TEST
DOI 10.1007/s11749-015-0464-0

ORIGINAL PAPER

A plug-in approach to sparse and robust principal component analysis

Luca Greco[1] · **Alessio Farcomeni**[2]

Received: 9 February 2015 / Accepted: 15 October 2015

Abstract We propose a method for sparse and robust principal component analysis. The methodology is structured in two steps: first, a robust estimate of the covariance matrix is obtained, then this estimate is plugged-in into an elastic-net regression which enforces sparseness. Our approach provides an intuitive, general and flexible extension of sparse principal component analysis to the robust setting. We also show how to implement the algorithm when the dimensionality exceeds the number of observations by adapting the approach to the use of robust loadings from ROBPCA. The proposed technique is seen to compare well for simulated and real datasets.

Keywords Dimension reduction · Elastic net · Explained variance · L_1 norm · MCD · MM · Outliers · ROBPCA · Trade-off curve

Mathematics Subject Classification 62F35 · 62H25

1 Introduction

Principal component analysis (PCA) is a widely used technique for descriptive multivariate statistics and dimensionality reduction (Jolliffe 2005). The main aim of PCA

Electronic supplementary material The online version of this article (doi:10.1007/s11749-015-0464-0) contains supplementary material, which is available to authorized users.

✉ Luca Greco
luca.greco@unisannio.it

Alessio Farcomeni
alessio.farcomeni@uniroma1.it

[1] DEMM Department, University of Sannio, P.zza Arechi II, 1, 82100 Benevento, Italy

[2] Department of Public Health and Infectious Diseases, Sapienza, University of Rome, Rome, Italy

Published online: 02 November 2015

is finding a lower dimensional representation of the data that simplifies the interpretation of the associations among variables and highlights the more relevant features. Dimensionality reduction is made at the price of some information loss, which is kept minimal by classical PCA. The $n \times p$ data matrix X is mapped to a new set of orthogonal and uncorrelated variables, collected in an $n \times q$ matrix Y_q^u, with $q \leq p$. The trace of the covariance matrix of Y_q^u is the largest one can attain. Here the superscript u stands for *unconstrained*. The *principal components* are linear combinations of the original variables as in

$$Y_j^u = a_j^{\mathsf{T}} \mathbf{X} = a_{1j} X_1 + a_{2j} X_2 + \cdots + a_{pj} X_p, \quad j = 1, 2, \ldots, q.$$

The coefficients of the linear combination are called *loadings*. Large loadings identify the most important variables. PCA does not necessarily deliver interpretable components and the strategy to ignore features associated with loadings that are small in absolute value may be misleading (Cadima and Jolliffe 1995). This has motivated *sparse* PCA, some of whose loadings are *exactly* zero. The interpretation and labeling of the components from sparse PCA is much simpler than that of classical PCA and compression and storing of the data is enhanced as some variables might be discarded. One of the first formal approaches to sparse PCA is SCoTLASS (Jolliffe et al. 2003). Here, we focus on the procedure introduced by Zou et al. (2006) and based on elastic-net regression, named sparse PCA (SPCA).

A well-known issue with PCA (and consequently with sparse PCA) is that outliers may spoil loading estimates, artificially inflate variances, mask the relevant features of the clean part of the data. This has motivated the development of several robust techniques; see Maronna et al. (2006), Varmuza and Filzmoser (2008), Heritier et al. (2009), Farcomeni and Ventura (2012) and Farcomeni and Greco (2015) for general reviews. One can distinguish two main approaches to robust PCA. The first is based on the eigen-decomposition of a robust estimate of the covariance matrix (Croux and Haesbroeck 2000; Salibian-Barrera et al. 2006). The second relies on projection pursuit (PP) based on a robust scale estimator (Croux and Ruiz-Gazen 2005; Croux et al. 2007). In a PP framework, principal components are found in sequence, without the need to estimate the covariance matrix. A related approach is ROBPCA (Hubert et al. 2005; Engelen et al. 2005), that combines projection pursuit ideas with robust estimation of the covariance matrix. In particular, PP and ROBPCA are well suited to handle high-dimensional data and situations in which the number of variables exceeds the number of observations, that is $p > n$. There are also other methods, like spherical PCA (Locantore et al. 1999) and orthogonal PCA (Maronna 2005), that will be not considered in the rest of this paper.

