ROBPCA: a New Approach to Robust Principal Component Analysis

Mia Hubert*, Peter J. Rousseeuw† and Karlien Vanden Branden‡

Second revision: October 27, 2003

Abstract

In this paper we introduce a new method for robust principal component analysis. Classical PCA is based on the empirical covariance matrix of the data and hence it is highly sensitive to outlying observations. In the past, two robust approaches have been developed. The first is based on the eigenvectors of a robust scatter matrix such as the MCD or an S-estimator, and is limited to relatively low-dimensional data. The second approach is based on projection pursuit and can handle high-dimensional data. Here, we propose the ROBPCA approach which combines projection pursuit ideas with robust scatter matrix estimation. It yields more accurate estimates at non-contaminated data sets and more robust estimates at contaminated data. ROBPCA can be computed fast, and is able to detect exact fit situations. As a byproduct, ROBPCA produces a diagnostic plot which displays and classifies the outliers. The algorithm is applied to several data sets from chemometrics and engineering.

Key words: Principal Component Analysis, Robust methods, High-dimensional data, Projection Pursuit

*Assistant Professor, Department of Mathematics, Katholieke Universiteit Leuven, W. De Croylaan 54, B-3001 Leuven, Belgium, mia.hubert@wis.kuleuven.ac.be

†Professor, Department of Mathematics and Computer Science, University of Antwerp, Middelheimlaan 1, B-2020 Antwerpen , Belgium, peter.rousseeuw@ua.ac.be.

‡Assistant, Department of Mathematics, Katholieke Universiteit Leuven, W. De Croylaan 54, B-3001 Leuven, Belgium, karlien.vandenbranden@wis.kuleuven.ac.be

1 Introduction

Principal component analysis is a popular statistical method which tries to explain the covariance structure of data by means of a small number of components. These components are linear combinations of the original variables, and often allow for an interpretation and a better understanding of the different sources of variation. Because PCA is concerned with data reduction, it is widely used for the analysis of high-dimensional data which are frequently encountered in chemometrics, computer vision, engineering, genetics, and other domains. PCA is then often the first step of the data analysis, followed by discriminant analysis, cluster analysis, or other multivariate techniques. It is thus important to find those principal components that contain most of the information.

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

The goal of robust PCA methods is to obtain principal components that are not influenced much by outliers. A first group of methods is obtained by replacing the classical covariance matrix by a robust covariance estimator. Maronna (1976) and Campbell (1980) proposed to use affine equivariant M-estimators of scatter for this purpose, but these cannot resist many outliers. More recently, Croux and Haesbroeck (2000) used positive-breakdown estimators such as the minimum covariance determinant (MCD) method (Rousseeuw 1984) and S-estimators (Davies 1987, Rousseeuw and Leroy 1987). The result is more robust, but unfortunately limited to small to moderate dimensions. To see why, let us e.g. consider the MCD estimator. It is defined as the mean and the covariance matrix of the h observations (out of the whole data set of size n) whose covariance matrix has the smallest determinant. If p denotes the number of variables in our data set, the MCD estimator can only be computed if $p < h$, otherwise the covariance matrix of any h-subset has zero determinant. By default

h is about $0.75n$, and it may be chosen as small as $0.5n$; in any case p may never be larger than n. A second problem is the computation of these robust estimators in high dimensions. Today's fastest algorithms (Woodruff and Rocke 1994, Rousseeuw and Van Driessen 1999) can handle up to about 100 dimensions, whereas there are fields like chemometrics which need to analyze data with dimensions in the thousands.

A second approach to robust PCA uses Projection Pursuit (PP) techniques (Li and Chen 1985, Croux and Ruiz-Gazen 2000, Hubert et al. 2002). These methods maximize a robust measure of spread to obtain consecutive directions on which the data points are projected. This idea has also been generalized to common principal components (Boente et al. 2002). It yields transparent algorithms that can be applied to data sets with many variables and/or many observations.

In Section 2 we propose the ROBPCA method, which attempts to combine the advantages of both approaches. We also describe an accompanying diagnostic plot which can be used to detect and classify possible outliers. Several real data sets from chemometrics and engineering are analyzed in Section 3. Section 4 investigates the performance and robustness of ROBPCA through simulations. The concluding section outlines potential applications of ROBPCA in other types of multivariate data analysis.

2 The ROBPCA method

2.1 Description

The proposed ROBPCA method combines ideas of both projection pursuit and robust covariance estimation. The PP part is used for the initial dimension reduction. Some ideas based on the MCD estimator are then applied to this lower-dimensional data space. The combined approach yields more accurate estimates than the raw PP algorithm, as we will see in Section 4.

The complete description of the ROBPCA method is quite involved and relegated to the Appendix, but here is a rough sketch of how it works. We assume that the original data are stored in an $n \times p$ data matrix $X = X_{n,p}$ where n stands for the number of objects and p for the original number of variables. The ROBPCA method then proceeds in three major steps. First, the data are preprocessed such that the transformed data are lying in a subspace whose dimension is at most $n - 1$. Next, we construct a preliminary covariance

matrix S_0 that is used for selecting the number of components k that will be retained in the sequel, yielding a k-dimensional subspace that fits the data well. Then we project the data points on this subspace where we robustly estimate their location and their scatter matrix, of which we compute its k non-zero eigenvalues $l_1, \ldots, l_k$. The corresponding eigenvectors are the k robust principal components.

In the original space of dimension p, these k components span a k-dimensional subspace. Formally, writing the (column) eigenvectors next to each other yields the $p \times k$ matrix $P_{p,k}$ with orthogonal columns. The location estimate is denoted as the p-variate column vector $\hat{\boldsymbol{\mu}}$ and called the robust center. The scores are the entries of the $n \times k$ matrix

$$T_{n,k} = (X_{n,p} - \mathbf{1}_n \hat{\boldsymbol{\mu}}') P_{p,k} \tag{1}$$

where $\mathbf{1}_n$ is the column vector with all n components equal to 1. Moreover, the k robust principal components generate a $p \times p$ robust scatter matrix S of rank k given by

$$S = P_{p,k} L_{k,k} P'_{p,k} \tag{2}$$

where $L_{k,k}$ is the diagonal matrix with the eigenvalues $l_1, \ldots, l_k$.

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

$$T(XA' + \mathbf{1}_n \mathbf{v}') = (XA' + \mathbf{1}_n \mathbf{v}' - \mathbf{1}_n (A\hat{\boldsymbol{\mu}}_x + \mathbf{v})') AP = (X - \mathbf{1}_n \hat{\boldsymbol{\mu}}'_x) P = T(X).$$

Although these properties seem very natural for a PCA method, they are not shared by some other robust PCA estimators such as the resampling by half-means and the smallest half-volume methods of Egan and Morgan (1998).

2.2 Diagnostic plot

As it is the case for many robust methods, the goal of a robust PCA is twofold. First, it allows to find those linear combinations of the original variables that contain most of the

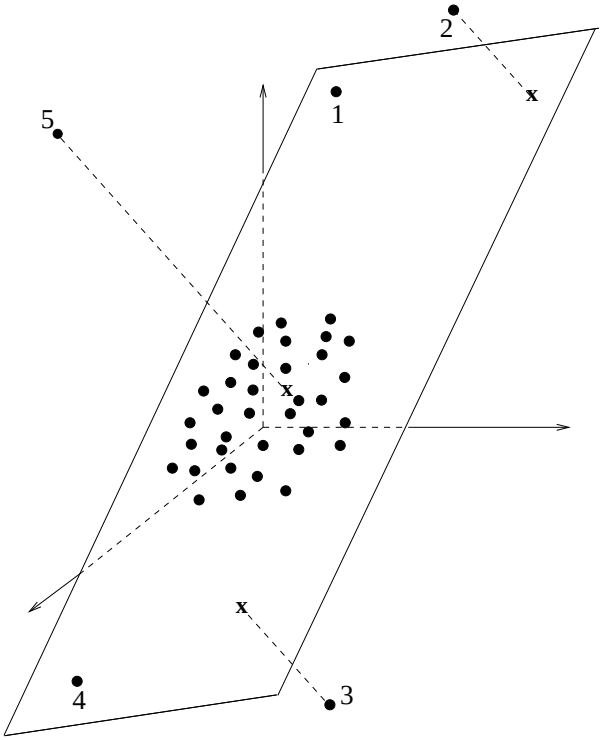


Figure 1: Different types of outliers when a three-dimensional data set is projected on a robust two-dimensional PCA-subspace.

information, even if there are outliers. Secondly, these estimates are useful to flag outliers and to determine of which type they are.

To see that there can be different types of outliers, consider Figure 1 where $p = 3$ and $k = 2$. Here we can distinguish between four types of observations. The *regular observations* form one homogeneous group that is close to the PCA subspace. Next, we have *good leverage points* that lie close to the PCA space but far from the regular observations, such as the observations 1 and 4 in Figure 1. We can also have *orthogonal outliers* whose orthogonal distance to the PCA space is large but which we cannot see when we only look at their projection on the PCA space, like observation 5. The fourth type of data points are the *bad leverage points* that have a large orthogonal distance and whose projection on the PCA subspace is remote from the typical projections, such as observations 2 and 3.

To distinguish between regular observations and the three types of outliers for higher-dimensional data, we construct a *diagnostic plot*. On the horizontal axis it plots the *robust*

score distance SD_i of each observation, given by

$$\mathrm{SD}_i = \sqrt{\sum_{j=1}^{k} \frac{t_{ij}^2}{l_j}} \tag{3}$$

where the scores t_{ij} are obtained from (1). If $k = 1$, we prefer to plot the (signed) standardized score $t_i/\sqrt{l_1}$. On the vertical axis of the diagnostic plot we display the *orthogonal distance* OD_i of each observation to the PCA subspace, defined as

$$\mathrm{OD}_i = \|\mathbf{x}_i - \hat{\boldsymbol{\mu}} - P_{p,k}\mathbf{t}_i'\| \tag{4}$$

where the ith observation is denoted as the p-variate column vector $\mathbf{x}_i$ and $\mathbf{t}_i'$ is the ith row of $T_{n,k}$.

To classify the observations we draw two cut-off lines. The cut-off value on the horizontal axis is $\sqrt{\chi^2_{k,0.975}}$ when $k > 1$ and $\pm\sqrt{\chi^2_{1,0.975}}$ when $k = 1$ (because the squared Mahalanobis distances of normally distributed scores are approximately χ^2_k-distributed). The cut-off value on the vertical axis is more difficult to determine because the distribution of the orthogonal distances is not known exactly. However, a scaled chi-squared distribution $g_1\chi^2_{g_2}$ gives a good approximation for the unknown distribution of the squared orthogonal distances (Box 1954). In Nomikos and MacGregor (1995) the method of moments is used to estimate the two unknown parameters g_1 and g_2. We prefer to follow a robust approach. We use the Wilson-Hilferty approximation for a chi-squared distribution. This implies that the orthogonal distances to the power 2/3 are approximately normally distributed with mean $\mu = (g_1 g_2)^{1/3}(1 - \frac{2}{9g_2})$ and variance $\sigma^2 = \frac{2g_1^{2/3}}{9g_2^{1/3}}$. We obtain estimates $\hat{\mu}$ and $\hat{\sigma}^2$ by means of the univariate MCD. The cut-off value on the vertical axis then equals $(\hat{\mu} + \hat{\sigma} z_{0.975})^{3/2}$ with $z_{0.975} = \Phi^{-1}(0.975)$ the 97.5% quantile of the Gaussian distribution.

Note the analogy of this diagnostic plot with that of Rousseeuw and Van Zomeren (1990) for robust regression. There the vertical axis gives the standardized residuals obtained with a robust regression method, with cut-off values at -2.5 and 2.5 (because for normally distributed data roughly 2.5% of the standardized residuals fall outside that interval).

3 Examples

We will illustrate the ROBPCA method and the diagnostic plot on several real data sets. We also compare the results from ROBPCA with four other PCA methods: classical PCA

(CPCA), RAPCA (Hubert et al. 2002), and spherical (SPHER) and ellipsoidal (ELL) PCA (Locantore et al. 1999). The latter three methods are also robust and designed for high-dimensional data.

