respect to PLS whereas the RSIMPLS estimator clearly does. The simple spatial sign based estimate performs surprisingly well, being in between both other estimators.

As soon as distributions deviate from the normal, the robust estimators take over the lead as they suffer much less from biasedness due to nonnormality. At the Laplace and $t_5$ distributions the effects are not clear but at the $t_2$, Cauchy and Slash distributions (models which generate a lot of extreme values) the classical estimator (PLS) completely breaks down whereas the robust estimators are still reliable. A further general trend is that the PRM seems to perform best, followed by the spatial sign preprocessed method and RSIMPLS. But at the last data configuration, which is common in chemometrics ($n/p = .01$), the differences between the robust estimators are small. They all are capable to counter the effect of nonnormality. the spatial sign preprocessed PLS estimator

seems to depend most on data dimensions, and its properties become worse as the number of predictors increases.

### 5.3.2 Bias and breakdown

As stated before, theoretical results regarding the MaxBias curve and breakdown point of the robust PLS estimators, has not yet been investigated. However, a study has been carried out to simulate the construction of a bias curve numerically.[55] Of course, the results shown there are not exact but they give a good indication as to what the theoretical MaxBias and breakdown properties of the various robust alternatives to PLS may be.

The results of the simulation study cited above lead to the following conclusions. At first, the PLS estimator completely breaks down even at the lowest considered percentage of bias contamination. Partial robust M regression and RSIMPLS show "normal" behaviour for a robust estimator: up to a certain amount of bad contamination, the bias is low whereas at a certain point it explodes. That point can then be considered to be the breakdown point of the estimator. With the settings considered in the simulation study, it is observed that PRM breaks down at 50% of bad contamination whereas RSIMPLS breaks down at 25% of bad contamination. Note that both methods have a tuneable parameter: these results depend on the settings chosen. In the simulation, the values for both tuneable parameters were those which are the standard values in the LIBRA[59] and TOMCAT[8] toolboxes (for RSIMPLS and PRM, respectively). Note furthermore that the observed breakdown counters a conjecture by Frosch-Møller et al.,[22] who assume that PRM's breakdown point equals maximally 30%, providing references to work which does not deal with latent variable methods nor concerns the same type of M estimators.

As regards spatial sign pre-processed PLS, the results look quite different. It shows an atypical simulated bias curve, having already a high degree of contamination for low amounts of contaminated data points. However, the method does not entirely break down until all data consist of bad contamination, awkwardly suggesting that the method has a breakdown point of 100%. Although this result may sound strange, it is not surprising since the spatial sign covariance matrix should be a multivariate generalisation of sign based correlation estimators. For these it has recently become known that, given a correct definition of the breakdown point is used, it equals 100% (see rejoinder in a discussion paper by Davies and Gather[10]).

## 5.4 An application

### 5.4.1 Application of robust techniques

In contrast to theoretical statistics, chemometrics is a field of research driven by applications. Robust statistics and thus robust chemometric methods can still be considered to form a niche inside the field of chemometrics, basically because practical situations in which application of robust techniques are appropriate, are much less common then applications where they are not needed. Hence, at present the commercial software which provides tools typically built for chemometric applications, does not include robust estimation methods. Nevertheless, though suitable applications of robust methods are more rare those of classical

chemometric methods, there exist different experimental situations in which a gain can be made by applying robust tools instead of the classical ones. Situations in which robust methods are worthwhile can be of varying natures, but the most common occurrences fall into three categories:

1. A group of outliers is present in the data. This likely, but not necessarily, causes the masking effect (see Section 1.3). Due to the masking effect, usage of classical chemometric tools for outlier detection will not detect any outliers. Ensuing application of classical calibration or classification methods will not be successful as these techniques will be influenced by the undetected group of outliers. In this respect the benefits of using a robust method are twofold: (i) The robust method can be used to efficiently detect the outliers prior to classical calibration or (ii) The robust method can be used to perform classification or calibration as such, since it is robust to the presence of outliers. Depending on the robustness properties of the robust estimator as well as on the time available for the analysis, one can opt for either of both strategies (the second option inevitably causes a wider uncertainty due to loss in efficiency). But we note that deletion of suspect points may also cause a loss in efficiency.

2. The process of data generation can be controlled well, but there is a wide source of natural variation in the data which cannot be excluded from the analysis. In such a case the distribution is likely to have wider tails than the normal and maybe to generate some outliers. Good examples for this type of data are data concerning e.g. biological organisms, where Nature's variation may be larger than in laboratory controlled experiments. If the data are more likely to follow e.g. a $t$ distribution instead of a normal distribution, application of robust estimators may already yield better results than applications of the classical estimator as the latter is only optimal at the normal model, but performance of the classical estimators tend to deteriorate fast as then distribution differs from the normal.

In what follows we will consider an example which falls into both categories: the benefit of robust calibration is shown to a data set containing a group of outliers for the calibration of one predictand, whereas it can be assumed to be close to normal for calibration to another predictand.

### 5.4.2 Prediction of the concentration of metal oxides in archæological glass vessels

The data set under consideration consists of 180 EPXMA spectra of archæological glass vessels. The experimental conditions under which the spectra were measured, are beyond the scope of the current example, but have been described in detail in.[33] It is noteworthy that the data contains samples which appertain to four different classes of archæological glass: sodic glass, potassic glass, calcic glass and potasso-calcic glass. The vast majority of samples belongs to the sodic group as only 10, 10 and 15 samples belong to the other three classes, respectively. Of the raw data set, the last 25 samples were identified to be outliers as they had been measured with a different detector efficiency. Hence, they are outlying in the space of the spectra, not the concentrations. In the statistical sense these spectra are thus leverage points. It is important to note that all

outliers are samples of sodic glass. The detection of these outliers was done by a tedious procedure: each spectrum was evaluated separately. After removal of the outliers, classical partial least squares calibration was performed and was shown to yield good agreement between predicted and true concentrations for a test set.[35]

In the current section we will show how the use of robust statistics could have sped up this procedure, making the manual spectrum evaluation step superfluous, without causing a sizeable loss in prediction capacity. With these purposes the partial robust M regression estimator was applied, as it is both efficient and can cope with leverage outliers in the data.

In order to perform calibration, the data were split up into two sets. A set for calibration was constructed as in Lemberge *et al.*,[35] where it had been decided that the training set should contain 20 sodic samples and 5 samples of each of the other types of glass. In addition to these 35 spectra, six spectra were added which belonged to the group of sodic samples measured at a different detector efficiency (bad leverage points). The remaining samples with correct detector efficiency were left for validation purposes.

In the original analysis, univariate PLS calibration was performed for all of the main constituents of the glass. These are sodium oxide, silicium dioxide, potassium oxide, calcium oxide, manganese oxide and iron (III) oxide. Here we will limit the description by showing the results for the prediction of sodium oxide and iron (III) oxide. The reason for this is the following: the lighter the element[2] for which to calibrate, the bigger the influence of the leverage points. Hence, showing the results for sodium and iron covers the trends which can be observed from this set of models, as sodium is the lightest and iron the heaviest element to be modelled. Why the model for sodium should be affected more by the outliers than the model for iron, can be explained by physics: a decrease in the detector efficiency function is caused by a contamination layer on the detector's surface. The number of X-ray photons which reach the detector, is inversely proportional to the thickness of the contamination layer. However, highly energetic photons will not be absorbed by the contamination layer. The characteristic energies for Na $K_\alpha$ and Fe $K_\alpha$ photons are 1.02 KeV and 6.4 KeV, respectively. Hence, one may expect the peaks corresponding to iron photons to be affected far less by the lower detector efficiency than the sodium peak.

We come to describing the results for this data set. A PLS and a PRM model were constructed for the calibration of the sodium and iron concentrations. In the former model eight latent variables were used, whereas in the latter seven were used, as in.[35] The concentrations of $Na_2O$ and $Fe_2O_3$ were estimated for the validation set. The root mean squared errors of prediction are given in Table 1. We also state the respective RMSEP's obtained by Lemberge *et al.* posterior to removal of the leverage points from all of the data from the training set. The latter are reported in Table 1 under the heading "cleaned" data, in contrast to the "original" data.