There are still very few methods for robust and sparse PCA. A notable exception is Croux et al. (2013), with a method based on projection pursuit (SPP). In this paper, we illustrate an alternative but complementary solution along the lines of plug-in principles (Hubert et al. 2008). We obtain *robust* loadings and enforce sparseness. With $n > p$, loadings can be usually obtained via eigen-decomposition of a robust estimate of covariance. In high dimensions or when $p > n$, loadings can be estimated through ROBPCA. In summary, we have two steps: when $n > p$, a robust estimate of the covariance matrix is obtained, then this estimate is plugged-in an elastic-net

regression leading to a sparse approximation of the robust loadings (Zou et al. 2006). We suggest to use the reweighted minimum Princiiance estimator (MCD) (Rousseeuw and Van Driessen 1999; Croux and Haesbroeck 2000) or the MM-estimator (Tatsuoka and Tyler 2000; Salibian-Barrera et al. 2006), whose properties and finite sample behavior are well known. In high dimensions or when $p > n$, it is not feasible to compute robust estimates of covariance. We proceed by using ROBPCA to estimate the first q robust loading vectors and corresponding eigenvalues, and, as a by product, a rank q robust estimate of the covariance matrix (Hubert et al. 2005). Then, this estimate is plugged into SPCA. Our method can be directly implemented as soon as a robust estimate (even not of full rank) of the covariance matrix is available. One referee kindly made us aware of Hubert et al. (2015), who obtain sparse and robust PCA via a modification of ROBPCA, named RObust Sparse PCA (ROSPCA). This approach shares some ideas with ours, but it is operationally different. ROSPCA is based on a first step identifying an outlier-free subset, then ScoTLASS is applied to the identified subset.

The paper is organized as follows: in Sect. 2 we give some background. In Sect. 3 we outline our robust plug-in sparse PCA. Numerical studies and real data applications are described in Sects. 4 and 5, respectively. Final remarks are given in Sect. 6.

2 Sparse and robust PCA

Sparse PCA can be obtained by introducing constraints on the loadings [e.g., on their L_1 or even L_0 norm as in Farcomeni (2009)]. One of the first formal approaches is SCoTLASS (Jolliffe et al. 2003): the variance of the jth component is maximized under an upper constraint on the sum of the absolute value of the loadings. The loadings l_j for the jth sparse component are, then,

$$l_j = \text{argmax}_{||a||=1} a^\mathsf{T} S a - \lambda_{1j}||a||_1, \quad a \perp l_1 \perp \cdots \perp l_{j-1} \tag{1}$$

where $||\cdot||_1$ denotes the L_1 norm, S is the sample variance–covariance matrix, λ_{1j} is a penalty parameter regulating the degree of sparsity of the jth component. Unconstrained PCA is obtained with $\lambda_{1j} = 0$.

A possibility for robust sparse PCA is projection pursuit (Croux et al. 2013). Robust PCA based on PP searches for directions that sequentially maximize a robust measure of variability $\hat{\sigma}^2(\cdot)$, where $\hat{\sigma}(\cdot)$ could be a scale M-estimate (Maronna 2005), the median absolute deviation (MAD), the Q_n estimator (Rousseeuw and Croux 1993), or the trimmed squared scale estimate (Rousseeuw and Leroy 1987). The advantage of robust PP is that one does not need estimating the covariance matrix. The jth sparse robust loading vector is given by

$$\ell_{\hat{\sigma};j} = \text{argmax}_{||a||=1} \hat{\sigma}^2(a^\mathsf{T} x_1, a^\mathsf{T} x_2, \ldots, a^\mathsf{T} x_n) - \lambda_{1j}||a||_1, \tag{2}$$

requiring that $a \perp \ell_1 \perp \cdots \perp \ell_{j-1}$. Eigenvalues are then computed as $v_{\hat{\sigma};j} = \hat{\sigma}^2(\ell_{\hat{\sigma};j}^\mathsf{T} x_1, \ell_{\hat{\sigma};j}^\mathsf{T} x_2, \ldots, \ell_{\hat{\sigma};j}^\mathsf{T} x_n)$. Guidelines for choosing λ_{1j} can be found in Leng and Wang (2009), Farcomeni (2009), Guo et al. (2010) and Croux et al. (2013). Even if, in general, we may want to select a different sparsity parameter for each component,

a simple shortcut, aimed at reducing the computational burden, is to set $\lambda_{1j} = \lambda, \forall j$. One may select λ by minimizing the criterion proposed by Croux et al. (2013), which is a robust counterpart of the proposal of Guo et al. (2010). Let L_q^s and L_q^u be the loading matrices from sparse and unconstrained (robust) PCA containing the first q PC directions, respectively. Let $R^s = X - XL^s(L^s)^\mathsf{T}$ and $R^u = X - XL^u(L^u)^\mathsf{T}$ be the corresponding residual matrices, whose jth columns are $r_j^s = (r_{1j}^s, r_{2j}^s, \ldots, r_{nj}^s)^\mathsf{T}$ and $r_j^u = (r_{1j}^u, r_{2j}^u, \ldots, r_{nj}^u)^\mathsf{T}$, respectively. A robust criterion is given by