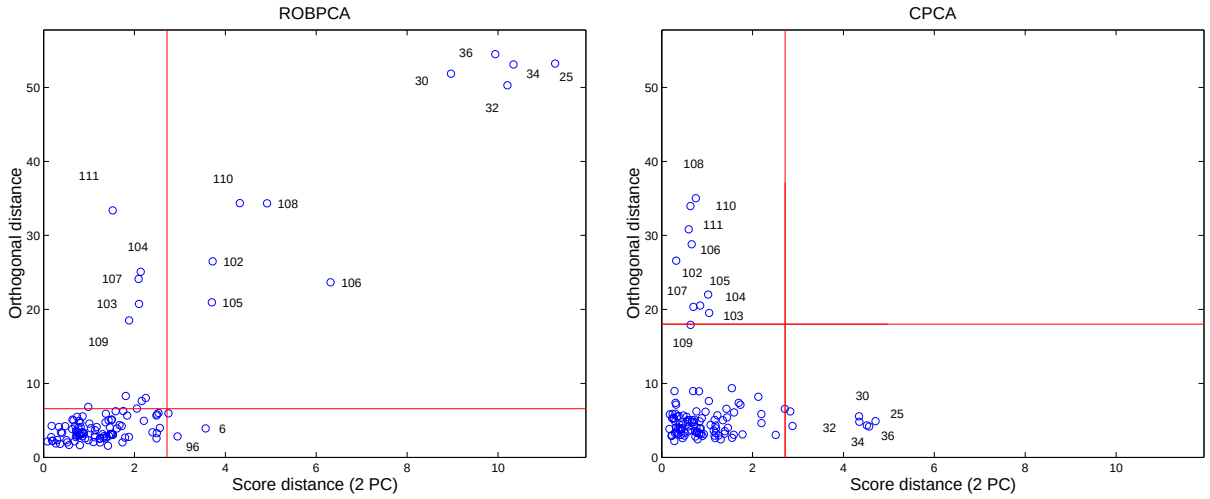
Li and Chen (1985) proposed the idea of projection pursuit for PCA, but their algorithm has a high computational cost. More attractive methods have been developed in Croux and Ruiz-Gazen (2000), but in high dimensions, these algorithms still have a numerical inaccuracy. Therefore, Hubert et al. (2002) have developed RAPCA, which is a fast two-step algorithm. It searches for the direction on which the projected observations have the largest robust scale, and then removes this dimension and repeats.

The spherical and ellipsoidal PCA method provide a very fast algorithm to perform robust PCA. After robustly centering the data, the observations are projected on a sphere (spherical PCA) or an ellipse (ellipsoidal PCA). The principal components are then derived as the eigenvectors of the covariance matrix of these projected data points. SPHER and ELL do not yield estimates of the eigenvalues, which makes it impossible to compute score distances. Therefore, in the examples we additionally applied the MCD estimator on the scores to compute robust distances in the PCA subspace.

3.1 Car data

Our first example is the low-dimensional car data set which is available in S-PLUS as the data frame `cu.dimensions`. For $n = 111$ cars, $p = 11$ characteristics were measured such as the length, the width and the height of the car. We first looked at pairwise scatter plots of the variables, and we computed pairwise Spearman rank correlations $\rho_S(X_i, X_j)$. This preliminary analysis already indicated that there are high correlations among the variables, e.g. $\rho_S(X_1, X_2) = 0.83$ and $\rho_S(X_3, X_9) = 0.87$. Hence, PCA seems an appropriate method to find the most important sources of variation in this data set.

When applying ROBPCA to these data, an important choice we need to make is how many principal components to keep. This is done by means of the eigenvalues $\tilde{l}_1 \geq \tilde{l}_2 \geq ... \geq \tilde{l}_r$ of S_0 with $r = \text{rank}(S_0)$, as obtained in the second stage of the algorithm (see also (10) in the Appendix). We can use these eigenvalues in various ways. For instance, we can look at the *scree plot*, which is a graph of the (monotone decreasing) eigenvalues (Joliffe 1986). We



(a) (b)

Figure 2: Diagnostic plot of the car data set based on (a) two robust principal components; (b) two classical principal components.

can also use a selection criterion, e.g. to choose k such that

$$\sum_{j=1}^{k} \tilde{l}_j / \sum_{j=1}^{r} \tilde{l}_j \approx 90\% \tag{5}$$

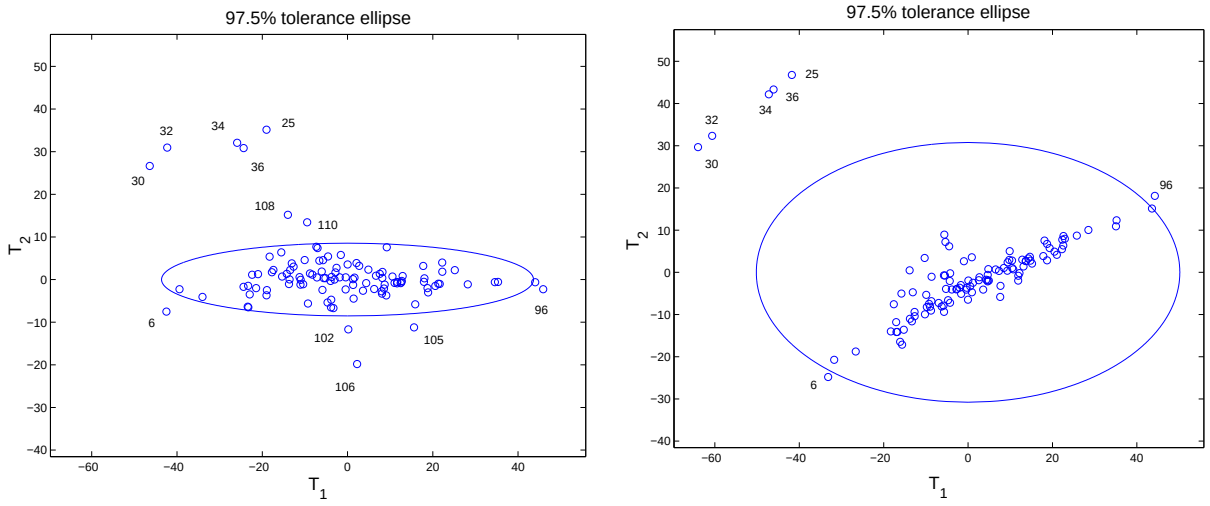
or for instance such that

$$\tilde{l}_k / \tilde{l}_1 \geq 10^{-3}. \tag{6}$$

Here we decided to retain $k = 2$ components based on criterion (5), because $(\tilde{l}_1 + \tilde{l}_2) / \sum_1^{11} \tilde{l}_j = 94\%$.

Figure 2(a) shows the resulting diagnostic plot. We can distinguish a group of orthogonal outliers (labelled 103–104, 107, 109, 111) and two groups of bad leverage points (cases 102, 105–106, 108, 110 and observations 25, 30, 32, 34, 36). A few good leverage points are also visible (6 and 96). If we look at the measurements, we notice that the five most important bad leverage points (25, 30, 32, 34, and 36) have the value -2 on four of the 11 original variables, namely $X_6 = \text{Rear.Hd}, X_8 = \text{Rear.Seat}, X_{10} = \text{Rear.Shld}, X_{11} = \text{luggage}$. None of the other observations share this property. The observations 102-111 have the value -2 for the last variable $X_{11} = \text{luggage}$ (and observation 109 the value -3).

Let us compare this robust result with a classical PCA analysis. The first two components account for 85% of the total variance. The diagnostic plot in Figure 2(b) looks completely



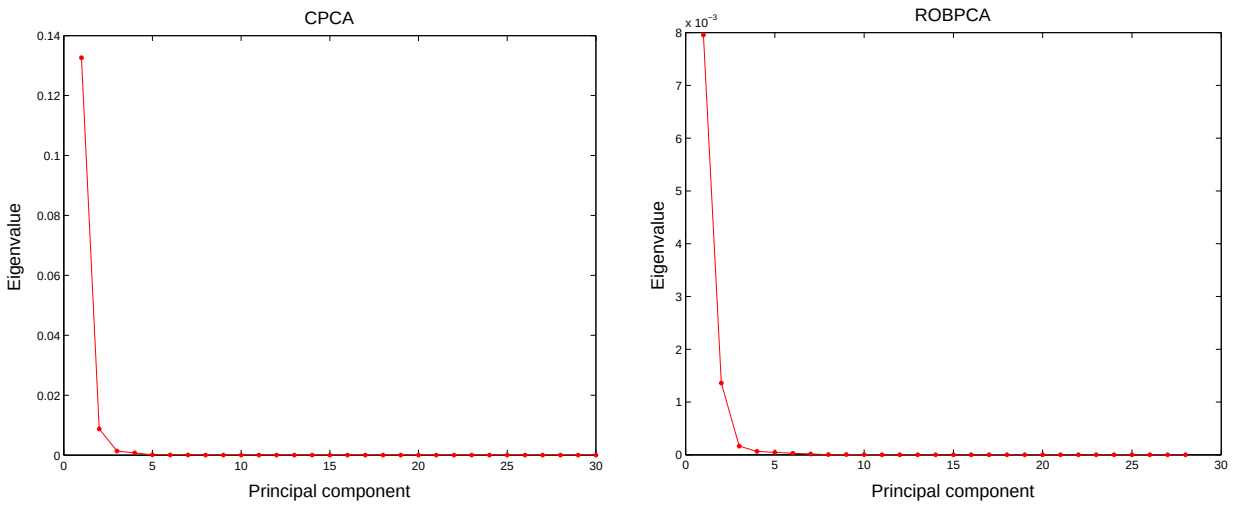
(a) (b)

Figure 3: The score plot with the 97.5% tolerance ellipse of the car data set for (a) ROBPCA; (b) CPCA.

different from the robust one, although the same set of outliers is detected. The most striking difference is that the group of bad leverage points from ROBPCA is converted into good leverage points. This shows how the subspace found by CPCA is attracted towards these bad leverage points.

Some of the differences between ROBPCA and CPCA are also visible in the plot of the scores (t_{i1}, t_{i2}) for all $i = 1, \ldots, n$. Figure 3(a) shows the score plot of ROBPCA, together with the 97.5% tolerance ellipse which is defined as the set of vectors in $\mathbb{R}^2$ whose score distance is equal to $\sqrt{\chi^2_{2,0.975}}$. Data points which fall outside the tolerance ellipse are by definition the good and the bad leverage points. We clearly see how well the robust tolerance ellipse encloses the regular data points. Figure 3(b) is the score plot obtained with CPCA. The corresponding tolerance ellipse is highly inflated toward the outliers 25, 30, 32, 34 and 36. The resulting eigenvectors are not lying in the direction of the highest variability of the other points. We also see how the second eigenvalue of CPCA is blown up by the same set of outliers.

We also performed robust PCA to this low-dimensional data set using the eigenvectors and eigenvalues of the MCD covariance matrix. The resulting diagnostic plot was almost identical to the ROBPCA plot and is therefore not included. Also the other robust methods detected the same set of outliers.



(a) (b)

Figure 4: Scree plot of the octane data set with (a) CPCA; (b) ROBPCA.

3.2 Octane data

Our second example is the octane data set described in Esbensen et al. (1994). It contains near-infrared (NIR) absorbance spectra over $p = 226$ wavelengths of $n = 39$ gasoline samples with certain octane numbers. It is known that six of the samples (25, 26, 36–39) contain added alcohol.

Both the classical scree plot, shown in Figure 4(a), and the ROBPCA scree plot in Figure 4(b) suggest to retain two principal components.

The CPCA diagnostic plot is shown in Figure 5(a). We see that the classical analysis only detects the outlying spectrum 26, which even does not stick out much above the border line. In contrast, we immediately clearly spot the six samples with added alcohol on the ROBPCA diagnostic plot in Figure 5(b). The first principal component from CPCA is clearly attracted by the six outliers, yielding a classical eigenvalue of 0.13. On the other hand the first robust eigenvalue l_1 is only 0.01.

Next, we wondered whether the robust loadings would be influenced by the outlying spectra if we would retain more than two components. To avoid the curse of dimensionality with $n = 39$ observations, it is generally advised that $n > 5k$ (see Rousseeuw and Van Zomeren 1990) so we considered $k_{\max} = 7$. From the robust diagnostic plot in Figure 5(c) we see that the outliers are still very far from the estimated robust subspace.