It is observed (see Table 1) that indeed for sodium oxide, the classical PLS model is vastly affected by the leverage points, as the root mean squared error of prediction is almost double compared to the cleaned data set. Using a robust method (PRM), the effect of the leverage points can to a big extent be

[2]In EPXMA one observes the characteristic peaks element-wise; it is thus equivalent to write "model for sodium" and "model for sodium oxide".

| | $Na_2O$ | | $Fe_2O_3$ | |
|---|---|---|---|---|
| | Original | Cleaned | Original | Cleaned |
| PLS | 2.66 | 1.26 | 0.14 | 0.12 |
| PRM | 1.50 | – | 0.10 | – |

Table 1: Root mean squared errors of prediction for the EPXMA data set using the PLS and PRM-estimator, once using the original training sample and once using a clean version of the training sample, as in.[35]

Table 2: Overview of the different robust PLS methods discussed in this section with their most important benefit and drawback

| Method | Pro | Contra | Reference |
|---|---|---|---|
| PP-PLS | Part RCR framework | High computational cost | [53] |
| Gil-Romera | Uses Stahel-Donoho | Not for $p > n$ | [23] |
| Spatial sign | Fastest method | Not tuneable | [55] |
| PRM | High efficiency | Not nested | [52] |
| RSIMPLS | Tuneable robustness | Nonsmooth IF | [32] |

countered, although in comparison to the cleaned classical model there is still a non-negligible increase in RMSEP.

For iron (III) oxide, the results also show the expected trends: the "outliers" are not as outlying as in the model for sodium oxide (due to the higher energy of the iron $K_\alpha$ characteristic photons) and thus PLS performs only slightly inferior on the data set containing outliers than on the cleaned data set, whereas PRM surprisingly performs best of all.

What we can conclude from these results is that at least a lot of time could have been gained by usage of a robust method. Depending on the requested accuracy, PRM could have been used directly for calibration or for detection of the outliers, which in both cases would have eliminated the tedious spectrum evaluation step in which the outliers were detected manually.

#### 5.4.3 Summary

The main goal of this section is to present the different existing robust alternatives to PLS and discussion their pros and cons. Hence, in Table 2 all discussed methods are listed, and for each method its most important pro and con are given.

# 6 Robust approaches to discriminant analysis

## 6.1 Discriminant analysis

In discriminant analysis we observe observations coming from several groups or populations. The group membership is known for the observed observations, they form the *training sample*. At the basis of this training sample it is desired to construct discriminant rules which allow to classify new observations with unknown group membership to one of the populations.

Suppose that $p$ characteristics or variables have been measured, and that $n$ observations are available from $g$ different populations $\pi_1, \ldots, \pi_g$. Since the group memberships are known for the $n$ training data, we can split them into the $g$ groups, resulting in subsamples of size $n_1, \ldots, n_g$ with $\sum_{j=1}^{g} n_j = n$. The observations will thus be denoted by $\mathbf{x}_{ij}$ with index $j = 1, \ldots, g$ representing the groups and index $i = 1, \ldots, n_j$ numbering the samples within a group.

We will first consider the *Bayesian discriminant rule* to classify new observations, and later the *Fisher discriminant rule*. Let us assume that the observations from group $j = 1, \ldots, g$ have been sampled from a population $\pi_j$ with an underlying density function $\mathbf{f}_j$, and denote $p_j$ as the prior probability of this group with $\sum_{j=1}^{g} p_j = 1$. If the subsample size $n_j$ reflects the prior probability of a group, $n_j/n$ can be used to estimate $p_j$. However, if the data have not been sampled completely at random from the mixture, this estimate can be quite unrealistic. For the density $\mathbf{f}_j$ we usually assume a $p$-dimensional normal density with mean $\boldsymbol{\mu}_j$ and covariance matrix $\boldsymbol{\Sigma}_j$. The Bayesian discriminant rule assigns an observation $\mathbf{x}$ to that population $\pi_k$ for which the expression $\ln(p_j \mathbf{f}_j(\mathbf{x}))$ is maximal over all groups $j = 1, \ldots, g$. With the assumption of normal distribution this rule translates to maximising the quadratic discriminant scores $d_j^Q(\mathbf{x})$, defined as

$$d_j^Q(\mathbf{x}) = -\frac{1}{2}\det(\boldsymbol{\Sigma}_j) - \frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_j)^T \boldsymbol{\Sigma}_j^{-1}(\mathbf{x} - \boldsymbol{\mu}_j) + \ln(p_j), \tag{35}$$

i.e. an observation $\mathbf{x}$ is assigned to that group $k$ for which the quadratic discriminant score is maximal.

The method discussed so far is called *quadratic discriminant analysis* (QDA) because the discriminant scores used for the assignment of an observation are quadratic in $\mathbf{x}$. Often, however, it can be assumed that the group covariances are equal, i.e. $\boldsymbol{\Sigma}_1 = \ldots = \boldsymbol{\Sigma}_g = \boldsymbol{\Sigma}$. In this case the discriminant rule simplifies to using the linear discriminant scores

$$d_j^L(\mathbf{x}) = \boldsymbol{\mu}_j^T \boldsymbol{\Sigma}^{-1} \mathbf{x} - \frac{1}{2}\boldsymbol{\mu}_j^T \boldsymbol{\Sigma}_j^{-1} \boldsymbol{\mu}_j + \ln(p_j) \tag{36}$$

in an analogous way for assigning an observation $\mathbf{x}$ to a group. The scores are now linear in $\mathbf{x}$ and thus the corresponding method is called *linear discriminant analysis* (LDA).

It is crucial how the parameters are estimated for the discriminant scores (35) or (36). Classically, the population means $\boldsymbol{\mu}_j$ are estimated by the arithmetic means $\bar{\mathbf{x}}_j = \frac{1}{n_j}\sum_{i=1}^{n_j} \mathbf{x}_{ij}$, and the population covariances $\boldsymbol{\Sigma}_j$ by the sample covariances $\mathbf{S}_j = \frac{1}{n_j - 1}\sum_{i=1}^{n_j}(\mathbf{x}_{ij} - \bar{\mathbf{x}}_j)(\mathbf{x}_{ij} - \bar{\mathbf{x}}_j)^T$. As noted above, the prior probabilities can be estimated by the relative frequencies of data points in a particular group. If equal covariances can be assumed (LDA) it is necessary to estimate the joint covariance matrix $\boldsymbol{\Sigma}$. This can be done by a pooled estimate of the group covariance matrices,

$$\mathbf{S}_{pooled} = \frac{1}{n_1 + \ldots + n_g - g}\left((n_1 - 1)\mathbf{S}_1 + \ldots + (n_g - 1)\mathbf{S}_g\right).$$

Thus, the resulting LDA rule is to assign an observation $\mathbf{x}$ to population $\pi_k$ if the estimated LDA score

$$\hat{d}_k^L(\mathbf{x}) = \bar{\mathbf{x}}_j^T \mathbf{S}_{pooled}^{-1} \mathbf{x} - \frac{1}{2}\bar{\mathbf{x}}_j^T \mathbf{S}_{pooled}^{-1} \bar{\mathbf{x}}_j + \ln\left(\frac{n_j}{n}\right) \tag{37}$$

is the largest of all LDA scores $\hat{d}_1^L(\mathbf{x}), \ldots, \hat{d}_g^L(\mathbf{x})$.

## 6.2 Robust LDA

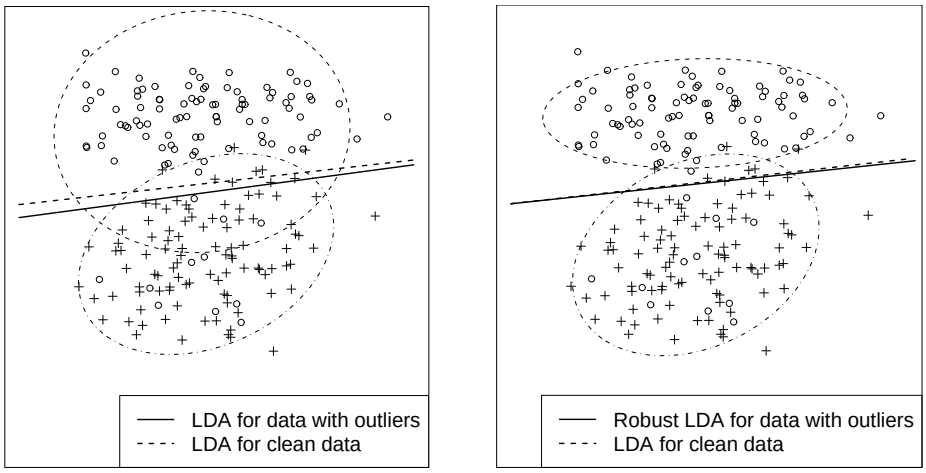
The LDA rule (37) with the classical estimates of location and covariance is vulnerable to outliers, and it can lead to a much higher misclassification error in the sense that the spoiled LDA rule will result in many wrongly classified observations.