$$Q_{\text{CFF}}(\lambda) = \frac{\sum_{j=1}^{q} \hat{\sigma}^2\left(r_j^s\right)}{\sum_{j=1}^{q} \sigma^2\left(\hat{r}_j^u\right)} + m(\lambda)\frac{\log n}{n}\,, \tag{3}$$

where $m(\lambda)$ denotes the cardinality (that is the number of non-zero loadings) of L_q^s. When we are interested in the selection of a different λ_{1j} for each component, we can select λ_{1j} by maximizing the trade-off product optimization as proposed in Croux et al. (2013), that is given by the explained variance multiplied by the number of zero loadings of the jth component, i.e.,

$$Q_{\text{TPO}}(\lambda_j) = \hat{\sigma}^2\left(Y_j^s\right)\left[p - m(\lambda_{1j})\right]\,. \tag{4}$$

Moreover, we could also try to select the same λ for each component by defining

$$Q_{\text{TPO}}(\lambda) = \sum_{j=1}^{q} \hat{\sigma}^2\left(Y_j^s\right)\left[qp - m(\lambda)\right]\,. \tag{5}$$

Another open issue is how to choose the number of components q. When PCA is based on the eigen-decomposition of a robust estimate of the covariance matrix, the explained robust variance is typically measured by the ratio of the sum of its first q larger eigenvalues to the sum of all of them. When robust eigenvectors and eigenvalues are obtained according to the PP approach, the percentage of explained robust variance is given by $\frac{\sum_{j=1}^{q} v_{\hat{\sigma};j}}{\sum_{j=1}^{p} \hat{\sigma}^2(X_j)}$. This strategy is naturally extended to the sparse setting. The percentage of robust variance explained by the first q sparse and robust components is

$$\text{EV}_q^{r_1} = \frac{\sum_{j=1}^{q} \hat{\sigma}^2\left(Y_j^s\right)}{\sum_{j=1}^{p} \hat{\sigma}^2(X_j)}\,. \tag{6}$$

An alternative route is to to measure the total robust variance based on unconstrained robust components

$$\text{EV}_q^{r_2} = \frac{\sum_{j=1}^{q} \hat{\sigma}^2\left(Y_j^s\right)}{\sum_{j=1}^{p} \hat{\sigma}^2\left(Y_j^u\right)}\,. \tag{7}$$

When $\hat{\sigma}^2$ is the sample variance, (6) and (7) coincide.

3 Robust plug-in sparse PCA

PCA is commonly performed through eigen-decomposition of the sample covariance matrix. It is shown in Zou et al. (2006) that an equivalent formulation can be obtained by defining a ridge regression problem. Let $A = [a_1, \ldots, a_q]$ and $B = [b_1, \ldots, b_q]$ be $p \times q$ matrices. Without loss of generality, we assume X is zero centered. The first q principal components are the solution to

$$\arg\min_{A,B} \sum_{i=1}^{n} ||x_i - AB^{\mathsf{T}} x_i||^2 + \lambda_2 \sum_{j=1}^{q} ||b_j||^2, \quad \lambda_2 > 0 \tag{8}$$

subject to $A^{\mathsf{T}} A = \mathrm{I}_q$, where I_q is the identity matrix of size q. The elements of B solving (8) are proportional to the loadings. As in (1) and (2), a sparse approximation of the loadings is obtained by adding an L_1 penalty into (8). Sparse loadings are indeed proportional to the solution of

$$\mathrm{argmin}_{A,B} \sum_{i=1}^{n} ||x_i - AB^{\mathsf{T}} x_i||^2 + \lambda_2 \sum_{j=1}^{q} ||b_j||^2 + \sum_{j=1}^{q} \lambda_{1j} ||b_j||_1 \,. \tag{9}$$

Problem (9) is usually referred to as the elastic net (Zou and Hastie 2005). See also Leng and Wang (2009) and Guo et al. (2010) for further improvements and Witten et al. (2009) for a more general approach. In Zou et al. (2006) it is shown that the elastic net problem (9) only depends on the original data matrix via the sample covariance matrix S and one can minimize

$$\sum_{j=1}^{q} (a_j - b_j)^{\mathsf{T}} S (a_j - b_j) + \lambda_2 \sum_{j=1}^{q} ||b_j||^2 + \sum_{j=1}^{q} \lambda_{1,j} ||b_j||_1. \tag{10}$$

The optimization problem in (10) yields the SPCA solution, that is clearly non-robust. A robust SPCA algorithm can be obtained by plugging-in a robust estimate $\hat{\Sigma}$ of the covariancematrix in (10). Now the data are centered according to a robust measure of location. Our proposal is to minimize

$$\sum_{j=1}^{q} (a_j - b_j)^{\mathsf{T}} \hat{\Sigma} (a_j - b_j) + \lambda_2 \sum_{j=1}^{q} ||b_j||^2 + \sum_{j=1}^{q} \lambda_{1,j} ||b_j||_1. \tag{11}$$