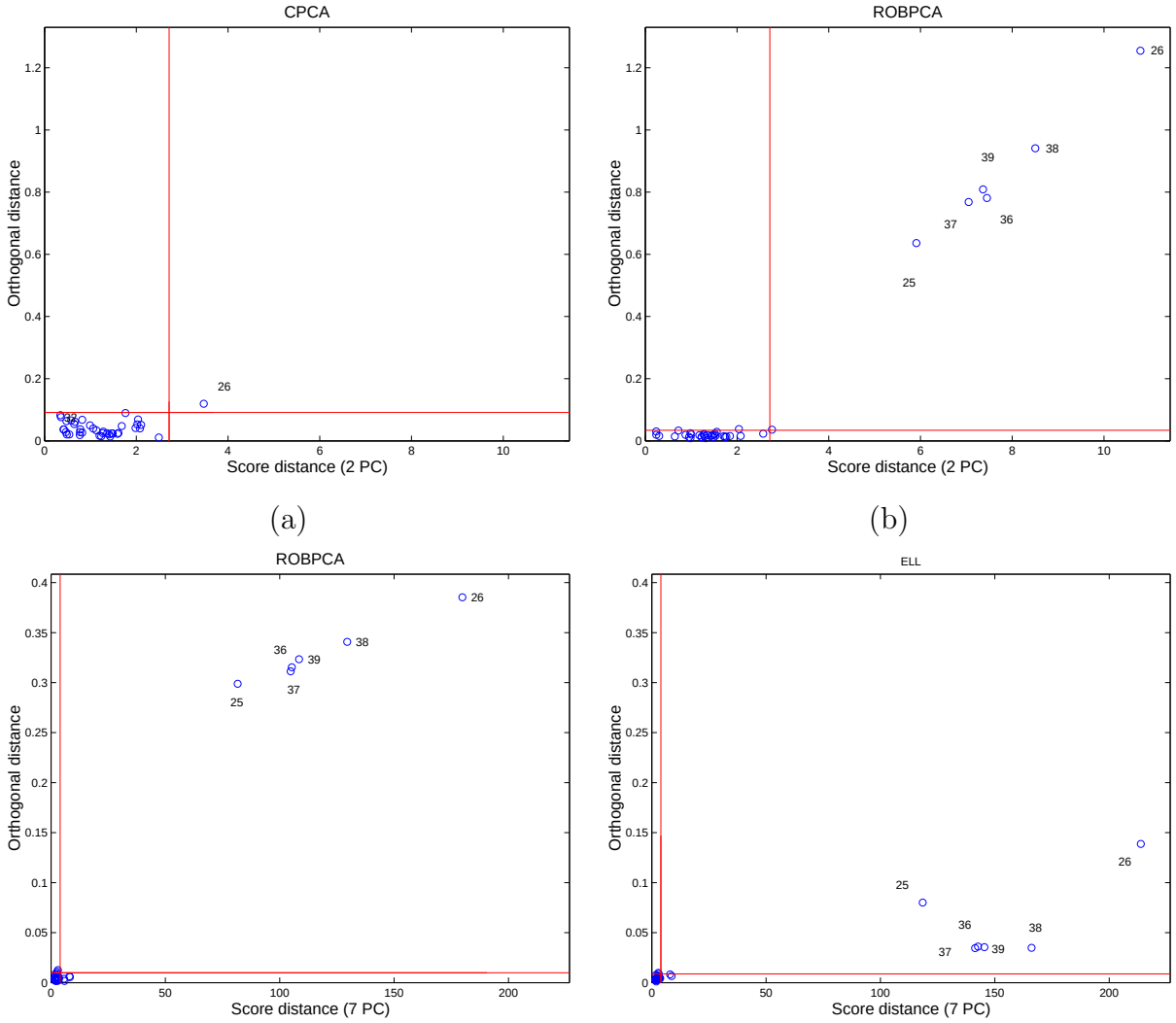


Figure 5: Diagnostic plot of the octane data set based on (a) two CPCA principal components; (b) two ROBPCA principal components; (c) seven ROBPCA principal components; (d) seven ELL principal components.

The diagnostic plots of RAPCA, SPHER and ELL were similar to Figure 5(b) for $k = 2$. But when we selected $k = 7$ components with ELL, we see from Figure 5(d) that the outliers have a much lower orthogonal distance. This illustrates their leverage effect on the estimated principal components.

3.3 Glass spectra

Our third data set consists of EPXMA spectra over $p = 750$ wavelengths collected on 180 different glass samples (Lemberge et al. 2000). The chemical analysis was performed using a Jeol JSM 6300 scanning electron microscope equipped with an energy-dispersive Si(Li) X-ray detection system (SEM-EDX).

We first performed ROBPCA with default value $h = 0.75n = 135$. However, the diagnostic plots then revealed a large amount of outliers. Therefore we analyzed the data set a second time with $h = 0.70n = 126$. Three components are retained for CPCA and ROBPCA yielding a classical explanation percentage of 99% and a robust explanation percentage (5) of 96%. We then obtain the diagnostic plots in Figure 6. From the classical diagnostic plot in Figure 6(a), we see that CPCA does not find important outliers. On the other hand the ROBPCA plot of Figure 6(b) clearly distinguishes two major groups in the data, a smaller group of bad leverage points, a few orthogonal outliers and the isolated case 180 in between the two major groups. A high-breakdown method such as ROBPCA treats the smaller group with cases 143–179 as one set of outliers. Later, it turned out that the window of the detector system had been cleaned before the last 38 spectra were measured. As a result of this less radiation (X-rays) is absorbed and more can be detected, resulting in higher X-ray intensities. If we look at the spectra, we can indeed observe these differences. The regular samples, shown in Figure 7(a), clearly have lower measurements at the channels 160–175 than the samples 143–179 of Figure 7(b). The spectrum of case 180 (not shown) was somewhat in between. Note that instead of plotting the raw data, we first robustly centered the spectra by subtracting the univariate MCD location estimator from each wavelength. Doing so we can observe more of the variability which is present in the data.

The other bad leverage points (57–63) and (74-76) are samples with a large concentration of calcic. In Figure 7(c) we see that their calcic alpha peak (around channels 340-370) and calcic beta peak (channels 375-400) is higher than for the other glass vessels. The orthogonal outliers (22, 23 and 30) whose spectra are shown in Figure 7(d) are rather boundary cases,

although they have larger measurements at the channels 215-245. This might indicate a larger concentration of phosphor.

RAPCA yielded a diagnostic plot similar to the ROBPCA plot. SPHER en ELL are also able to detect the outliers, as can be seen from Figure 6(c) and (d), but they turn the bad leverage points into good leverage points and orthogonal outliers.

4 Simulations

We conducted a simulation study to compare the performance and the robustness of ROBPCA with the four other principal component methods introduced in Section 3: classical PCA (CPCA), RAPCA (Hubert et al. 2001), and spherical (SPHER) and elliptical (ELL) PCA (Locantore et al. 1999).

We generated 1000 samples of size n from the contamination model

$$(1-\varepsilon)N_p(\mathbf{0},\Sigma)+\varepsilon N_p(\tilde{\boldsymbol{\mu}},\tilde{\Sigma})$$

or

$$(1-\varepsilon)t_5(\mathbf{0},\Sigma)+\varepsilon t_5(\tilde{\boldsymbol{\mu}},\tilde{\Sigma})$$

for different values of $n, p, \varepsilon, \Sigma, \tilde{\boldsymbol{\mu}}$ and $\tilde{\Sigma}$. That is, $n(1-\varepsilon)$ of the observations were generated from the p-variate gaussian distribution $N_p(\mathbf{0},\Sigma)$ or the p-variate elliptical $t_5(\mathbf{0},\Sigma)$ distribution and $n\varepsilon$ of the observations were generated from $N_p(\tilde{\boldsymbol{\mu}},\tilde{\Sigma})$ or from $t_5(\tilde{\boldsymbol{\mu}},\tilde{\Sigma})$.

Note that the consistency factor in the FAST-MCD algorithm, which is used within ROBPCA, is constructed under the assumption that the regular observations are normally distributed. Then the denominator equals $\chi^2_{k,1-\alpha}$, the $(1-\alpha)$ quantile of the χ^2 distribution with k degrees of freedom. Hence, the best results of the simulations with t_ν (and here $\nu=5$) is obtained by replacing the denominator with $k((\nu-2)/\nu)F_{k,\nu,1-\alpha}$, with $F_{k,\nu,1-\alpha}$ the $(1-\alpha)$ quantile of the F distribution with k and ν degrees of freedom. However, in real examples any foreknowledge of the true underlying distribution is mostly unavailable. Therefore, and also to make a fair comparison with RAPCA, we did not adjust the consistency factor.

In Tables 1–2 and Figures 8–12 we report some typical results, that were obtained in the following situations:

1. $n=100, p=4$, $\Sigma=\mathrm{diag}(8,4,2,1)$ and $k=3$ (because then $(\sum_1^3\lambda_i)/(\sum_1^4\lambda_i)=93.3\%$.)