The discriminant rules (35) and (36) can easily be robustified by plugging in robust estimates for the group means and covariances. For obtaining a robust LDA rule the joint covariance matrix has to be estimated. There are several proposals for this purpose: one can derive a pooled estimate like in the classical case by averaging the robust group covariances. Another possibility is based on centring the observations of each group by their robust estimate of location, and then to derive a robust estimate of covariance out of all centred observations.[26] In an iterative procedure the group centres can again be updated by the robust location estimate of all observations. Finally, one could obtain a pooled covariance matrix by using weights for the arithmetic group mean and a joint sample covariance matrix which are derived from robust covariance estimation of all data points jointly.[25]

The remaining question is which robust location and covariance matrix should be used for plugging in into the discriminant rules. MCD estimators were used by,[31] and S estimators by,[26] the latter showing slightly better performance for the misclassification probabilities.

To illustrate the effects of classical and robust parameter estimation for LDA and QDA we generate two groups of data with 100 observations each. The data in each group are sampled from bivariate normal distributions with certain means and covariances (unequal). However, only 90 data points of the first group are following the group distribution, but the remaining 10 points are generated with the parameters of the second group. The idea behind this data generation is a scenario where part of the available training data have been assigned incorrectly: The 10% "outliers" in the first group belong to the second group, but have been assigned by an incorrect decision to the first group. Note that this wrong assignment of training data can easily occur in practice, and that a proportion of 10% outliers can usually be considered as rather low.

Figure 10 shows the two groups of data, the first group with $\circ$ and the second with $+$ as symbols. The ellipses shown in the figures are 90% tolerance ellipses using the classical estimates (Figure 10a) and the MCD estimates (Figure 10b). Clearly, the classical tolerance ellipse for the first group is inflated by the outliers. The two straight lines in each picture show the separation lines for the data without outliers (dashed) and for the complete data set (solid). The separation lines are obtained by computing the LDA scores (36) for each point $\mathbf{x}$ in the two dimensional space, and by assigning each point to the corresponding group. Points on the separation line would have equal group assignment. In Figure 10a the parameters for (36) were estimated in the classical way by using formula (37) for the estimated LDA scores. The difference in the resulting separation lines with and without outliers is not big, it leads to 3 additional misclassifications of the second group. The solid separation line in Figure 10b uses MCD estimates for the parameters while the dashed line is obtained by classical estimates on the clean data. The latter line almost coincides with the



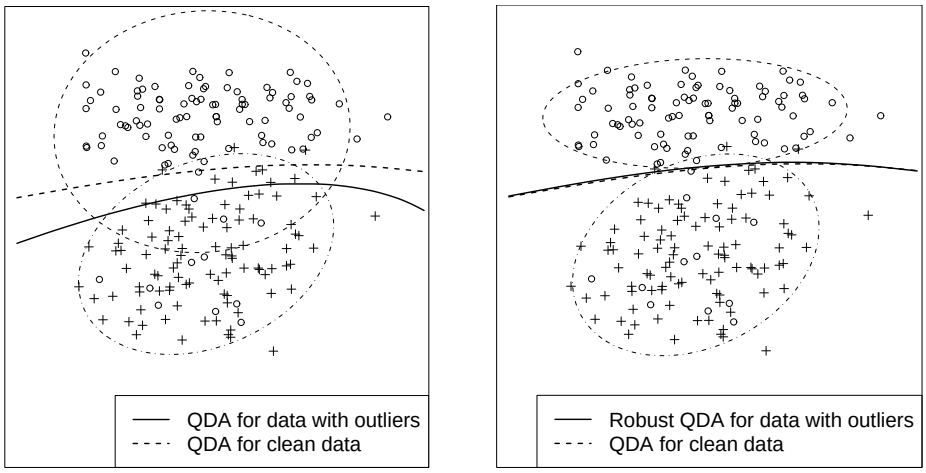

(a) (b)

Figure 10: Two groups of data simulated from bivariate normal distributions with different means and covariances. 10% of the data points from the first group (∘) were sampled from the distribution of the second group (+). LDA separation lines are shown when using classical parameter estimates (a) and robust MCD estimates (b).

line obtained by robust estimation, and thus both solutions result in the same number of misclassifications (which is unavoidable because the groups are overlapping and since we applied LDA to data with different covariance structures).

The resulting separation curves of classical and robust QDA estimation for the same simulated data are shown in Figure 11. Similar to LDA, the QDA separation curves were obtained by estimating the QDA scores (35) with classical and robust MCD estimates, respectively, for any data point in two dimensions, and assigning each data point to the corresponding group. Points on the separation curve indicate equal group membership. Like for LDA, the classical estimation of the QDA scores leads to different solutions if all data were used or if only the non-outlying data were analysed. Here, 8 additional observations were misclassified as a result of using classical estimation for data with incorrect group assignment. The robust estimation gives the same result as the classical estimation on the clean data since the separation curves in Figure 11b practically coincide.

Once more this example underlines the advantages of robust parameter estimation over classical estimation. On purpose we did not construct data with extreme outliers, but simply data with wrong group assignment. Thus, the difference in the results is not dramatical, but still essential. Here, we only considered misclassifications of the training data, but more important will be misclassifications of future observations. Especially for observations which are close to the decision boundary the estimates to be used for the discriminant rule will be important.




(a) (b)

Figure 11: Two groups of data simulated from bivariate normal distributions with different means and covariances. 10% of the data points from the first group (∘) were sampled from the distribution of the second group (+). QDA separation curves are shown when using classical parameter estimates (a) and robust MCD estimates (b).

## 6.3 Robust Fisher LDA

Discriminant analysis was originally introduced by Fisher[20] for the separation of two groups. The extension of this method to several groups was done by Rao.[45] The method leads to linear discriminant functions, and it turns out that the problem can be formulated as an eigenvector/eigenvalue problem with a relatively simple solution. Nevertheless, the method is frequently used in practical applications and turns out to be powerful also in comparison to more advanced methods.

We use the same notation as in the previous sections. Additionally, the overall weighted mean for all populations is denoted by $\bar{\boldsymbol{\mu}} = \sum_{j=1}^{g} p_j \boldsymbol{\mu}_j$. Then the covariance matrix $\mathbf{B}$ describing the variation *between the groups* is defined as

$$\mathbf{B} = \sum_{j=1}^{g} p_j (\boldsymbol{\mu}_j - \bar{\boldsymbol{\mu}})(\boldsymbol{\mu}_j - \bar{\boldsymbol{\mu}})^T,$$

and the *within groups covariance matrix* $\mathbf{W}$ is defined as

$$\mathbf{W} = \sum_{j=1}^{g} p_j \boldsymbol{\Sigma}_j.$$

$\mathbf{W}$ can be seen as a pooled version of the group covariance matrices. Under the assumption that the group covariance matrices are all equal it can be shown that the group centres can be best separated by maximising the expression

$$\frac{\mathbf{a}^T \mathbf{B} \mathbf{a}}{\mathbf{a}^T \mathbf{W} \mathbf{a}} \quad \text{for } \mathbf{a} \in \Re^p, \, \mathbf{a} \neq \mathbf{0}. \tag{38}$$

The solution of maximising (38) is given by the eigenvectors $\mathbf{v}_1, \ldots, \mathbf{v}_l$ of the matrix $\mathbf{W}^{-1}\mathbf{B}$, scaled so that $\mathbf{v}_i^t \mathbf{W} \mathbf{v}_i = 1$ for $i = 1, \ldots, l$. The number $l$ of strictly positive eigenvalues of the eigen-decomposition of $\mathbf{W}^{-1}\mathbf{B}$ can be shown to be $l \leq \min(g-1, p)$. By arranging the eigenvectors $\mathbf{v}_1, \ldots, \mathbf{v}_l$ as columns in the matrix $\mathbf{V}$ we can define the *Fisher discriminant scores* for an observation $\mathbf{x}$ as

$$d_j^F(\mathbf{x}) = \Big( (\mathbf{x} - \boldsymbol{\mu}_j)^T \mathbf{V}\mathbf{V}^T (\mathbf{x} - \boldsymbol{\mu}_j) - 2\ln p_j \Big)^{\frac{1}{2}} \tag{39}$$

for $j = 1, \ldots, g$. A new observation $\mathbf{x}$ to classify is assigned to population $\pi_k$ if its Fisher discriminant score $d_k^F(\mathbf{x})$ is the smallest among the scores for all groups $d_1^F(\mathbf{x}), \ldots, d_g^F(\mathbf{x})$.