This yields sparse and robust components quite naturally. We call this solution SRPCA. The optimization problem (11) is solved similarly to Zou et al. (2006) through an alternating least squares algorithm, whose general iteration is given in Algorithm 1. In practice, λ_2 is fixed as a small positive number, and only the parameters $\lambda_{1,j}$ are selected according to the chosen criterion. It can be shown that for any finite λ_2 there

exist a sequence of $\lambda_{1,j}$ leading to the same solutions (Zou et al. 2006). The vectors a_j, $j = 1, 2, \ldots, q$ can be initialized to the leading eigenvectors of $\hat{\Sigma}$. To investigate the stability of the algorithm we randomly perturbed the leading eigenvectors of $\hat{\Sigma}$ by adding noise uniform over the interval $(-\epsilon, \epsilon)$ in our simulated and real data examples. We did not find any remarkable change in the final solution for reasonable values of ϵ.

Algorithm 1 SRPCA

Update B
for $j = 1, \ldots, q$ **do**
$b_j = \text{argmin}_b (a_j - b)^\mathsf{T} \hat{\Sigma} (a_j - b) + \lambda_2 ||b||^2 + \lambda_{1j} ||b||_1$
end for
Update A
Solve $\hat{\Sigma} B = U D V^\mathsf{T}$.
Let $\hat{A} = U V^\mathsf{T}$

Algorithm 1 is feasible as long as we are able to compute a robust estimate $\hat{\Sigma}$. A major concern is that popular and effective robust estimates such as the MM or the reweighted MCD are limited to moderate dimensions ($p \leq 100$ in some cases) and to $n > p$. One could in other cases plug-in the spatial sign covariance matrix. This approach has been also suggested by Croux et al. (2013), who use SCoTLASS instead of SPCA, but as they also note a robust solution is *not* obtained. In order to obtain a sparse and robust PCA in the challenging $p > n$ situation and/or in high dimensions, we suggest to base the first step on an unconstrained robust PCA well suited to handle $p > n$ and/or high-dimensional data, such as ROBPCA (Hubert et al. 2005). ROBPCA yields a rank q robust estimate of the covariance matrix $\hat{\Sigma}_q = L_q^u M_q \left(L_q^u\right)^\mathsf{T}$, where $M_q = \text{diag}(v_j)$, $j = 1, 2, \ldots, q$ is the diagonal matrix of eigenvalues. In the $p > n$ situation and high dimensions, Zou et al. (2006) suggested to modify Algorithm 1 by letting $\lambda_2 \to \infty$. This reduces the computational burden and corresponds to *soft-thresholding* the loadings, where the threshold is given by a function of $\lambda_{1,j}$. In our method, the estimate $\hat{\Sigma}_q$ is plugged into SPCA based on soft-thresholding, leading to a robust plug-in solution. This corresponds to *s*parsifing the subspace spanned by the first q robust loadings found by ROBPCA. The resulting method is outlined in Algorithm 2, that is a special case of Algorithm 1. The soft-thresholding operator is

$$ST(y, \delta) = (|y| - \delta)_+ \text{sign}(y) = \begin{cases} y - \delta & \text{if } y > \delta, y > 0 \\ 0 & \text{if } |y| < \delta \\ y + \delta & \text{if } y < -\delta, y < 0 \end{cases}$$

and is related to coordinate descent (Friedman et al. 2007).

Algorithms 1 and 2 are currently implemented into the `R` functions `spca` and `arrayspc` in package `elasticnet`, respectively, for non-robust cases. SRPCA is obtained by using a robust estimate of covariance as input in `spca`. SRPCA by soft-thresholding is slightly more cumbersome. We provide our code as supplementary material with this paper.

Algorithm 2 SRPCA by soft-thresholding

Update B
for $j = 1, \ldots, q$ **do**
$b_j = ST\left(a_j^\mathsf{T} \hat{\Sigma}_q, \frac{\lambda_{1j}}{2}\right)$
end for
Update A
Solve $\hat{\Sigma}_q B = UDV^\mathsf{T}$.
Let $\hat{A} = UV^\mathsf{T}$

The sparse components based on (9) and their robust counterparts driven by (11) are (usually mildly) correlated. As a consequence, the explained (robust) variance accounted for by the first q components will not correspond to the sum of their variances. In the non-robust setting, Zou et al. (2006) suggested to use the *adjusted variance*. Let u_j be the vector of residuals of the linear regression where the jth sparse component is the response and the previous ones are predictors. The total variance explained by the first q components is $\sum_{j=1}^{q} \mathrm{Var}(u_j)$. The explained robust variance can be estimated similarly. We suggest to obtain the residuals u_j with robust regression, such as least trimmed squares (LTS) (Rousseeuw 1984; Rousseeuw and Leroy 1987; Rousseeuw and Van Driessen 1999; Pison et al. 2002) or MM regression (Salibian-Barrera and Yohai 2006; Maronna et al. 2006) and, then, estimate the adjusted robust variance as $\sum_{j=1}^{q} \hat{\sigma}^2(u_j)$. See Farcomeni and Greco (2015) for more details on this.