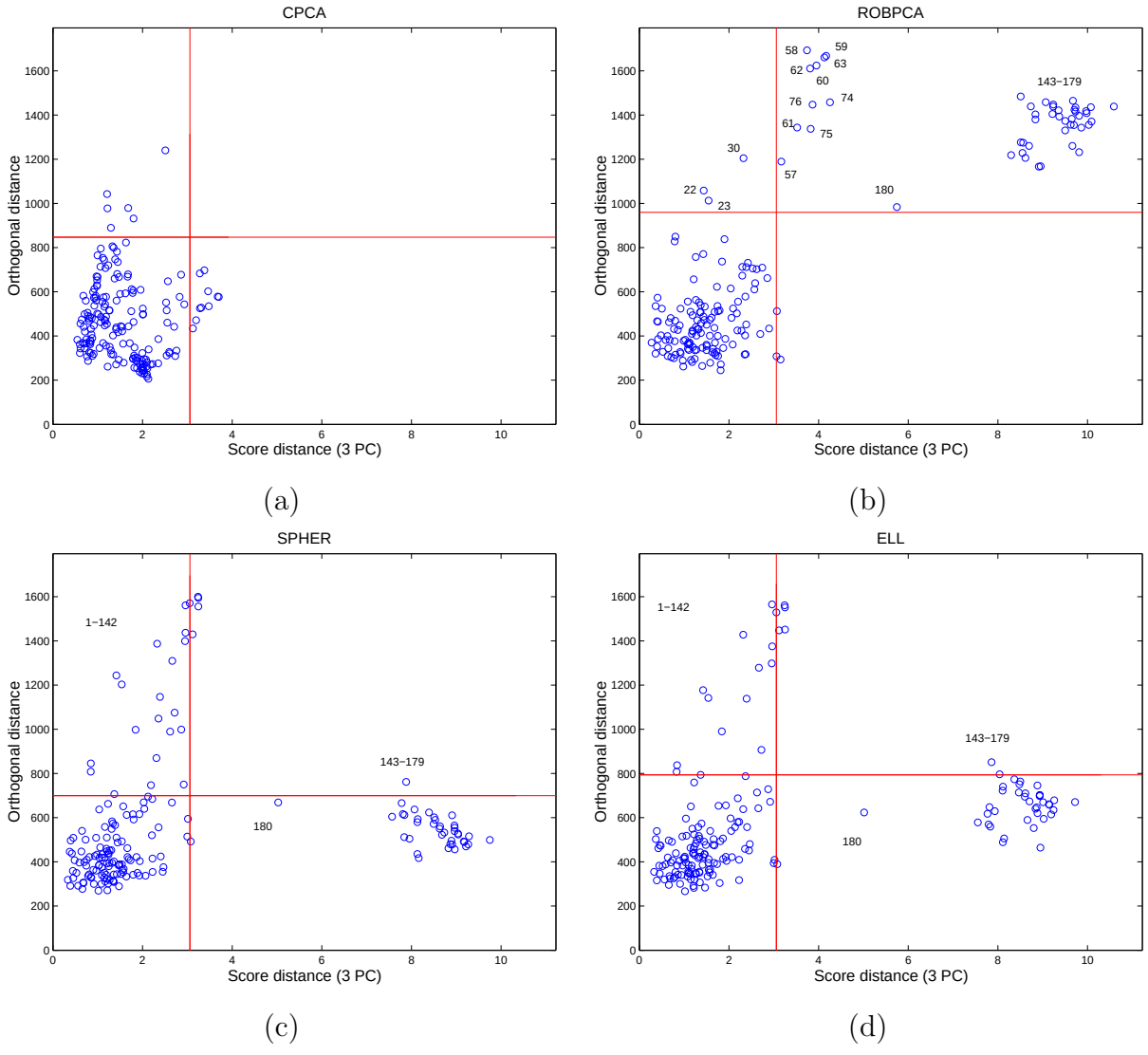
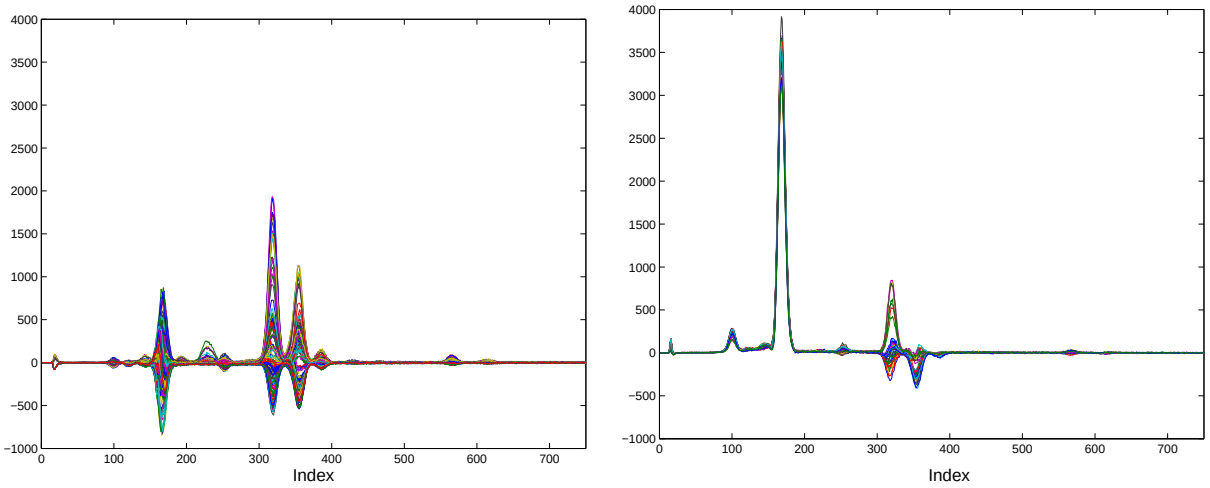
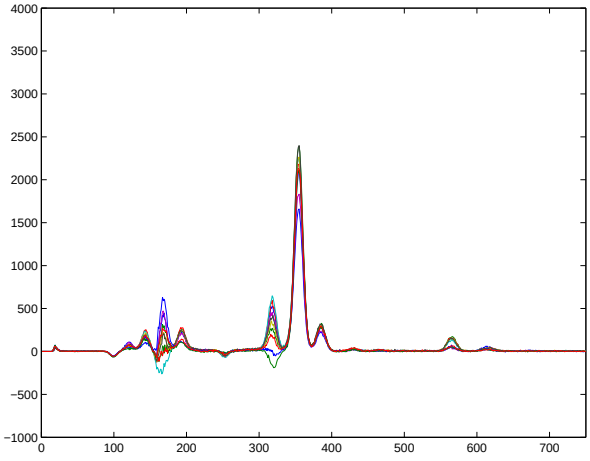


Figure 6: Diagnostic plot of the glass data set based on three principal components computed with (a) CPCA; (b) ROBPCA; (c) SPHER and (d) ELL.

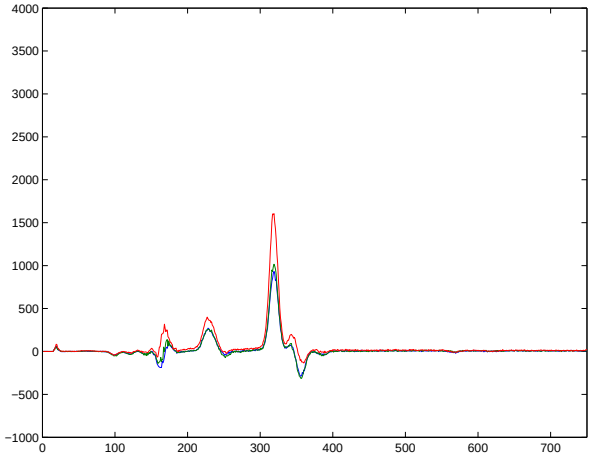


(a)

(b)



(c)



(d)

Figure 7: The glass data set (a) regular observations; (b) bad leverage points 143–179; (c) bad leverage points 57–63 and 74-76; (d) orthogonal outliers 22, 23 and 30.

(4a) $\varepsilon = 0$ (no contamination).

(4b) $\varepsilon = 10\%$ or $\varepsilon = 20\%$, $\tilde{\boldsymbol{\mu}} = f_1 \mathbf{e}_4 = (0, 0, 0, f_1)', \tilde{\Sigma} = \frac{\Sigma}{f_2}$ where $f_1 = 6, 8, 10, \ldots, 20$ and $f_2 = 1$ or $f_2 = 15$.

2. $n = 50, p = 100$, $\Sigma = \text{diag}(17, 13.5, 8, 3, 1, 0.095, \ldots, 0.002, 0.001)'$ and $k = 5$ (because here $(\sum_1^5 \lambda_i)/(\sum_1^{100} \lambda_i) = 90.3\%$.)

(100a) $\varepsilon = 0$ (no contamination).

(100b) $\varepsilon = 10\%$ or $\varepsilon = 20\%$, $\tilde{\boldsymbol{\mu}} = f_1 \mathbf{e}_6, \tilde{\Sigma} = \frac{\Sigma}{f_2}$ where $f_1 = 6, 8, 10, \ldots, 20$ and $f_2 = 1$ or $f_2 = 15$.

Note that $\varepsilon = 0\%$ also corresponds to $f_1 = 0$ and $f_2 = 1$. The subspace spanned by the first k eigenvectors of Σ is denoted by $E_k = \text{span}\{\mathbf{e}_1, \ldots, \mathbf{e}_k\}$ with $\mathbf{e}_j$ the jth column of $I_{p,k}$.

The settings (4a) and (4b) consider low-dimensional data ($p = 4$) of not too small size $n = 100$, whereas in (100a) and (100b) we generate high-dimensional data with $n = 50$ being rather small, and even less than $p = 100$. In the settings (4b) and (100b) the contaminated data are shifted by a distance f_1 in the direction of the $(k+1)$th principal component. We started with $f_1 = 6$, otherwise the outliers could not be distinguished from the regular data points. The factor f_2 determines how strongly the contaminated data are concentrated. In Rocke and Woodruff (1996) it is shown that shifted outliers with the same covariance structure as the regular points are the most difficult ones to detect. This situation corresponds with $f_2 = 1$. Note that because of the orthogonal equivariance of the ROBPCA method, we only need to consider diagonal covariance matrices.

For each simulation setting, we summarized the result of the estimation procedure (CPCA, RAPCA, SPHER, ELL and ROBPCA) as follows:

- for each method we consider the maximal angle between E_k and the estimated PCA subspace, which is spanned by the columns of $P_{p,k}$. Krzanowski (1979) proposed a measure to calculate this angle, which we will denote by maxsub:

$$\text{maxsub} = \arccos(\sqrt{\lambda_k})$$

where λ_k is the smallest eigenvalue of $I'_{k,p} P_{p,k} P'_{k,p} I_{p,k}$. It represents the largest angle between a vector in E_k and the vector most parallel to it in the estimated PCA subspace. To standardize this value, we have divided it by $\frac{\pi}{2}$.

- we compute the proportion of variability that is explained by the estimated eigenvalues. This is done by comparing the sum of the k largest eigenvalues to the sum of all p known eigenvalues. We report the mean proportion of explained variability:

$$\frac{1}{1000}\sum_{l=1}^{1000}\frac{\hat{\lambda}_1^{(l)}+\hat{\lambda}_2^{(l)}+\ldots+\hat{\lambda}_k^{(l)}}{\lambda_1+\lambda_2+\ldots+\lambda_k+\ldots\lambda_p}$$

  where $\hat{\lambda}_j^{(l)}$ is the estimated value of λ_j at the lth replication. It would be more elegant if the denominator also contained the estimated eigenvalues, but ROBPCA and RAPCA only estimate the first k eigenvalues. We report these results for the settings without contamination, and for a specific situation with 10% contamination ($f_1 = 10, f_2 = 1$). As SPHER and ELL only estimate the principal components and not their eigenvalues, they are not included in the comparison.

- for the k largest eigenvalues we also compute their mean squared error (MSE) defined as

$$\text{MSE}(\hat{\lambda}_j) = \frac{1}{1000}\sum_{l=1}^{1000}(\hat{\lambda}_j^{(l)} - \lambda_j)^2.$$

  We only report the results for $f_2 = 1$, because they were very similar for $f_2 = 15$.

The ideal value of maxsub and MSE in the tables and figures is thus 0. For the mean proportion of explained variability the optimal values are 93.3% for low-dimensional data and 90.3% for high-dimensional data.

Table 1: Simulation results of maxsub in settings (4a) and (100a) when there is no contamination.

Distribution	n	p	CPCA	RAPCA	SPHER	ELL	ROBPCA
normal	100	4	0.094	0.160	0.127	0.087	0.176
	50	100	0.215	0.707	0.272	0.213	0.282
t_5	100	4	0.130	0.183	0.127	0.086	0.133
	50	100	0.308	0.701	0.272	0.213	0.311

Table 1 reports the simulation results of maxsub for the settings (4a) and (100a). We see that elliptical PCA yields the best results for maxsub when there is no contamination. For low-dimensional data, the results for the other methods are more or less comparable, whereas for high-dimensional data, RAPCA is clearly the less efficient approach.

Table 2: Simulation results of the mean proportion of explained variability when there is no contamination and with 10% contamination ($f_1 = 10$ and $f_2 = 1$).

	Multivariate normal			Multivariate t_5		
	CPCA	RAPCA	ROBPCA	CPCA	RAPCA	ROBPCA
	$n = 100$, $p = 4$					
$\epsilon = 0\%$	93.4%	94.7%	83.9%	98.7%	72.1%	60.2%
$\epsilon = 10\%$	147.8%	112.9%	88.5%	135.2%	86.8%	67.5%
	$n = 50$, $p = 100$					
$\epsilon = 0\%$	91.6%	83.6%	79.5%	99.2%	65.1%	57.3%
$\epsilon = 10\%$	109.4%	86.1%	79.3%	110.9%	66.9%	56.7%

From Table 2 we see that CPCA provides the best mean proportion of explained variability when there is no contamination in the data. RAPCA attains higher values than ROBPCA for both distributions. When contamination is added to the data the eigenvalues obtained with CPCA are overestimated, which results in estimated percentages which are even larger than 100%! The robust methods are much less sensitive to the outliers, but also RAPCA attains a value larger than 100% at the contaminated low-dimensional normal distribution. Note that when the consistency factor in ROBPCA is adapted to the t_5 distribution, the results improve substantially. For the low-dimensional data we obtain 80% without and 82.8% with contamination, whereas in high dimensions the mean percentage of explained variability is 69.3% and 69.4% respectively.

The results of the maxsub measure for simulations (4b) and (100b) are summarized in Figures 8–11. In every situation, CPCA clearly fails and it provides the worst possible result because maxsub is always very close to 1. This implies that the estimated PCA subspace has been attracted by outliers in such a way that at least one principal component is orthogonal to E_k. Also RAPCA, SPHER and ELL are clearly influenced by the outliers, the most strongly

when the data are high-dimensional or when there is a large percentage of contamination. In all situations, ROBPCA outperforms the other methods. ROBPCA only attains high values for maxsub at the long–tailed t_5 when f_1 is between 6 and 8. This is because in this case the outliers are not yet very well separated from the regular data group. Also the other methods fail in such a situation. As soon as the contamination lies somewhat further, ROBPCA is capable to distinguish the outliers and maxsub remains almost constant.