Note that the Fisher discriminant scores (39) are penalised by the term $-2\ln p_j$ with the effect that an observation is less likely to be assigned to groups with smaller prior probabilities. Using this penalty term, the Fisher discriminant rule is optimal in the sense of having a minimal total probability of misclassification for populations which are normally distributed and have equal covariance matrices. This optimality is lost if fewer eigenvectors are taken for computing the Fisher discriminant scores than the number of strictly positive eigenvalues of $\mathbf{W}^{-1}\mathbf{B}$. On the other hand, since $l \leq \min(g-1, p)$, Fisher's method allows for a reduction of the dimensionality which can be especially useful for graphical representations of the observations in the new discriminant space. In fact, this is the main advantage of the Fisher method that it is able to find a graphical representation of the data in a form that best captures the differences among the groups.

For applying Fisher's rule to real data one has to estimate the group means and covariances at the basis of the training data. Then the eigenvectors of $\hat{\mathbf{W}}^{-1}\hat{\mathbf{B}}$ to the strictly positive eigenvalues can be used to estimate the Fisher discriminant scores $\hat{d}_j^F$ for each group $j$. The classical way of estimating group means and covariances are sample mean and sample covariance. In presence of deviations from the assumption of multivariate normally distributed data of each group a robust approach is recommended. As outlined in the previous section, several proposals have been made in the literature for robustly estimating the parameters, like MCD or S estimators.

## 6.4 Discriminant analysis if $p > n$

Linear discriminant analysis is one of the most frequently used classification tools in multivariate statistics. Howbeit, direct application of a linear discriminant to sundry chemometric problems fails due to dimensionality prerequisites. There are essentially four ways in which one can proceed with a data set with dimensionality $p > n$ to which one wants to apply a linear discriminant classifier. These are subsequently highlighted in the ensuing paragraphs.

### 6.4.1 Singular value decomposition as a preprocessing step

A simple way in which the dimensionality can be reduced without loss of information, is a singular value decomposition retaining all components. This comes down to decomposing the original data matrix $\boldsymbol{X}$ as

$$\boldsymbol{X}^T = \boldsymbol{V}\boldsymbol{S}\boldsymbol{U}^T, \tag{40}$$

where $\boldsymbol{S}$ is a diagonal matrix whose diagonal elements are the $n$ singular values of $\boldsymbol{X}$, and $\boldsymbol{U}$ an $n \times n$ orthogonal matrix. Any further analysis can then be carried out on the reduced data matrix $\grave{\boldsymbol{X}} = \boldsymbol{US}$. Any estimated entity can be back-transformed to the space of the original variables.

As this operation does not cause any loss of information from the original data, it can be seen as a preprocessing step which mainly serves to make the data set tractable and to speed up calculations. As such it is used as a standard preprocessing step in several robust multivariate estimation techniques such as RAPCA,[29] ROBPCA,[32] Projection Pursuit PCA[5] (for this method see also the discussion on SVD preprocessing in[56]), RSIMPLS[30] and PRM regression.[52] The publically available implementations of these methods as part of the LIBRA[59] (includes RAPCA, ROBPCA and RSIMPLS) and TOMCAT toolboxes[8] (includes PP-PCA and PRM) do make usage of and SVD preprocessing step for the case $p > n$.

This preprocessing step reduces the dimensionality of the data to size $\grave{\boldsymbol{X}} \in \Re^{n \times n}$. For several algorithms this dimensionality is sufficient and leads to an increase in computational efficiency (especially for algorithms based on projection pursuit or iterative reweighting, this is the case). However, as it merely comprises reduction of dimensions, robust estimation still has to follow. For many robust estimators (such as most robust estimators for the covariance matrix), more stringent requirements on the data dimensionality are necessary in order to have good robustness properties. Different strategies to cope with the high dimensionality for LDA based on such covariance matrices, have to be thought of. A simple strategy to follow is the approach of doing LDA in a PCA score space.

### 6.4.2 LDA in a PCA score space

A second approach to perform LDA on undersampled data consists of the following two steps:

1. do PCA on the original data; estimate the optimal number of PCs ($k$) and retain PCA scores and loadings of this dimensionality;

2. perform LDA on the estimated scores.

It should be noted that such an order of proceeding, as is the case for the material presented in the next two sections, is no longer equivariant to linear transformations.

This strategy is frequently applied on clean data sets; however, caution has to be taken when using this approach on data containing outliers. The reason is simple: PCA can be affected by the outliers such that a dimension reduction to a score space of $k$ components might lead to a loss of relevant information as the information retained can be to a far too great extent related to the outliers.

This problem can readily be solved by using instead of classical PCA, one of the robust methods for PCA described in Section 4. In this way one is sure that the relevant information concerning the non-contaminated data points, is contained in the robust score space. However, still classical LDA cannot be applied to this robust score space as the outliers will be projected into it further away from the centre than in a classical score space. This would cause the linear discriminant to be vastly affected by the outliers, and would probably lead to

erroneous classification. The straightforward way to tackle this problem is by applying one of the robust methods for discriminant analysis described in the previous part of this Section in the robust score space. Summarising, a good order of proceeding consists of:

1. do robust PCA on the original data; estimate the optimal number of PCs ($k$) and retain robust PCA scores and loadings of this dimensionality;
2. perform robust LDA on the estimated scores.

This order of proceeding is sufficient if only one data set is available. However, if one or more independent properties are measured on the same samples, such that also an $\boldsymbol{Y}$ data matrix exists, another option can consist of doing LDA in a PLS score space.

### 6.4.3 LDA in a PLS score space

When two sets of data matrices $\boldsymbol{X}$ and $\boldsymbol{Y}$ exist, $\boldsymbol{X}$ being undersampled and a linear discriminant classifier has to be applied, the dimension reduction can be more effective using a PLS score space. PLS components describe best this part of variation in the data which is relevant to the dependent variable $\boldsymbol{Y}$. It is thus a viable strategy to modify the approach described in the previous paragraphs to starting from a PLS score space. The same remarks hold, i.e. the PLS score space has to be estimated robustly by one of the methods described in Section 5, whereafter a robust linear discriminant classifier has to be used.

### 6.4.4 D-PLS

Finally, linear classification can also be adapted to the undersampled situation using a simpler strategy: doing regression for classification purposes by using a binary predictand. The binary predictand can be coded in different ways; no agreement exists on which is best. In practice mostly a [-1/1] or a [0/1] predictand is used, meaning that if a case belongs to the first class, the corresponding value in the predictand is set to 1, whereas in the other case, it is set to -1 or 0, respectively. In order to classify new samples, a rule of thumb has to be used. Several rules have appeared in the literature, but the most frequently applied seems to be choosing the arithmetic mean value between both as the decision boundary. In practice this implies that if a [-1/1] coding is chosen, future samples whose response is negative, are considered to belong to class 2, whereas samples whose predicted response is positive, are considered to class one.