4 Numerical studies

In order to illustrate the finite sample behavior of SRPCA we use two synthetic examples. SRPCA will be based on both the MCD and the MM-estimators when $n > p$, whereas it will rely on ROBPCA when $p > n$. In the first example we study the distribution of the maximal angle between the fitted and true subspaces, whereas in the second one we mainly focus on the distribution of the loadings and the percentage of explained robust variance.

4.1 Example 1

The first example is along the lines of Croux et al. (2013). The true sparse structured loading matrix is assumed to be

$$L = \begin{pmatrix} \sqrt{0.5} & 0 & \sqrt{0.5} & 0 & 0 & \ldots & 0 \\ \sqrt{0.5} & 0 & -\sqrt{0.5} & 0 & 0 & \ldots & 0 \\ 0 & \sqrt{0.5} & 0 & \sqrt{0.5} & 0 & \ldots & 0 \\ 0 & \sqrt{0.5} & 0 & -\sqrt{0.5} & 0 & \ldots & 0 \\ 0 & 0 & 0 & 0 & 1 & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & 0 & 0 & 0 & 0 & \ldots & 1 \end{pmatrix}$$

of dimension $p = 10, 200$, and the eigenvalues are $v = (1, 0.5, 0.1, \ldots, 0.1), n = 50$. Data are generated from an ϵ-contaminated multivariate normal distribution $(1 - \epsilon)N_p(0, LVL^T) + \epsilon H$, $V = \text{diag}(v_j)$, with outliers generated from H, that is assumed to be p-variate Normal $N\left(\xi, \mathrm{I}_p\right)$ with mean vector ξ obtained by stacking $\frac{p}{10}$ times the vector $(2, 4, 2, 4, 0, -1, 1, 0, 1, -1)$. The results are averaged over 100 replicates. We consider clean data ($\epsilon = 0$) and four different contamination rates, ranging from $\epsilon = 10\,\%$ to $\epsilon = 40\,\%$. Figures 1 and 2 show the average value of $2\phi/\pi$ (called average deviation), where ϕ is the maximal angle between the subspaces spanned by the first two columns of L and the resulting first two components from SRPCA as a function of the penalty parameter λ. The maximal angle is scaled to lie in $[0, 1]$. SRPCA is based on both the 50 % breakdown point MCD estimator and the 50 % breakdown point and 95 % scale efficient MM-estimator for $n > p$, whereas it is based on the first two eigenvectors from ROBPCA with 50 % breakdown point when $p > n$. The results have the expected pattern, similarly to the SPP methodology in Croux et al. (2013). In absence of contamination, when the underlying model is sparse, the sparse techniques improve over their unconstrained counterparts. The average deviation decreases until a minimum is reached and then it starts to increase as a function of the penalty parameter. Lack of efficiency of SRPCA under the assumed model is tolerable, indeed. Under contamination, SRPCA still gives accurate results, whereas outliers break down SPCA. Table 1 gives the median (with MAD) of the minimum values $2\phi/\pi$ obtained in each trial. SRPCA leads to more accurate results than SPP based on the Q_n estimator of scale in all scenarios considered. The plug-in approach based on ROBPCA works properly leading to suitable and robust results when $p > n$, whereas the plug-in approach discussed in Croux et al. (2013) did not lead to a robust solution.

The computing time is reasonable. Figure 3 gives the CPU time on an Intel Core i7 at 2.4 GHz for both SRPCA and SPP when $\lambda = \lambda^*$, where λ^* is the value for which the minimum average deviation for non-contaminated data was obtained. When $p = 10$ SRPCA based on Algorithm 1 is slightly more time consuming than SPP on median and it exhibits a noticeably larger variability. On the other hand, SRPCA based on soft-thresholding leads to a substantial decrease in computational time. When $p = 200$, SRPCA combined with soft-thresholding, exhibits a noticeably lower computational burden than SPP, whose CPU time is about ten times larger. Table 2 gives the mean CPU time (in seconds) for $n = 50, 100, 500$ and $p = 50, 100, 500, 1000$ over 100 trials for SRPCA based on ROBPCA and soft-thresholding and SPP based on Q_n, for $q = 2$ and $\lambda = 1$. Here and in the following we used the function `SPcaGrid` from package `rrcovHD`.