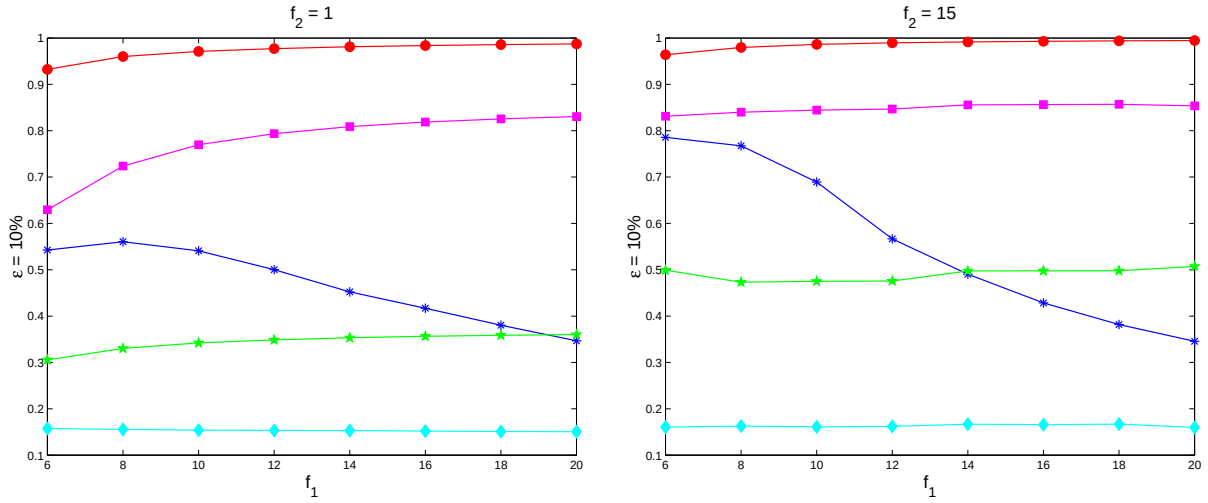
Finally, some results for the MSE's of the eigenvalues are summarized in Figure 12. Here we display the ratio of the MSE's of CPCA versus ROBPCA, and RAPCA versus ROBPCA, for the normally distributed data with $\varepsilon = 10\%$ contamination and $f_2 = 1$. Figures 12(a) and 12(b) show the results for the low-dimensional data, whereas Figures 12(c) and 12(d) present the results for the high-dimensional ones. When we compare CPCA and ROBPCA, we see that the MSE of the first CPCA eigenvalue increases strongly when the contamination is shifted further away from the regular points. Also the MSE's of the other CPCA eigenvalues are much larger than those of ROBPCA. Only MSE($\hat{\lambda}_2$) and MSE($\hat{\lambda}_3$) in Figure 12(c) are of the same order of magnitude.

Figures 12(b) and 12(d) show the superiority of ROBPCA over RAPCA. At high-dimensional data the differences are most prominent in the fifth eigenvalue. This explains the bad results for maxsub obtained with RAPCA in this situation. The first four eigenvalues (and their eigenvectors) are well estimated, but the fifth one is clearly attracted by the outliers.

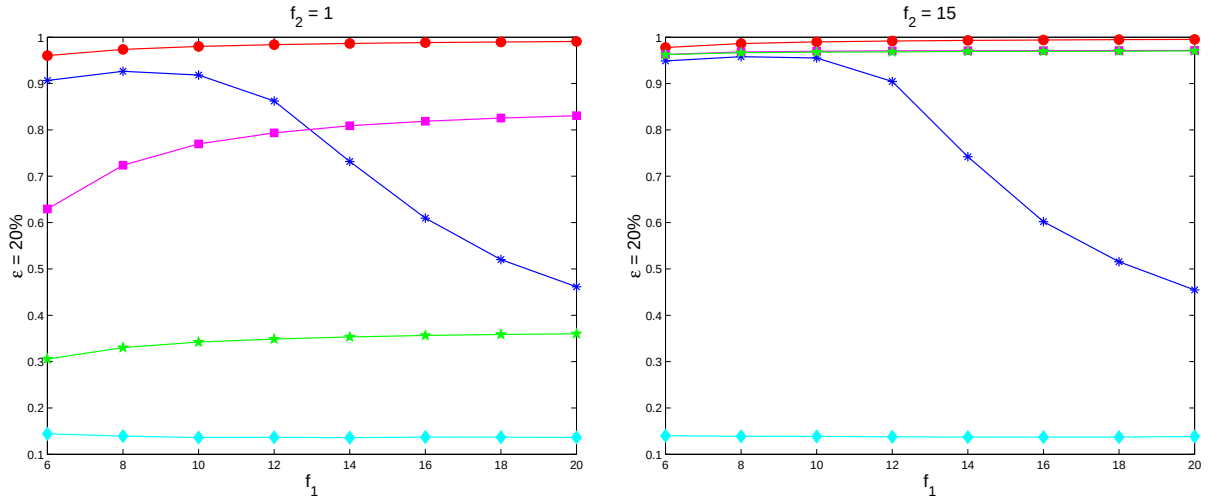
Computation time

Although ROBPCA is slower than the other methods discussed in this paper, its computation time is still very low. Our Matlab implementation needs only 3.19 seconds for the car data set ($n = 111, p = 11$), 3.06 seconds for the octane data set ($n = 39, p = 226$) and 4.16 seconds for the glass data set ($n = 180, p = 750$) on a Pentium IV with 2.40 GHz.

Figure 13(a) gives the mean CPU-time in seconds over 100 runs for different low-dimensional normal data. The sample sizes vary from 50 to 5000, and p is relatively small ($p = 4$ or $p = 10$). We see that the computation time is linear in n and k. From Figure 13(b), the same conclusion can be drawn for high-dimensional data sets. We also looked at the effect of varying p while holding $n = 100$ and $k = 4$ constant. Then the mean CPU-time was 3.2 seconds for $p = 10$ and increased to only 4.3 seconds for $p = 3000$.

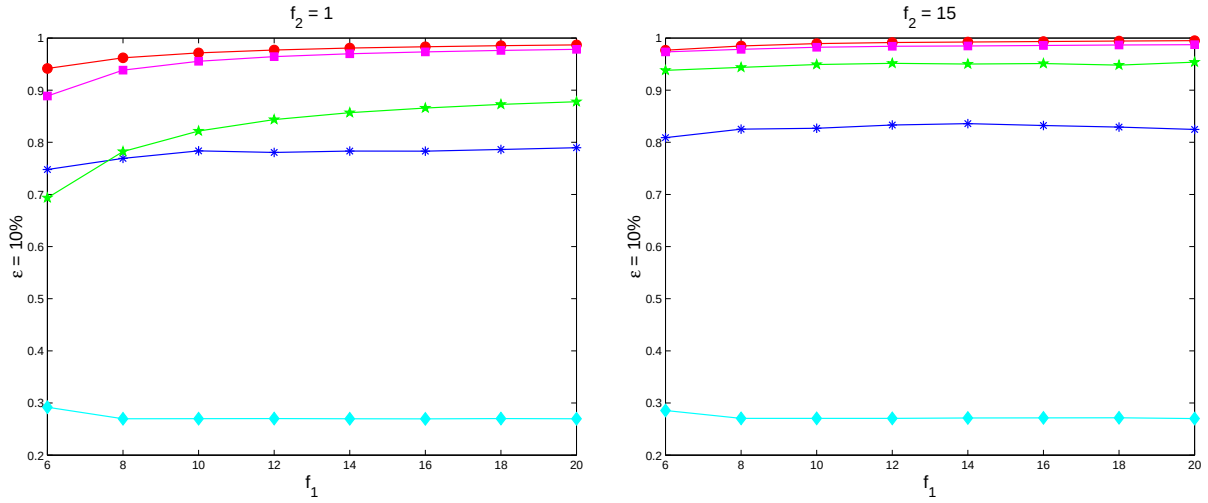


(a) (b)

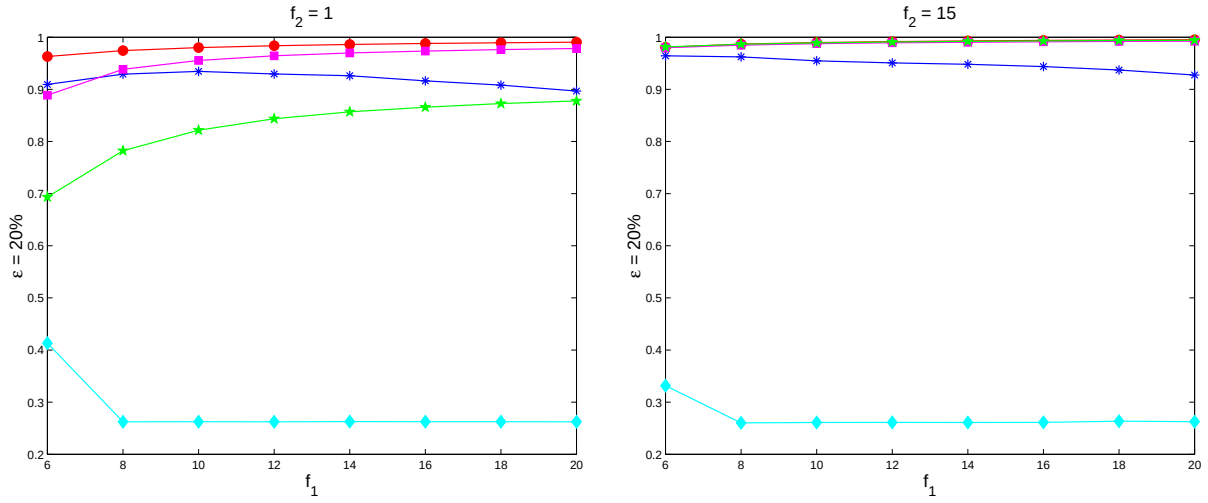


(c) (d)

Figure 8: The value maxsub of the low-dimensional multivariate normal data for (a) $\epsilon = 10\%$ and $f_2 = 1$; (b) $\epsilon = 10\%$ and $f_2 = 15$; (c) $\epsilon = 20\%$ and $f_2 = 1$; (d) $\epsilon = 20\%$ and $f_2 = 15$. The curves represent the results for CPCA ($\bullet$), SPHER ($\blacksquare$), ELL ($\star$), RAPCA ($*$) and ROBPCA ($\blacklozenge$).



(a) (b)



(c) (d)

Figure 9: The value maxsub of the high-dimensional multivariate normal data for (a) $\epsilon = 10\%$ and $f_2 = 1$; (b) $\epsilon = 10\%$ and $f_2 = 15$; (c) $\epsilon = 20\%$ and $f_2 = 1$; (d) $\epsilon = 20\%$ and $f_2 = 15$. The curves represent the results for CPCA ($\bullet$), SPHER ($\blacksquare$), ELL ($\star$), RAPCA ($*$) and ROBPCA ($\blacklozenge$).