If least squares regression is used to do calibration and prediction, one obtains a method which is very closely related to linear discriminant analysis. If the data contain more variables than cases this is no longer possible, but is is of course straightforward to use one the regularised regression techniques which are appropriate for such a data structure, such as ridge regression, principal component regression or (probably the most popular one) partial least squares. If PLS is used for these purposes, it is commonly referred to as *discrimination partial least squares (D-PLS)* (see e.g.[43]). D-PLS, and some generalisations, are frequently reported to perform well in applications to various types of data, for a few recent applications see e.g.[44,51]). The main topics of discussion concerning this method are how the decision boundary should be set, and whether univariate or multivariate PLS (PLS2) should be used.

If outliers are present in the data, the D-PLS method can be robustified by using a robust alternative to PLS regression, such as RSIMPLS or PRM with the same coded predictand. Note that if multivariate PLS is preferred, only RSIMPLS can be used as no multivariate equivalent of PRM exists. Although this approach to robust classification seems simple, up to our knowledge no papers have appeared describing its properties.

# 7 Validation

## 7.1 Precision and uncertainty

### 7.1.1 How to evaluate uncertainty for robust estimators?

Robust methods of regression are often used for quantitative purposes: to actually *predict* one or several entities from data. However, if a robust method is used for prediction, then the following question will automatically be raised: "Just how precise are these predictions?" Even if a robust method is used for outliers detection this question could make sense as well. If the score space is used to detect the outliers, it is maybe worthwhile to know how big the uncertainty of the scores for the individual samples is whilst drawing any inference from them. Note that accompanying scores with uncertainties is currently not common practice – but can make sense.

In any case, if uncertainties are required, the variance of the estimator has to be known. For classical estimators the variance is well known and is generally given by well tractable closed form expressions (e.g. for least squares regression it is known that the variance of the vector of regression coefficients equals $\text{var}(\hat{\boldsymbol{\beta}}) = \hat{\sigma}^2(\boldsymbol{X}^T\boldsymbol{X})^{-1}$, with the estimated residual variance $\hat{\sigma}^2$). Already for some classical estimators an exact expression cannot be obtained. For instance, the partial least squares regression estimator is not linear in the predictand, due to which an exact equation cannot be derived. The most precise estimate of variance for the PLS regression coefficients can be constructed by retaining the first two terms of a first order Taylor series expansion from which the Jacobian matrix is computed. Note that in the case of PLS, a fast algorithm exists to estimate the variance of the vector of regression coefficients by this method.[50]

For robust estimators, the situation is in most cases even worse. Not only are the estimators frequently nonlinear in the predictand, but they are often also implicitly defined, such that it is very hard, or even impossible in some cases, to obtain exact or approximate analytical expressions for the estimators' variance. Nonetheless, one can still rely on numerical methods, the most popular of which is certainly the bootstrap. But also related methods like cross-validation or the jackknife (which is in fact an approximation to the bootstrap), are used in practice.

### 7.1.2 The bootstrap

The bootstrap is a numerical method which provides estimates of uncertainty for an estimator (denoted $T$) under consideration. Its main idea is the following. In statistics, data are virtually always assumed to consist of a sample of $n$ cases, which are $n$ observations of a $p$ variate distribution $G$. In theory, the precision of any estimates obtained from these $n$ samples could be checked easily

by repeating this process many times, for instance by drawing 5000 samples, all of which contain $n$ cases, from the distribution. Then for each of these samples, the estimator under consideration can be evaluated, such that 5000 evaluations of this estimator $T(G_n^{(i)})$ are obtained (here, the notation $G_n^{(i)}$ means the distribution which puts weight $1/n$ at each case in sample $i$). By then computing the standard deviation of these 5000 evaluations, an estimate of scale for the estimator $T$ is obtained.

Of course, in practice it is impossible to have 5000 samples of size $n$: this would imply that $5000n$ times a measurement has to be done. But what can be done, is to mimic this process of drawing samples from the distribution, only using the one sample of $n$ cases which is available. The idea of the bootstrap thus consists of the following: resample a large number of times ($m$) from the available sample, evaluate for all resamples the estimator and eventually compute the scale of these $m$ estimates. Resampling is done in practice by sampling with replacement.

A basic outline of a bootstrap algorithm is:

1. compute the required estimate $T(G_n)$ from the original data;
2. select a random number $\check{n} \leqslant n$ of samples from the data matrix $\boldsymbol{X}$ (or from the augmented data matrix $(\boldsymbol{X}, \boldsymbol{y})$ if a predictand exists) and replace them with $\check{n}$ randomly chosen samples from the same data matrix;
3. repeat the previous process $m$ times, hence constructing $m$ bootstrap data matrices $\boldsymbol{X}^{(i)}$ (and if appropriate $\boldsymbol{y}^{(i)}$), to which correspond $m$ empirical distributions $G_n^{(i)}$;
4. compute $m$ boostrap estimates of $a_i = T(G_n^{(i)})$ from these bootstrap data matrices;
5. compute a measure of scale $S(\{a_i\})$ from these $m$ estimates; henceforth it is assumed that $S(\{a_i\}) = \sqrt{\widehat{\mathrm{var}}(T(G))}$.

The basic algorithm can be modified in several ways, such as a correction for bias and an acceleration. For more details we refer the interested reader to a monograph on the bootstrap,[17] but we note that the basic algorithm presented here already performs well in most situations.

Some questions need to be answered concerning the bootstrap method. At first, which scale estimator should be used? If asymptotic normality of the estimator is assumed, it is probably most appropriate to use the classical standard deviation. In other cases, if it is not sure which distribution can be expected, another approach can be followed, by simply taking the $100\frac{1-\alpha}{2}$th and $100\frac{\alpha}{2}$th percentiles as confidence bands for the estimator in question, the explicit computation of a scale estimator thus being by-passed.

Secondly, how many bootstrap resamples should be constructed? The quality of the bootstrap is evidently proportional to the number of bootstrap samples that are being constructed. It is considered good practice to use $m = 2000$ bootstrap samples if one envisages the construction of confidence intervals.[17]

The bootstrap has been successfully applied to virtually all branches of statistics, and is gaining acceptancy in the applied fields as well. For typical chemometric tools the bootstrap has been evaluated to provide good estimates of uncertainty, for PLS see,[14] for PARAFAC see[34] and for tri-PLS see.[54]

### 7.1.3 The robust bootstrap

The bootstrap is a straightforward method for (approximately) obtaining uncertainties of an estimate. However, one of its basic assumptions is that all cases in the data are taken from a single underlying distribution $G$. When outliers are present in the data, these outlying cases may be assumed not to have been generated by that distribution, such that the basic principle of the bootstrap is not respected. Hence it follows that the bootstrap estimator for the uncertainty, may break down as well for contaminated data sets.[48]

The reason for breakdown of the bootstrap is the following: one replaces an arbitrary number of cases $\check{n}$ from the original $n$ cases by some randomly chosen set of $\check{n}$ cases from the original data. This new set of $\check{n}$ cases may accidentally be identical to the original set, such that in fact the bootstrap sample is the original data matrix $\boldsymbol{X}$. But it may also happen that, by coincidence, the randomly chosen set of $\check{n}$ cases only consists of outliers. Now assume that the estimator $T$ can resist a fraction $\nu = \tilde{n}/n$ of outliers, where $\tilde{n}$ denotes the number of outliers, and that $\frac{\check{n}}{n} > \nu$. This means that $T$ will break down for the bootstrap sample coincidentally containing $\check{n}$ outliers, whereas it does not break down for the original data. This violates the basic principles of the bootstrap, as for doing the bootstrap one tries to mimic $m$ times the way in which the original set of samples has been drawn. As in the latter only $\ell$ outliers were present, so should this be the case for the bootstrap samples, otherwise the latter are drawn in a different way from the population.

The effect of coincidental concentration of outliers in the bootstrap is illustrated for the data set described in Section 5.4.2. Five hundred bootstrap data matrices were constructed with the basic algorithm described above, using the implementation from the MATLAB Statistics Toolbox (The MathWorks, Natick, MA, USA). As has been heeded in Section 5.4.2, the PRM regression estimator can resist well to the leverage points in the data and thus does not break down. For each of these bootstrap data matrices, the vector of regression coefficients was estimated by PLS as well as by PRM, by virtue of which vectors the sodium oxide concentration was predicted. This means that for each of these predicted concentrations, a set of 500 bootstrap predicted concentrations was available, allowing us to have an idea of the distribution of these predicted concentrations. Although no exact theoretical results are available as regards the distribution of predicted responses by PLS nor by PRM, it seems viable to expect that the distribution be unimodal and symmetric.