According to the same data generating scheme, we also performed a further numerical investigation. For each sample, we computed the value $2\phi/\pi$ driven by SPCA, MM-SRPCA and MCD-SRPCA from both Algorithms 1 and 2 (the three latter ones here are denoted as SPCA2, MM-SRPCA2 and MCD-SRPCA2), by selecting the parameter λ according to the criterion (3) based on the Q_n. We also included ROSPCA (with 50 % breakdown point) and SPP based on Q_n for comparison purposes. From Fig. 4 we notice that SRPCA always has a low bias, well comparable with that of SPCA when contamination does not occur, and lower than that of SPP. A similar behavior is observed for ROSPCA. Furthermore, both Algorithms 1 and 2 exhibit close performances even if Algorithm 2 leads to a slightly larger variability in the maximal angles.

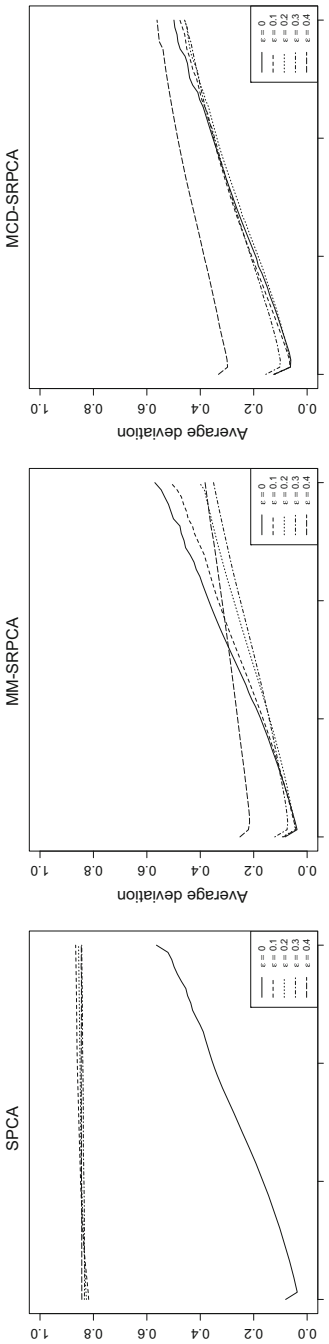

Fig. 1 Example 1. Average deviation between estimated and true loadings from SPCA (*left*) and SRPCA based on MM-estimation (*middle*) and the MCD (*right*) when $p = 10$

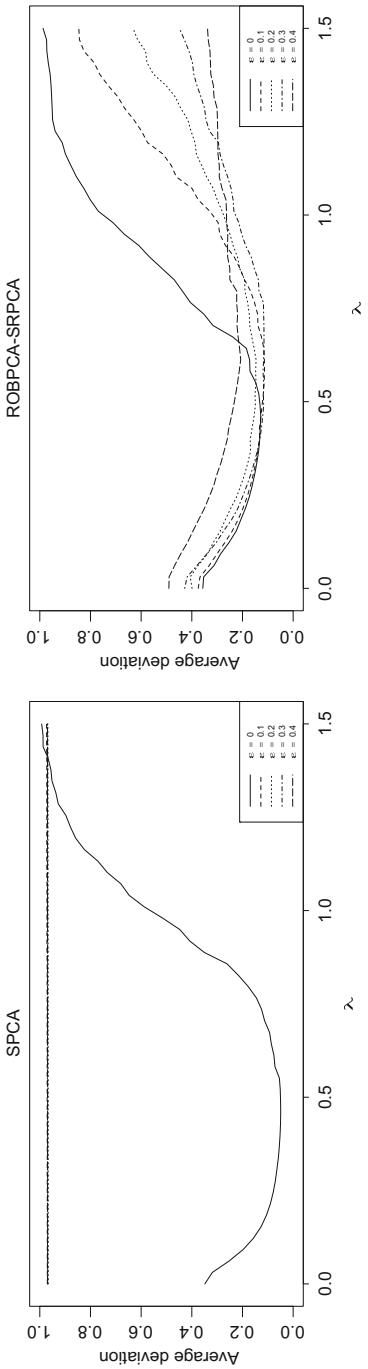

Fig. 2 Example 1. Average deviation between estimated and true loadings from SPCA (*left*) and SRPCA based on ROBPCA (*right*) when $p = 200$