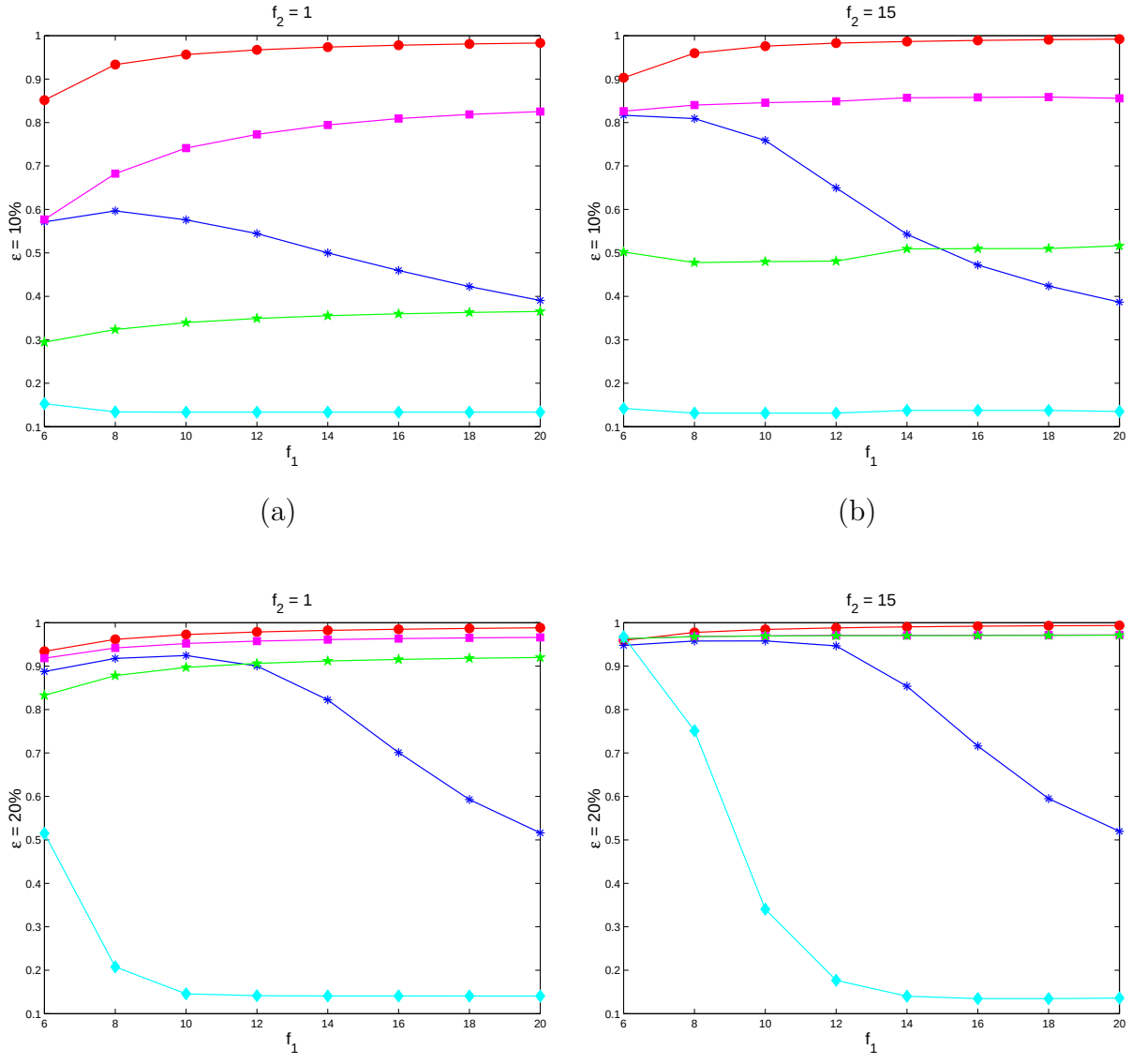
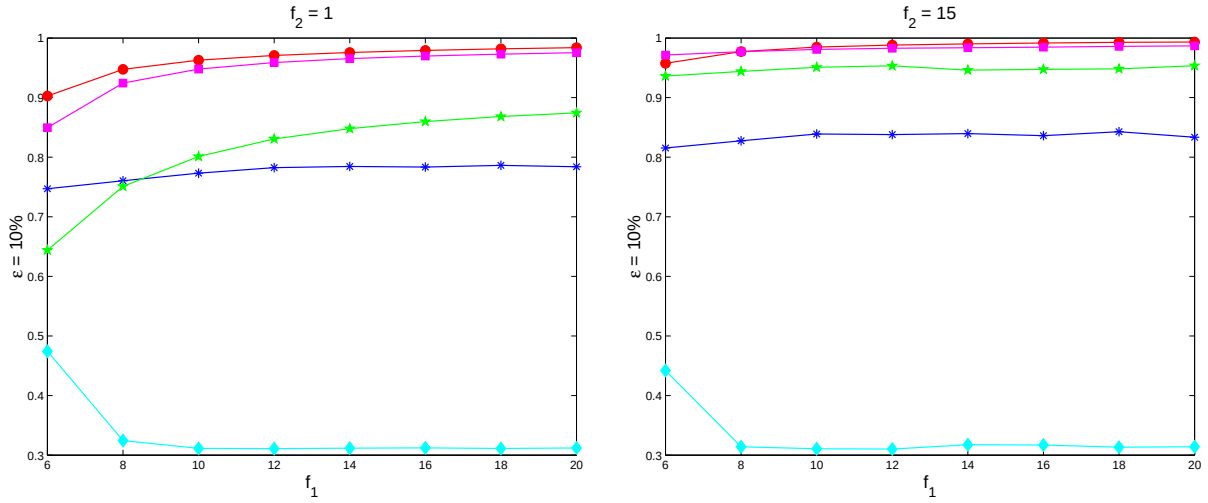
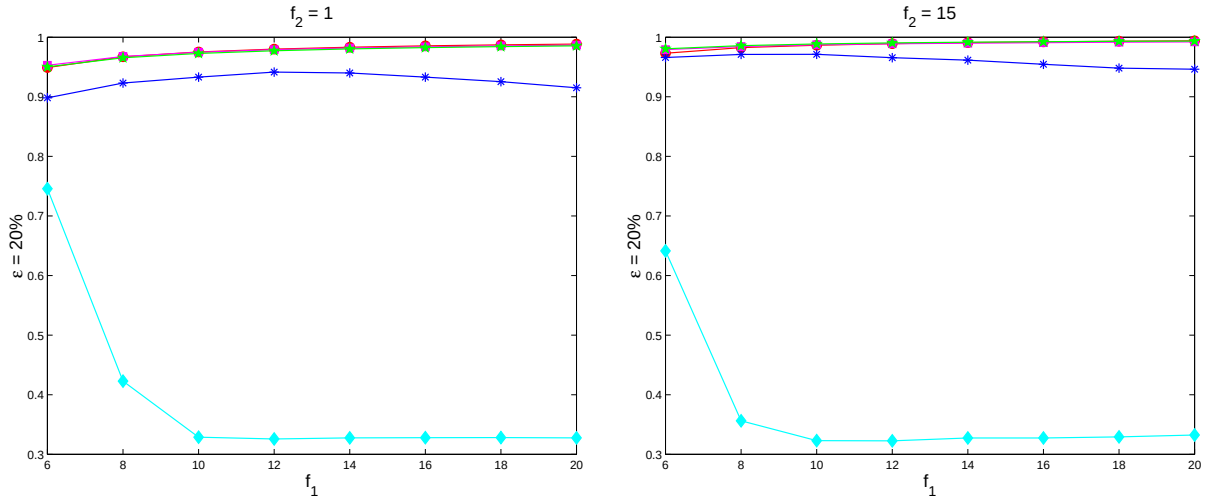


Figure 10: The value maxsub of the low-dimensional multivariate t_5 data for (a) $\epsilon = 10\%$ and $f_2 = 1$; (b) $\epsilon = 10\%$ and $f_2 = 15$; (c) $\epsilon = 20\%$ and $f_2 = 1$; (d) $\epsilon = 20\%$ and $f_2 = 15$. The curves represent the results for CPCA ($\bullet$), SPHER ($\blacksquare$), ELL ($\star$), RAPCA ($*$) and ROBPCA ($\blacklozenge$).



(a) (b)



(c) (d)

Figure 11: The value maxsub of the high-dimensional multivariate t_5 data for (a) $\epsilon = 10\%$ and $f_2 = 1$; (b) $\epsilon = 10\%$ and $f_2 = 15$; (c) $\epsilon = 20\%$ and $f_2 = 1$; (d) $\epsilon = 20\%$ and $f_2 = 15$. The curves represent the results for CPCA ($\bullet$), SPHER ($\blacksquare$), ELL ($\star$), RAPCA ($*$) and ROBPCA ($\blacklozenge$).

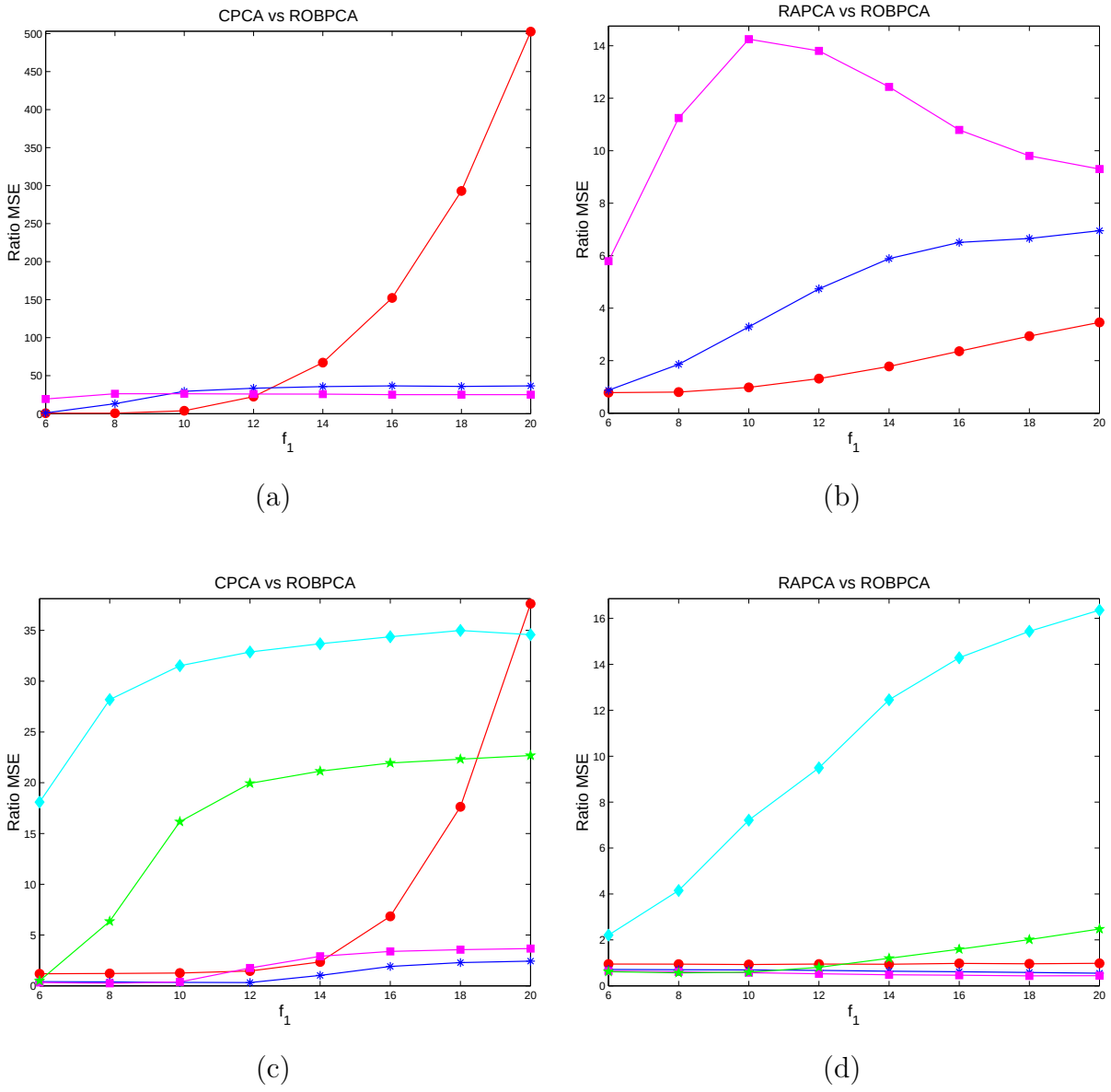
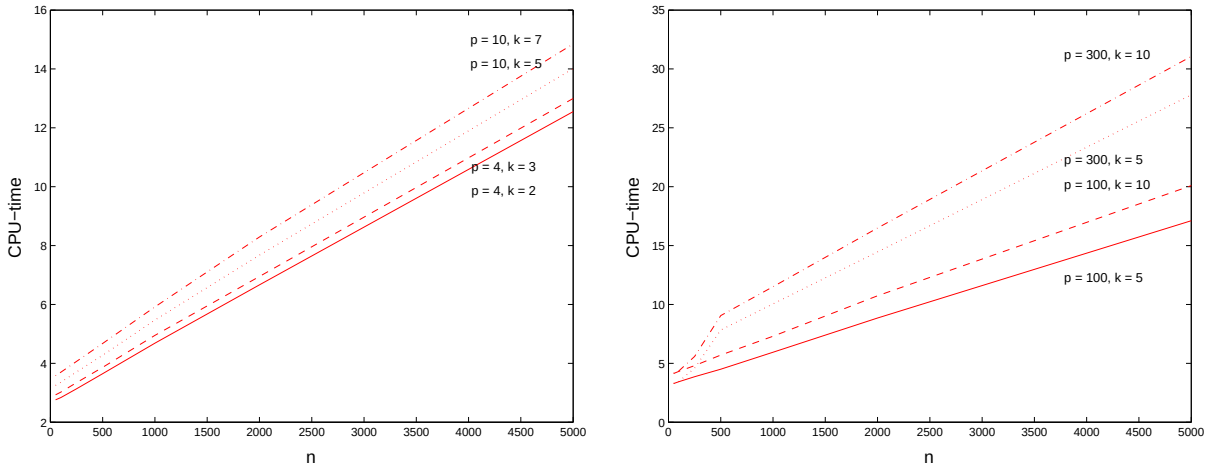


Figure 12: The ratio of the MSE's of the normal data, $\varepsilon = 10\%$, and $f_2 = 1$ for (a) CPCA versus ROBPCA and (b) RAPCA versus ROBPCA for the low-dimensional data; (c) CPCA versus ROBPCA and (d) RAPCA versus ROBPCA for the high-dimensional data. The curves represent the ratio of the MSE of $\hat{\lambda}_1$ ($\bullet$), $\hat{\lambda}_2$ ($\blacksquare$), $\hat{\lambda}_3$ ($\star$), $\hat{\lambda}_4$ ($*$) and $\hat{\lambda}_5$ ($\blacklozenge$).