In Figure 12, the results are shown for the predictions of two cases from the independent validation set. These are cases 1 and 50 from this set, and noteworthily these are glasses of the potasso-calcic and sodic type, respectively. At first it can be observed that PLS suffers a lot from the presence of the leverage points in the calibration set: the predicted responses vary over a wide range in concentrations and, especially for case 1, appears to be trimodal. For PRM, the distributions are more narrow, but nevertheless, some predictions seem to fall into bins far away from the centre (on the right hand side for case 1 and on the left hand side of the centre for case 50). This is exactly the effect of up-concentration of the outliers: some bootstrap data matrices did contain in practice far more leverage points than the original calibration set, such that even PRM yields erroneous predictions for these few bootstrap samples. It is thus clear that the bootstrap needs a slight modification in order to be applicable to

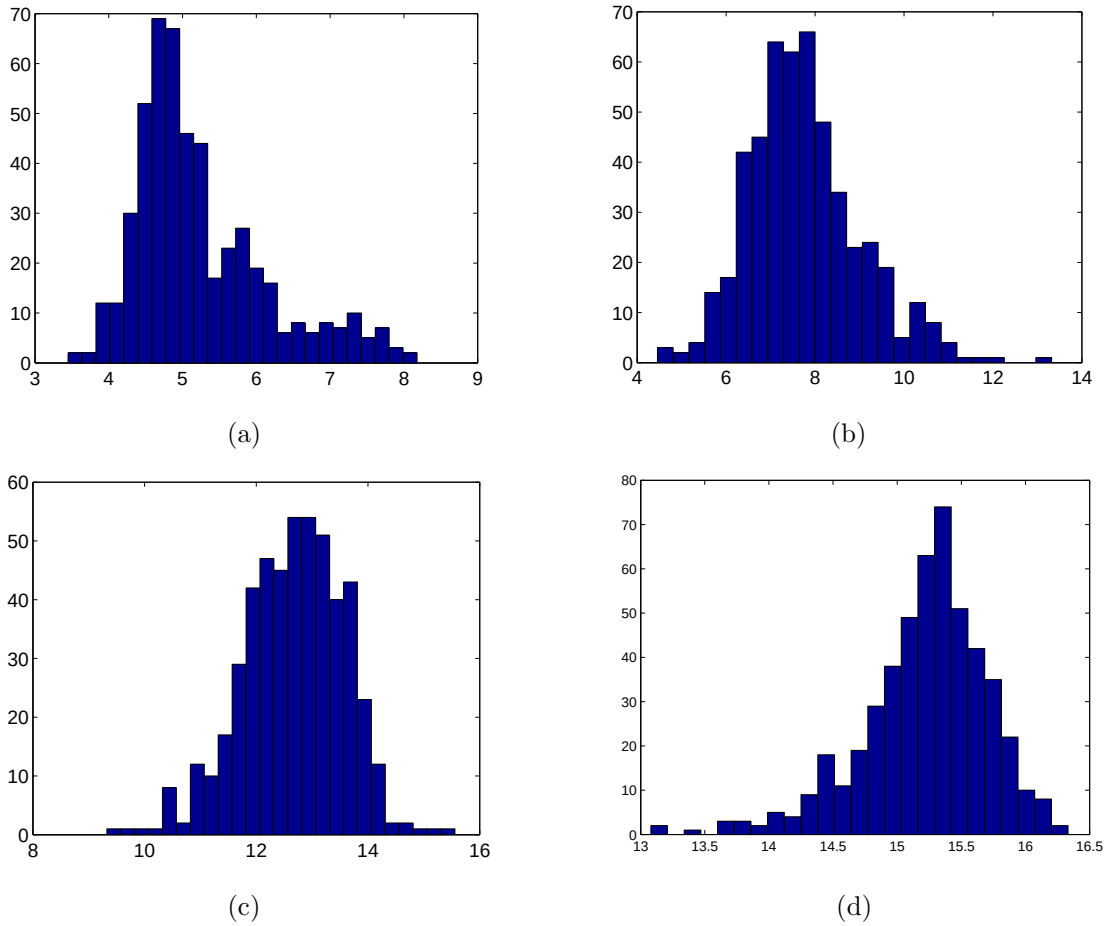



Figure 12: Bootstrap distributions of the predicted $Na_2O$ concentrations by PLS (left column) and PRM (right column), for sample 1 (top row) and sample 50 (bottom row).

data containing outliers.

The bootstrap can be modified in two ways: either controlling the number of outliers in the bootstrap sample or limiting the influence on the outcome by using a robust estimator of scale $S$ (in step 5 of the basic algorithm, Section 7.1.2). The first option is more cumbersome: it implies outlier identification and is hard to apply if many borderline cases are present (can they be samples in the bootstrap or not?). Hence, the second option is most often recommended: use a robust scale estimator on the empirical distribution of the estimator (such as the distributions shown in Figure 12, right column). It is best to use a robust scale estimator which is consistent with the robust estimator which one wants to bootstrap, e.g. if one uses RCR with a 20% trimmed scale to estimate the concentrations, it is preferable to use a 20% trimmed scale here as well.

### 7.1.4 Computational efficiency

Robust estimators are nearly always harder to compute than classical estimators. They frequently involve complex iterative schemes; even some of the most simple robust estimators (in the computational sense), require far more floating point operations than their classical counterparts. For instance, many M-estimators can be computed by an iterative reweighting algorithm. If the classical estimator has on average to be iterated $\kappa$ times, then the overall computation time for the M-estimator is slightly more than $\kappa$ times the computation time of the classical estimator.

As robust estimators tend to be rather slow in terms of computation, bootstrapping them may be a very slow procedure: the robust estimator may have to be computed 500 or, even worse, 2000 times. Thus, straightforward implementation of the basic algorithm given in Section 7.1.2 is only feasible for the fastest robust estimators. For the most common chemometric tools, this implies the methods based on the spatial sign (spatial sign based PCA[37] and PLS[55]), iterative reweighting (PRM[52]) or the methods based on ROBPCA (ROBPCA[32] and RSIMPLS[30]).

For some of the regression estimators described in Section 3, a specially designed fast bootstrap procedure exists. These *fast and robust bootstrap* procedures basically consist of the following: the estimating equations of the estimator are bootstrapped, such that not in every loop the whole estimator has to be constructed. Fast and robust bootstrap exists (up to our knowledge) for S-estimators,[62] and least trimmed squares.[61]

## 7.2 Model selection

The last important issue concerning robust estimation, is model selection. For the latent variables based robust estimators, the question is: how many latent variables are optimally used to model the data at hand? The most popular technique to obtain an answer to this question is cross-validation. Cross-validation exists in various flavours; in the context of PLS recent studies favour a so-called *full or Monte Carlo* cross-validation, where for a large number of iterations, the data are randomly split up into a model and a test set (now named the CV model and CV test sets, respectively). For each of these iterations, calibration is done based on the CV model set for all possible complexities and the responses are predicted for the CV test with using these models of different complexities.

From these predicted responses a root mean-squared error of cross-validation is computed. The complexity leading to the minimum in RMSECV is considered to be optimal. For a thorough simulation study concerning different types of cross-validation, see.[1,2]

When outliers are present in the data, the same problems may also affect cross-validation. At first, it is possible that due to random selection of the cases which belong to the CV model and test sets, the CV model set will contain a higher fraction of outliers than the original data set (cf. the bootstrap, Section 7.1.3). Hence, it is possible that the robust estimator breaks down for the CV model set whereas it would not for the original data set, leading to misleading root mean squared error of cross validation curves. Secondly, it is also probable that the CV test set contains outliers. The robust estimator is supposed to fit the clean data well and to give good predictions for regular cases. However, the robust estimator should not fit the outliers well, and should thus also not provide good predictions for outlying cases. Hence, the outliers which are possibly part of the CV test set, will be predicted badly, giving rise to an artificially high root mean squared error of cross validation.