Table 1 Example 1

ϵ %	$p = 10$				$p = 200$		
	SPCA	MCD-SRPCA	MM-SRPCA	Q_n-SPP	SPCA	ROBPCA-SRPCA	Q_n-SPP
0	0.030 (0.014)	0.048 (0.028)	0.032 (0.016)	0.076 (0.037)	0.033 (0.016)	0.043 (0.025)	0.151 (0.035)
10	0.838 (0.142)	0.051 (0.035)	0.033 (0.022)	0.072 (0.039)	0.972 (0.024)	0.038 (0.024)	0.152 (0.037)
20	0.834 (0.140)	0.047 (0.031)	0.036 (0.019)	0.071 (0.039)	0.968 (0.022)	0.046 (0.032)	0.152 (0.040)
30	0.816 (0.124)	0.048 (0.032)	0.034 (0.023)	0.078 (0.027)	0.970 (0.025)	0.047 (0.034)	0.153 (0.041)
40	0.823 (0.139)	0.065 (0.074)	0.052 (0.049)	0.093 (0.047)	0.971 (0.023)	0.057 (0.043)	0.159 (0.057)

Median (with MAD) of the minimum values of $2\phi/\pi$ from SPCA, SRPCA and SPP, for $n = 50$, $\epsilon = 0, 0.1, 0.2, 0.3, 0.4$, for $p = 10, 200$

Figure 5 shows the distribution of the selected sparsity parameter λ for SRPCA and SRPCA2 both in the non-contaminated and contaminated scenarios. The $p > n$ case is illustrated in Fig. 6. Here SRPCA is based on the first two ROBPCA loadings and Algorithm 2 has been used. As before, we observe a good behavior of SRPCA both when no contamination occurs and in the presence of outliers. In particular, SRPCA seems to be more accurate than ROSPCA, at the cost of a slightly larger variability.

4.2 Example 2

In our second example, we consider the same simulation scheme of Guo et al. (2010) and Farcomeni (2009), augmented with the outlier generating process in Croux et al. (2013). The proposed design is characterized by three hidden factors $V_1 \sim N(0, 290)$, $V_2 \sim N(0, 300)$ and $V_3 = -0.3V_1 + 0.925V_2 + \epsilon$, where $\epsilon \sim N(0, 1)$ and V_1, V_2 and ϵ are independent. Then, p variables are obtained according to the following scheme:

$$X_j = \begin{cases} V_1 + \epsilon_j, & 1 \leq j \leq \frac{4p}{10} \\ V_2 + \epsilon_j, & \frac{4p}{10} + 1 \leq j \leq \frac{8p}{10} \\ V_3 + \epsilon_j, & \frac{8p}{10} + 1 \leq j \leq p \end{cases}$$

We set $n = 20$, $p = 10, 40$ and $\epsilon = 0, 10$ %. We use the 25 % breakdown point MCD estimator and the 50 % breakdown point and 95 % scale efficient MM-estimator for $n > p$, and ROBPCA with 25 % breakdown point for $p > n$. The numerical studies are based on 100 replicates. By construction, an ideal sparse representation of the loadings should consist of two PCs of cardinality $\frac{6p}{10}$ and $\frac{4p}{10}$, respectively. The first block of variables from X_1 to $X_{\frac{4p}{10}}$ should have high loadings on the second component and zero loadings on the first one. The second block of variables from $X_{\frac{4p}{10}+1}$ to $X_{\frac{8p}{10}}$ is expected to have high loadings on the first component and zero loadings on the second one. The remaining variables in the third block should have larger loadings on the first component and a sparse approach should shrink them toward zero on the second component. Let us consider the case $n = 20$ and $p = 10$ first. Table 3 reports the median (with MAD) of the loadings of the first two components obtained with

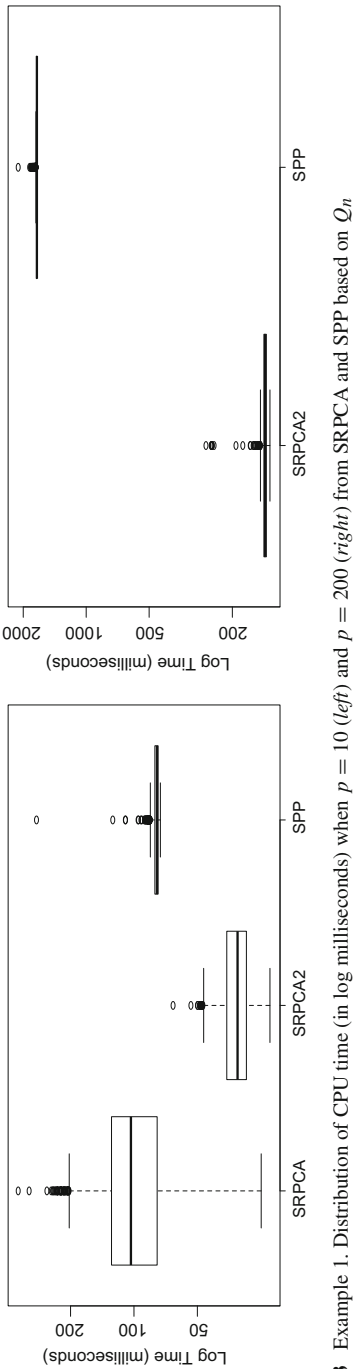

Fig. 3 Example 1. Distribution of CPU time (in log milliseconds) when $p = 10$ (*left*) and $p = 200$ (*right*) from SRPCA and SPP based on Q_n