(a) (b)

Figure 13: The mean CPU-time in seconds over 100 runs of ROBPCA for (a) low-dimensional data; (b) high-dimensional data.

5 Conclusion and Outlook

We have constructed a fast and robust algorithm for principal component analysis of high-dimensional data. It first applies projection pursuit techniques in the original data space. These results are then used to project the observations into a subspace of small to moderate dimension. Within this subspace we then apply ideas of robust covariance estimation. Throughout we are able to detect exact fit situations and to reduce the dimension accordingly. Simulations and applications to real data show that this ROBPCA algorithm yields very robust estimates when the data contains outliers. The associated diagnostic plot is a useful graphical tool which allows to visualize and classify the outliers.

As mentioned in the introduction, data analysis often starts with PCA. We have used a robust PCA before applying a robust discriminant analysis technique (Hubert and Van Driessen 2003, Hubert and Engelen 2003) and a robust method for logistic regression (Rousseeuw and Christmann 2003). Also the use of ROBPCA in robust Principal Components Regression (Hubert and Verboven 2003) and robust Partial Least Squares (Hubert and Vanden Branden 2003) has been investigated. The ROBPCA method thus opens a door to practical robust multivariate calibration and to the analysis of regression data with both outliers and multicollinearity.

The Matlab program *robpca* and auxiliary functions are available at the web sites

`http://win-www.uia.ac.be/u/statis` and
`http://www.wis.kuleuven.ac.be/stat/robust.html` as part of the Matlab toolbox for Robust Calibration.

Acknowledgements

We thank the associate editor and two referees for their constructive remarks on the first version of this paper. We are grateful to Oxana Rodionova for sending us the octane data set, and to Pascal Lemberge for providing the glass data set. Steve Marron kindly gave us the Matlab code of the spherical and elliptical PCA method.

Appendix

This describes the ROBPCA method in detail, following the sketch in Section 2.

Stage 1. As proposed in Hubert et al. (2002), we start by reducing the data space to the affine subspace spanned by the n observations. This is especially useful when $p \geq n$, but even when $p < n$ the observations may span less than the whole p-dimensional space. A convenient way to do this is by a singular value decomposition of the mean-centered data matrix, yielding

$$X_{n,p} - \mathbf{1}_n \hat{\boldsymbol{\mu}}_0' = U_{n,r_0} D_{r_0,r_0} V_{r_0,p}' \tag{7}$$

with $\hat{\boldsymbol{\mu}}_0$ the classical mean vector, $r_0 = \text{rank}(X_{n,p} - \mathbf{1}_n \hat{\boldsymbol{\mu}}_0')$, D an $r_0 \times r_0$ diagonal matrix, and $U'U = I_{r_0} = V'V$ where I_{r_0} is the $r_0 \times r_0$ identity matrix. When $p > n$ we carry out the decomposition in (7) using the kernel approach which is based on computing the eigenvectors and eigenvalues of $(X - \mathbf{1}_n \hat{\boldsymbol{\mu}}_0')(X - \mathbf{1}_n \hat{\boldsymbol{\mu}}_0')'$ (Wu et al. 1997). Since the latter matrix has n rows and columns, its decomposition can be obtained faster than the decomposition of the $p \times p$ matrix $(X - \mathbf{1}_n \hat{\boldsymbol{\mu}}_0')'(X - \mathbf{1}_n \hat{\boldsymbol{\mu}}_0')$.

Without losing any information, we now work in the subspace spanned by the r_0 columns of V. That is, $Z_{n,r_0} = UD$ becomes our new data matrix. Note that this singular value decomposition is just an affine transformation of the data. We do not use it to retain only the first eigenvectors of the covariance matrix of $X_{n,p}$. This would imply that we were performing classical PCA, which is of course not robust. Here, we only represent the data in its own

dimensionality.

Stage 2. In this stage we try to find the $h < n$ 'least outlying' data points. Their covariance matrix will then be used to obtain a preliminary subspace of dimension k_0. The value of h can be chosen by the user, but $n - h$ should exceed the number of outliers in the data set. Moreover h needs to be larger than $[(n+k_0+1)/2]$, for reasons that will be explained in Stage 3 of the algorithm. Because we do not know the number of outliers or k_0 at this moment, we take $h = \max\{[\alpha n], [(n + k_{\max} + 1)/2]\}$ where $k_{\max}$ stands for the maximal number of components that will be computed, and is set to 10 by default. The parameter α can be chosen as any real value between 0.5 and 1. The higher α, the more efficient the estimates will be at uncontaminated data. On the other hand, setting a lower value for α will increase the robustness of the algorithm at contaminated samples. Our default that is also used in the simulations is $\alpha = 0.75$.

To find the h 'least outlying' data points we proceed in the following way:

1. For each data point $\mathbf{x}_i$ we compute its outlyingness. The Stahel-Donoho affine invariant outlyingness (Stahel 1981, Donoho 1982) is defined as

$$\mathrm{outl}_A(\mathbf{x}_i) = \max_{\mathbf{v} \in B} \frac{|\mathbf{x}_i'\mathbf{v} - \mathrm{med}(\mathbf{x}_j'\mathbf{v})|}{\mathrm{mad}(\mathbf{x}_j'\mathbf{v})} \tag{8}$$

   where B contains all non-zero vectors, $\mathrm{med}(\mathbf{x}_j^t\mathbf{v})$ is the median of $\{\mathbf{x}_j'\mathbf{v}, j = 1, \ldots, n\}$ and $\mathrm{mad}(\mathbf{x}_j'\mathbf{v}) = \mathrm{med}|\mathbf{x}_j'\mathbf{v} - \mathrm{med}(\mathbf{x}_l'\mathbf{v})|$. In a PCA analysis we only need an orthogonally invariant measure, so we can restrict the set B to all directions through two data points. If $\binom{n}{2} > 250$ we take at random 250 directions from B. Moreover, we replace the median and the mad in (8) by the univariate MCD location and scale estimator (Rousseeuw 1984), denoted by t_{MCD} resp. s_{MCD}. These estimators are defined as the mean, resp. the standard deviation of the h observations with smallest variance. Both estimators t_{MCD} and s_{MCD} can easily be computed in $\mathrm{O}(n \log(n))$ time (Rousseeuw and Leroy 1987, page 171). Summarizing, for each direction $\mathbf{v} \in B$ we project the n data points $\mathbf{x}_i$ on $\mathbf{v}$ and compute their robustly standardized absolute residual $|\mathbf{x}_i'\mathbf{v} - t_{MCD}(\mathbf{x}_j'\mathbf{v})|/\ s_{MCD}(\mathbf{x}_j'\mathbf{v})$.

   This leads to the orthogonally invariant outlyingness

$$\mathrm{outl}_{\mathrm{O}}(\mathbf{x}_i) = \max_{\mathbf{v} \in B} \frac{|\mathbf{x}_i'\mathbf{v} - t_{MCD}(\mathbf{x}_j'\mathbf{v})|}{s_{MCD}(\mathbf{x}_j'\mathbf{v})}. \tag{9}$$

(a) When all robust scales s_{MCD} are non-zero, we can compute $\text{outl}_{\text{O}}(\mathbf{x}_i)$ for all data points and consider the h observations with smallest outlyingness. Their indices are stored in the set H_0.

(b) When we encounter a direction $\mathbf{v}$ in which the projected observations have zero robust scale, i.e. $s_{MCD}(\mathbf{x}_j'\mathbf{v}) = 0$, we have in fact found a hyperplane $H_{\mathbf{v}}$ orthogonal to $\mathbf{v}$ that contains h observations. This is called an exact fit situation. When this happens we project all the data points on $H_{\mathbf{v}}$, thereby reducing the true dimension by one. To perform this projection we apply the reflection step, described in detail in (Hubert et al. 2002). The reflection step starts by reflecting all data such that the normalized vector $\mathbf{v}/\|\mathbf{v}\|$ coincides with the first basis vector $\mathbf{e}_1$. The projection on the orthogonal complement of $\mathbf{v}$ then simply corresponds to removing the first coordinate of each data point. We then repeat the search for the h least outlying data points in $H_{\mathbf{v}}$, i.e. we go back to Step 1 above.

Note that the exact fit situation can occur more than once, in which case we reduce the working dimension sequentially. We end up with a data set in some dimension $r_1 \leq r_0$ and a set H_0 indexing the h data points with smallest outlyingness. For convenience, we still denote our lower-dimensional data points by $\mathbf{x}_i$.

2. We now consider $\hat{\boldsymbol{\mu}}_1$ and S_0 as the mean and covariance matrix of the h observations in H_0. We will follow the convention that the eigenvalues of any scatter matrix are sorted in descending order, and that the eigenvectors are indexed accordingly. This means that the eigenvector $\mathbf{v}_1$ corresponds to the largest eigenvalue, $\mathbf{v}_2$ to the second largest eigenvalue, and so on. The spectral decomposition of S_0 is denoted as

$$S_0 = P_0 L_0 P_0' \tag{10}$$

with $L = \text{diag}(\tilde{l}_1 \ldots, \tilde{l}_r)$ and $r \leq r_1$.

The covariance matrix S_0 is used to decide how many principal components $k_0 \leq r$ will be retained in the further analysis. We can do this in a variety of ways. For instance, we can look at the scree plot, which is a graph of the (monotone decreasing) eigenvalues, or we can use a selection criterion such as (5) or (6).

3. Finally, we project the data points on the subspace spanned by the first k_0 eigenvectors

of S_0. To implement this step, we set

$$X^*_{n,k_0} = (X_{n,r_1} - \mathbf{1}_n \hat{\boldsymbol{\mu}}_1')P_{r_1,k_0}$$

where P_{r_1,k_0} consists of the first k_0 columns of P_0 in (10).

Stage 3. In the third stage of the algorithm we robustly estimate the scatter matrix of the data points in X^*_{n,k_0} using the MCD estimator. Recall that for this we need to find h data points whose covariance matrix has minimal determinant. Since in general we cannot consider all h-subsets, we must rely on approximate algorithms. Here we slightly adapt the FAST-MCD algorithm of (Rousseeuw and Van Driessen 1999) by taking advantage of the result of Stage 2 which used the outlyingness measure (9).