In order to tackle these problems, two solutions can be presented, analogous to the bootstrap (see Section 7.1.3). On the one hand, one can do an outlier detection prior to cross-validation, exclude these outliers from the CV test sets and keep their fraction in the CV model sets quasi constant. Howbeit, this procedure give rise to some questions, e.g. if you are doing robust PCA in order to detect better the outliers, and you do cross validation to know the complexity of this model, in fact the outliers have not yet been detected such that they cannot be monitored in cross validation. Thus also for cross validation, the more straightforward strategy is to compute a robust variant of the root mean squared error of cross validation. This is simply done by selecting a percentage of trimming corresponding to the fraction of outliers that is likely to arise. This modus operandi has been implemented in the TOMCAT toolbox.[8]

Finally, in the computational sense it holds again that cross validation is only applicable to the fastest robust methods (cf. the bootstrap, Section 7.1.4). A specifically designed fast and robust cross validation method has, up to our knowledge, only been tailored for the ROBPCA method.[18]

## Suggestions for further reading

In contrast to the research areas covered in several other chapters of this reference work, construction and application of robust methods for chemometrics is still very much a research topic, such that there does not, up to our knowledge, exist a textbook which solely addresses this subject. We thus refer the interested reader to the several textbooks on chemometric multivariate methods, to be found in the suggestions for further reading provided in several of the chapters preceding the current one. In addition to these works, we advise to consult recent textbooks on robust statistics. A particularly nice and thorough introduction to robust statistics can be found in: *Robust Statistics: Theory and Methods* by Ricardo A. Maronna, Douglas R. Martin, Victor J. Yohai, Wiley, 2006. A somewhat older, nicely written textbook is *Robust Estimation and Testing* by Robert G. Staudte, Simon J. Sheather, Wiley, 1990. An old textbook which is specific to robust regression is: *Robust Regression and Outlier Detec-*

*tion* by Peter J. Rousseeuw and Annick M. Leroy, Wiley, 1987. For material on specific robust methods for chemometrics, as no textbook on this subject exists, the reader is referred to articles, i.e. the corresponding references in the list given below.

## The authors

Peter Filzmoser (born 1968) studied Applied Mathematics at the Vienna University of Technology, Austria, where he also wrote his doctoral thesis and habilitation devoted to the field of multivariate statistics. His research was attracted by robust statistics, resulting in many international collaborations and various scientific papers in this area. His interest in applications of robust methods resulted in the development of R software packages. He was and is involved in the organisation of several scientific events devoted to robust statistics. Since 2001 he is dozent at the Statistics Department at Vienna University of Technology. He was visiting professor at the Universities of Vienna and Toulouse I.

Sven Serneels (born 1980) graduated at the department of Chemistry at the University of Antwerp, where he consecutively wrote his PhD thesis, entirely devoted to chemometrics. Among others, he made research visits to the Institute of Statistics and Probability Theory of the Technical University of Vienna and to the Department of Chemometrics of the Silesian University. Among other topics, both during his PhD thesis and afterwards, his work comprises contributions to robust methods in chemometrics. He is currently a Research Chemometrician in the Statistics and Chemometrics group of Shell Global Solutions.

Ricardo Maronna was born in 1940 in Bahía Blanca, Argentina. He obtained his M.S. in Mathematics at National Southern University in 1961 and his PhD in Mathematics at the University of Buenos Aires in 1975. He is currently professor at the Faculty of Exact Sciences, University of La Plata, Argentina, principal researcher at C.I.C.P.B.A., and visiting professor at the Faculty of Exact Sciences, University of Buenos Aires. He has been visiting professor at ETH Zürich, Carlos III University (Madrid), and IMPA (Rio de Janeiro, Brazil), among others; and visiting researcher at the Universities of Washington and British Columbia. He has also been consultant for government and private organisations. His main research area is Robust Statistics. He has authored or co-authored numerous research articles on statistical methods and their applications to science and technology, and two books.

Pierre Van Espen (born 1952) was one of the first PhD students at the University of Antwerp. He is currently a professor at the Department of Chemistry at the University of Antwerp and *Profesor Invitado* at the University of Havana, Cuba. He accomplished some pioneering work on the analysis of X-ray spectra by means of the development of software for the PDP 11/45 computer. On a later stage, this resulted in the AXIL (Analysis of X-Ray spectra by Iterative Least Squares regression) package, which gained international acceptance and widespread use through the International Atomic Energy Agency (IAEA). Apart from the AXIL package, his work also comprises application and extension of chemometric methods for the analysis of data from X-ray spectrometry.

## References


[1] Baumann, K.; Albert, H.; von Korff, M. A systematic evaluation of the benefits and hazards of variable selection in latent variable regression. Part I. Search algorithm, theory and simulations. *J. Chemometr.* **2002**, *16*, 339-350.

[2] Baumann, K.; von Korff, M.; Albert, H. A systematic evaluation of the benefits and hazards of variable selection in latent variable regression. Part II. Practical applications. *J. Chemometr.* **2002**, *16*, 351-360.

[3] Croux, C.; Haesbroeck, G. Principal component analysis based on robust estimators of the covariance or correlation matrix: Influence functions and efficiencies. *Biometrika* **2000**, *87*, 603-618.

[4] Croux, C.; Ruiz-Gazen, A. A fast algorithm for robust principal components based on projection pursuit. In: *COMPSTAT: Proceedings in computational statistics*; Prat, A., Ed.; Physica-Verlag, Heidelberg, 1996, pp. 211-216.

[5] Croux, C.; Ruiz-Gazen, A. High breakdown estimators for principal components: the projection-pursuit approach revisited. *J. Multivariate Anal.* **2005**, *95*, 206-226.

[6] Croux, C.; Filzmoser, P.; Oliveira, M.R. Algorithms for projection-pursuit robust principal component analysis. *Chemometr. Intell. Lab. Syst.* **2007**, *87*, 218-225.

[7] Cummins D.J.; Andrews C. Iteratively reweighted partial least squares: A performance analysis by Monte Carlo simulation. *J. Chemometr.* **1995**, *9*, 489-507.

[8] Daszykowski, M.; Serneels, S.; Kaczmarek, K.; Van Espen, P.J.; Croux, C.; Walczak, B. TOMCAT: A MATLAB toolbox for multivariate calibration techniques. *Chemometr. Intell. Lab. Syst.* **2007**, *85*, 269-277.

[9] Davies, P.L. Asymptotic behavior of S-estimators of multivariate location parameters and dispersion matrices. *Ann. Statist.* **1987**, *15*, 1269-1292.

[10] Davies, L.P.; Gather, U. Breakdown and groups. *Ann. Statist.* **2005**, *33*, 977-1035.

[11] Debruyne, M.; Hubert, M. The influence function of Stahel-Donoho type methods for robust covariance estimation and PCA. *Scand. J. Statist.*, submitted for publication.

[12] de Jong, S. SIMPLS: an alternative approach to partial least squares regression. *Chemometr. Intell. Lab. Syst.* **1993**, *42*, 251-263.

[13] Dempster, A.P.; Laird, N.M.; Rubin, D.B. Iteratively reweighted least squares for linear regression when the errors are normal/independent distributed. In: *Multivariate analysis V*, Krishnaiah, P.R., Ed., North Holland, 1980, pp. 35-57.

[14] Denham, M.C. Prediction intervals in partial least squares. *J. Chemometr.* **1997**, *11*, 39-52.

[15] Dodge, Y.; Kondylis, A.; Whittaker, J. Extending PLS1 to PLAD regression and the use of the L1 norm in soft modelling. In: *Compstat 2004: Proceedings in Computational Statistics*, Antoch, J., Ed., pp. 935-942, Springer-Verlag, Heidelberg, 2004.

[16] Donoho, D.L. Breakdown properties of multivariate location estimators. Ph.D. Qualifying paper, Harvard University, 1982.

[17] Efron, B.; Tibshirani, R.J. *An Introduction to the Bootstrap*, Chapman and Hall, New York, 1993.

[18] Engelen, S.; Hubert, M. Fast model selection for robust calibration methods. *Anal. Chim. Acta* **2005**, *544*, 219-228.

[19] Filzmoser, P.; Serneels, S.; Croux, C.; Van Espen, P.J. Robust multivariate methods: The projection pursuit approach. In *From Data and Information Analysis to Knowledge Engineering*, Spiliopoulou, M., Kruse, R., Borgelt, C., Nürnberger, A.,Gaul, W., Eds., Springer Verlag, Berlin, 2006, pp. 270-277.