1. We first apply C-steps starting from the $\mathbf{x}^*_i$ with $i \in H_0$ (the index set H_0 was obtained in Step 2 in Stage 2). The C-step was proposed in Rousseeuw and Van Driessen (1999) where it plays a fundamental role in the fast computation of the MCD estimator. It is defined as follows: let $\mathbf{m}_0$ and C_0 be the mean and the covariance matrix of the h points in H_0. Then,

   (a) if $\det(C_0) > 0$ we compute the robust distances of all data points with respect to $\mathbf{m}_0$ and C_0, denoted as

   $$d_{\mathbf{m}_0,C_0}(i) = \sqrt{(\mathbf{x}^*_i - \mathbf{m}_0)' C_0^{-1} (\mathbf{x}^*_i - \mathbf{m}_0)} \qquad \text{for } i = 1, \ldots, n. \tag{11}$$

   We then define the subset H_1 as the (indices of the) h points with smallest robust distances $d_{\mathbf{m}_0,C_0}(i)$. This subset is then used to compute $\mathbf{m}_1$, C_1 and all robust distances $d_{\mathbf{m}_1,C_1}(i)$. Rousseeuw and Van Driessen (1999) proved that always $\det(C_1) \leq \det(C_0)$. We continue updating the subset until the determinant of the covariance matrix no longer decreases.

   (b) if at some iteration step $m = 0, 1, \ldots$ a covariance matrix C_m is found to be singular, we project the data points on the lower-dimensional space spanned by the eigenvectors of C_m that correspond to its nonzero eigenvalues, and we continue the C-steps inside that space.

Upon convergence we obtain a data matrix, still denoted as X^*_{n,k_1}, with $k_1 \leq k_0$ variables, and indices of the final h-subset which are stored in the set H_1.

2. We now apply the FAST-MCD algorithm to X^*_{n,k_1}. This algorithm draws many random subsets of size $(k_1 + 1)$ out of X^*. In each subset the mean and the covariance matrix are computed. Then the robust distances (11) with respect to this mean and covariance matrix are obtained for all observations. Next, the $(k_1 + 1)$-subset is enlarged to a h-subset by considering the h observations with smallest robust distance. This h-subset is then used to start C-steps. Note that the FAST-MCD algorithm generates quasi-random h-subsets, whereas in Step 1 of Stage 3 we have applied C-steps starting from one specific h-subset H_1.

   The FAST-MCD algorithm contains several time-saving techniques. For instance, it does not apply C-steps until convergence for each h-subset under consideration. Instead, it carries out two C-steps for each, selects the 10 best results, and only iterates fully starting from these. Moreover, when n is large, the algorithm constructs several non-overlapping representative subsets of 300 to 600 cases. It first applies C-steps to those subsets, then the best solutions are used as starts for C-steps in the union of those subsets, and in the end the best solutions are iterated in the whole data set.

   Whereas FAST-MCD draws 500 random subsets of size $(k_1 + 1)$ by default, we use 250 random subsets in the ROBPCA algorithm. This reduces the computation time considerably, and in simulations had almost no effect on the estimates. This is because we already used the outlyingness measure (9) which gave us a very good initial h-subset, allowing us to draw fewer random subsets.

   In this step of the ROBPCA algorithm we could also have used other algorithms for robust covariance estimation. One reason for choosing the FAST-MCD algorithm was that, to the best of our knowledge, it is currently the only algorithm that can deal with exact fit situations. When they occur, the algorithm reduces the working dimension.

   The final data set is denoted by $\tilde{X}_{n,k}$ with $k \leq k_1$. Let $\hat{\boldsymbol{\mu}}_2$ and S_1 denote the mean and covariance matrix of the h-subset found in Step 1, and $\hat{\boldsymbol{\mu}}_3$ and S_2 the mean and covariance matrix found by the FAST-MCD algorithm. If $\det(S_1) < \det(S_2)$ we continue our computations based on $\hat{\boldsymbol{\mu}}_2$ and S_1. For this, we set $\hat{\boldsymbol{\mu}}_4 = \hat{\boldsymbol{\mu}}_2$ and $S_3 = S_1$. Otherwise, we let $\hat{\boldsymbol{\mu}}_4 = \hat{\boldsymbol{\mu}}_3$ and $S_3 = S_2$.

3. Based on $\hat{\boldsymbol{\mu}}_4$ and S_3 we compute a reweighted mean and covariance matrix in order to increase the statistical efficiency. First we multiply S_3 by a consistency factor c_1

to make the estimator unbiased at normal distributions. As proposed by Rocke and Woodruff (1996) we use the consistency factor of Rousseeuw and Van Driessen (1999) adapted with the hth quantile of the robust distances instead of their median, so

$$c_1 = \frac{\{d^2_{\hat{\mu}_4,S_3}\}_{(h)}}{\chi^2_{k,\frac{h}{n}}}$$

with $\{d^2_{\hat{\mu}_4,S_3}\}_{(1)} \leq \{d^2_{\hat{\mu}_4,S_3}\}_{(2)} \leq \ldots \leq \{d^2_{\hat{\mu}_4,S_3}\}_{(n)}$. Let $d_i \;\; (i = 1, \ldots, n)$ be the robust distances of all observations with respect to $\hat{\boldsymbol{\mu}}_4$ and $c_1 S_3$, let w be a weight function and put $w_i = w(d_i)$ for all i. Then the center and scatter of the data are estimated by

$$\hat{\boldsymbol{\mu}}_5 = \frac{\sum_{i=1}^n w_i \tilde{\mathbf{x}}_i}{\sum_{i=1}^n w_i}$$

and

$$S_4 = \frac{\sum_{i=1}^n w_i (\tilde{\mathbf{x}}_i - \hat{\boldsymbol{\mu}}_5)(\tilde{\mathbf{x}}_i - \hat{\boldsymbol{\mu}}_5)'}{\sum_{i=1}^n w_i - 1}.$$

In our implementation we use 'hard rejection' by taking $w(d_i) = I(d_i \leq \sqrt{\chi^2_{k,0.975}})$ where I stands for the indicator function.

The spectral decomposition of S_4 can be written as $S_4 = P_2 L_2 P_2^t$ where the columns of $P_2 = P_{k,k}$ contain the eigenvectors of S_4 and $L_2 = L_{k,k}$ is the diagonal matrix with the corresponding eigenvalues. The final scores are now given by

$$T_{n,k} = (\tilde{X}_{n,k} - \mathbf{1}_n \hat{\boldsymbol{\mu}}_5') P_2. \tag{12}$$

4. The last step transforms the columns of P_2 back to $\mathbb{R}^p$, yielding the final robust principal components $P_{p,k}$. The final robust center $\hat{\boldsymbol{\mu}}$ is obtained by transforming $\hat{\boldsymbol{\mu}}_5$ back to $\mathbb{R}^p$, and the final p-dimensional robust scatter matrix S of rank k is given by (2). The scores (12) can be written as the equivalent formula (1) in $\mathbb{R}^p$. (Note that the robust score distance SD_i of (3) can be computed in the k-dimensional PCA space by the equivalent formula $SD_i = \sqrt{(\tilde{\mathbf{x}}_i - \hat{\boldsymbol{\mu}}_5)' S_4^{-1} (\tilde{\mathbf{x}}_i - \hat{\boldsymbol{\mu}}_5)}$ which saves computation time in high-dimensional applications.)

References

Boente, G., Pires, A.M., and Rodrigues, I. (2002), "Influence Functions and Outlier Detection under the Common Principal Components Model: A Robust Approach," *Biometrika,* 89, 861–875.

Box, G.E.P. (1954), "Some Theorems on Quadratic Forms Applied in the Study of Analysis of Variance Problems: Effect of Inequality of Variance in One-Way Classification," *The Annals of Mathematical Statistics*, 25, 290–302.

Campbell, N.A. (1980), "Robust Procedures in Multivariate Analysis I: Robust Covariance Estimation," *Applied Statistics,* 29, 231–237.

Croux, C., and Haesbroeck, G. (2000), "Principal Components Analysis based on Robust Estimators of the Covariance or Correlation Matrix: Influence Functions and Efficiencies," *Biometrika*, 87, 603-618.

Croux, C., and Ruiz-Gazen, A. (2000), "High Breakdown Estimators for Principal Components: the Projection-Pursuit Approach Revisited," under revision.

Davies, L. (1987), Asymptotic Behavior of S-estimators of Multivariate Location and Dispersion Matrices," *Annals of Statistics,* 15, 1269–1292.

Donoho, D.L. (1982), *Breakdown Properties of Multivariate Location Estimators,* Ph.D. Qualifying paper, Harvard University.

Egan, W.J., Morgan, S.L. (1998), "Outlier Detection in Multivariate Analytical Chemical Data," *Analytical Chemistry*, 70, 2372–2379.

Esbensen, K.H., Schönkopf, S., and Midtgaard, T. (1994), *Multivariate Analysis in Practice.* Camo, Trondheim.

Hubert, M., and Engelen, S. (2003), "Robust PCA and Classification in Biosciences," submitted.

Hubert, M., Rousseeuw, P.J., and Verboven, S. (2002), "A Fast Method for Robust Principal Components with Applications to Chemometrics," *Chemometrics and Intelligent Laboratory Systems*, 60, 101–111.

Hubert, M., and Vanden Branden, K. (2003), "Robust Methods for Partial Least Squares Regression," to appear in *Journal of Chemometrics.*

Hubert, M., and Van Driessen, K. (2003), "Fast and Robust Discriminant Analysis," to appear in *Computational Statistics and Data Analysis.*

Hubert, M., and Verboven, S. (2003), "A Robust PCR Method for High-Dimensional Regressors," *Journal of Chemometrics*, 17, 438–452.

Joliffe, I.T. (1986), *Principal Component Analysis* , New York, Springer.

Krzanowski, W.J. (1979), "Between-Groups Comparison of Principal Components," *Journal of the American Statistical Association,* 74, 703–707.

Lemberge, P., De Raedt, I., Janssens, K.H., Wei, F., and Van Espen, P.J. (2000), "Quantitative Z-Analysis of the 16–17th Century Archaelogical Glass Vessels using PLS Regression of EPXMA and μ-XRF Data," *Journal of Chemometrics*, 14, 751–763.

Li, G., and Chen, Z. (1985), "Projection-Pursuit Approach to Robust Dispersion Matrices and Principal Components: Primary Theory and Monte Carlo," *Journal of the American Statistical Association*, 80, 759–766.

Locantore, N., Marron, J.S., Simpson, D.G., Tripoli, N., Zhang, J.T., and Cohen, K.L. (1999), "Robust Principal Component Analysis for Functional Data," *Test*, 8, 1–73.

Maronna, R.A. (1976), "Robust M-estimators of Multivariate Location and Scatter," *Annals of Statistics,* 4, 51–67.

Nomikos, P., and MacGregor, J.F. (1995), "Multivariate SPC Charts for Monitoring Batch Processers," *Technometrics*, 37, 41–59.

Rocke, D.M., and Woodruff, D.L. (1996), "Identification of Outliers in Multivariate Data," *Journal of the American Statistical Association,* 91, 1047–1061.

Rousseeuw, P.J. (1984), "Least Median of Squares Regression," *Journal of the American Statistical Association,* 79, 871–880.

Rousseeuw, P.J, and Christmann, A. (2003), "Robustness against Separation and Outliers in Logistic Regression," *Computational Statistics and Data Analysis.*, 43, 315–332.

Rousseeuw, P.J., and Leroy, A. (1987), *Robust Regression and Outlier Detection,* New York: John Wiley.

Rousseeuw, P.J., and Van Driessen, K. (1999), "A Fast Algorithm for the Minimum Covariance Determinant Estimator," *Technometrics,* 41, 212–223.

Rousseeuw, P.J., and Van Zomeren, B.C. (1990), "Unmasking Multivariate Outliers and Leverage Points," *Journal of the American Statistical Association,* 85, 633–651.

Stahel, W.A. (1981), *Robust Estimation: Infinitesimal Optimality and Covariance Matrix Estimators,* Ph.D. thesis, ETH, Zürich.

Woodruff, D.L., and Rocke, D.M (1994), "Computable Robust Estimation of Multivariate Location and Shape in High Dimension using Compound Estimators," *Journal of the American Statistical Association,* 89, 888–896.

Wu, W., Massart, D.L., and de Jong, S. (1997), "The Kernel PCA Algorithms for Wide Data. Part I: Theory and Algorithms," *Chemometrics and Intelligent Laboratory Systems,* 36, 165–172.