[20] Fisher, R.A. The statistical utilization of multiple measurements. *Ann. Eugen.* **1938**, *8*, 376-386.

[21] Friedman, J.H.; Tukey, J.W. A projection pursuit algorithm for exploratory data analysis, *IEEE Transactions on Computers* **1974**, *c-23 (9)*, 881-890.

[22] Frosch-Møller, S.; von Frese, J.; Bro, R. Robust methods for multivariate data analysis. *J. Chemometr.* **2005**, *19*, 549-563.

[23] Gil, J.A.; Romera, R. On robust partial least squares (PLS) methods. *J. Chemometr.* **1998**, *12*, 365-378.

[24] Hampel, F.R.; Ronchetti, E.M.; Rousseeuw, P.J.; Stahel, W.A. *Robust Statistics: The Approach Based on Influence Functions*. Wiley, New York, 1986.

[25] Hawkins, D.M.; McLachlan, G.J. High-breakdown linear discriminant analysis. *J. Am. Statist. Assoc.* **1997**, *92*, 136-143.

[26] He, X.; Fung, W.K. High breakdown estimation for multiple populations with applications to discriminant analysis. *J. Multivariate Anal.* **2000**, *72*, 151-162.

[27] Huber, P.J. *Robust statistics*. Wiley, New York, 1981.

[28] Huber, P. Projection pursuit. *Ann. Statist.* **1985**, *13*, 435-475.

[29] Hubert, M.; Rousseeuw, P.J.; Verboven, S. A fast method for robust principal components with applications to chemometrics. *Chemometr. Intell. Lab. Syst.* **2002**, *60*, 101-111.

[30] Hubert, M.; Vanden Branden, K. Robust methods for partial least squares regression. *J. Chemometr.* **2003**, *17*, 537-549.

[31] Hubert, M.; Van Driessen, K. Fast and robust discriminant analysis. *Computat. Statist. Data Anal.* **2004**, *45*, 301-320.

[32] Hubert, M., Rousseeuw, P.J., Vanden Branden, K. ROBPCA: a new approach to robust principal components analysis. *Technometr.* **2005**, *47*, 64-79.

[33] Janssens, K.H.A.; De Raedt, I.; Schalm, O.; Veeckman, J. Composition of $15^{\text{th}}$-$17^{\text{th}}$ century archaeological glass vessels excavated in Antwerp. *Mikrochim. Acta* **1998**, *15 (Suppl.)*, 253-267.

[34] Kiers, H.A.L. Bootstrap confidence intervals for three-way methods. *J. Chemometr.* **2004**, *18*, 22-36.

[35] Lemberge, P.; De Raedt, I.; Janssens, K.H.; Wei, F.Z.; Van Espen, P.J. Quantitative analysis of 16-17th century archaeological glass vessels using PLS regression of EPXMA and $\mu$-XRF data. *J. Chemometr.* **2000**, *14*, 751-763.

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

[37] Locantore, N.; Marron, J.S.; Simpson, D.G.; Tripoli, N.; Zhang, J.T.; Cohen, K.L. Principal component analysis for functional data. *Test* **1999**, *8*, 1-73.

[38] Lopuhaä, H.P. On the relation between S-estimators and M-estimators of multivariate location and covariance. *Ann. Statist.* **1989**, *17*, 1662-1683.

[39] Lopuhaä, H.P. Highly efficient estimators of multivariate location with high breakdown point. *Ann. Statist.* **1992**, *20*, 398-413.

[40] Maronna, R.A. Principal components and orthogonal regression based on robust scales. *Technometr.* **2005**, *47*, 264-273.

[41] Maronna, R.A.; Martin, R.D.; Yohai, V.J. *Robust statistics. Theory and Methods.* Wiley, New York, 2006.

[42] Maronna R.; Yohai V. The behavior of the Stahel-Donoho robust multivariate estimator. *J. Am. Statist. Assoc.* **1995**, *90*, 330-341.

[43] Naes T.; Isaksson T.; Fearn T.; Davies A. *A User-friendly Guide to Multivariate Calibration and Classification.* NIR Publications, Chichester (UK), 2002.

[44] Pérez-Marín, D.C.; Garrido-Varo, A.; Guerrero, J.E. Optimization of discriminant partial least squares regression models for the detection of animal by-product meals in compound feedingstuffs by near infrared spectroscopy. *Appl. Spectrosc.* **2006**, *60*, 1432-1437.

[45] Rao, C.R. The utilization of multiple measurements in problems of biological classification. *J. R. Statist. Soc.* **1948**, *Series B, 10*, 159-203.

[46] Rousseeuw, P.J.; Van Driessen, K. Computing LTS regression for large data sets. *Data Mining Knowl. Disc.* **2006**, *12*, 29-45.

[47] Rousseeuw, P.J., Van Driessen, K. A fast algorithm for the Minimum Covariance Determinant estimator. *Technometr.* **1999**, *41*, 212-223.

[48] Salibian-Barrera, M.; Zamar, R.H. Bootstrapping robust estimates of regression. *Ann. Statist.* **2002**, *30*, 556-582.

[49] Serneels, S.; Croux, C.; Van Espen, P.J. Influence properties of partial least squares regression. *Chemometr. Intell. Lab. Syst.* **2004**, *71*, 13-20.

[50] Serneels, S.; Lemberge, P.; Van Espen, P.J. Calculation of PLS prediction intervals using efficient recursive relations for the Jacobian matrix. *J. Chemometr.* **2004**, *18*, 76-80.

[51] Serneels, S.; Moens, M.; Van Espen, P.J.; Blockhuys, F. Identification of micro-organisms by dint of the electronic nose and trilinear partial least squares regression. *Anal. Chim. Acta* **2004**, *516*, 1-5.

[52] Serneels, S.; Croux, C.; Filzmoser, P.; Van Espen, P.J. Partial robust M regression. *Chemometr. Intell. Lab. Syst.* **2005**, *79*, 55-64.

[53] Serneels, S.; Filzmoser, P.; Croux, C.; Van Espen, P.J. Robust continuum regression. *Chemometr. Intell. Lab. Syst.* **2005**, *76*, 197-204.

[54] Serneels, S.; Van Espen, P.J. Bootstrap confidence intervals for trilinear partial least squares regression. *Anal. Chim. Acta* **2005**, *544*, 153-158.

[55] Serneels, S.; De Nolf, E.; Van Espen, P.J. Spatial sign pre-processing: a simple way to impart moderate robustness to multivariate estimators. *J. Chem. Info. Model.* **2006**, *46*, 1402-1409.

[56] Stanimirova, I.; Walczak, B.; Massart, D.L.; Simeonov, V. A Comparison between two robust PCA algorithms. *Chemometr. Intell. Lab. Syst.* **2004**, *71*, 83-95.

[57] Stahel, W.A. *Breakdown of covariance estimators.* Research Report 31, Fachgruppe für Statistik, ETH-Zürich, Zürich, 1981.

[58] Tenenhaus, M.; Esposito Vinzi, V.; Chatelin, Y.-M.; Lauro, C. PLS path modeling. *Computat. Statist. Data Anal.* **2005**, *48*, 159-205.

[59] Verboven, S.; Hubert, M. LIBRA: a MATLAB library for robust analysis. *Chemometr. Intell. Lab. Syst.* **2005**, *75*, 127-136.

[60] Visuri, S.; Koivunen, V.; Oja, H. Sign and rank covariance matrices. *J. Statist. Plan. Infer.* **2000**, *91*, 557-575.

[61] Willems, G.; Van Aelst, S. Fast and robust bootstrap for LTS. *Computat. Statist. Data Anal.* **2005**, *48*, 703-715.

[62] Willems, G.; Van Aelst, S. Multivariate regression S estimators for robust estimation and inference. *Statist. Sinica* **2005**, *15*, 981-2001.

[63] Wold, H. Soft modeling by latent variables: the non-linear iterative partial least squares approach. In: *Perspectives in probability and statistics, papers in honor of M.S. Bartlett*, Gani, J., Ed., Academic Press, London, 1975, pp. 117-142.