Table 2 Example 1

Mean CPU time (in seconds) needed for SRPCA with soft-thresholding (bold) and SPP

n/p	50	100	500	1000
50	**0.155**	**0.163**	**0.554**	**3.589**
	0.506	0.595	6.506	29.866
100	**0.261**	**0.289**	**1.184**	**15.085**
	1.234	3.569	14.856	61.233
500	**1.248**	**1.886**	**47.927**	**57.620**
	16.907	18.274	131.344	219.578

SPCA and SRPCA, along with the median percentage of explained adjusted robust variance, evaluated according to (7). We use Q_n as robust measure of scale and LTS as regression scheme for decorrelating the components. For each sample, we searched for the same penalty parameter on both components according to the criterion (5) based on the Q_n. Similar results were obtained using (3). When the data are not contaminated, all methods are able to recover the ideal sparse representation. The slightly larger variability that characterizes the robust estimates is tolerable. When outliers occur, SPCA is not able anymore to recover the true sparse structure, whereas SRPCA still leads to reliable results. In addition to Table 3 and Fig. 7 shows the boxplots concerning the distribution of the loadings (in absolute value, to handle the indeterminacy in the sign of the loadings) of the first two components of SPCA and SRPCA.

Now we investigate the $p > n$ situation. We compare SRPCA with soft-thresholding based on ROBPCA with ROSPCA and SPP based on the Q_n estimator. The results are summarized in Table 4, where we report the median (with MAD) of the loadings in each of the three blocks of variables. The median percentage of explained adjusted robust variance is given in the last line. SRPCA estimates successfully the expected loadings' pattern for both clean and contaminated data. The estimated loadings are close to those of ROSPCA, and have a smaller variability than those obtained SPP. SRPCA also allows to explain a larger rate of total robust variability. Figure 8 shows the distribution of the loadings (in absolute value) on the first two components based on SPCA, ROBPCA-SRPCA, ROSPCA and SPP.

We conclude by noting that a larger contamination rate may yield numerical instability in the MCD. If we use a contamination rate 20 %, when $n = 20$, $p = 10$, the MCD is unfeasible given the small sample size with respect to the number of dimensions. The procedure based on MM is still reliable. Figure 9 shows loadings of MM-SRPCA for a contamination rate $\epsilon = 20$ %. If we increase the sample size there are no problems also with the MCD.

5 Real data applications

We now consider two real data applications. The first example concerns a dataset with $n > p$, the second one deals with the $p > n$ case.

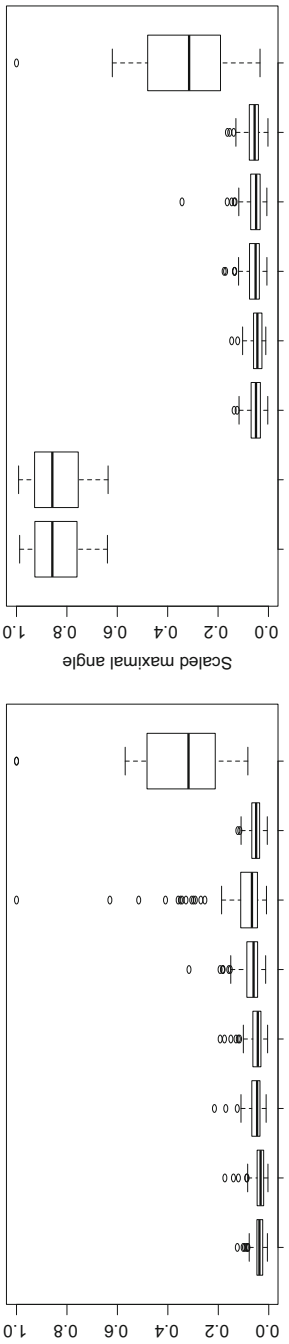

Fig. 4 Example 1. Distribution of the scaled maximal angle $2\phi/\pi$ by using SPCA (*a*), SPCA2 (*b*), MM-SRPCA (*c*), MM-SRPCA2 (*d*), MCD-SRPCA (*e*), MCD-SRPCA2 (*f*), ROSPCA (*g*) and Q_n-SPP (*h*), for $n = 50$, $p = 10$, $q = 2$, $\epsilon = 0$ (*left*), $\epsilon = 0.20$ (*right*)

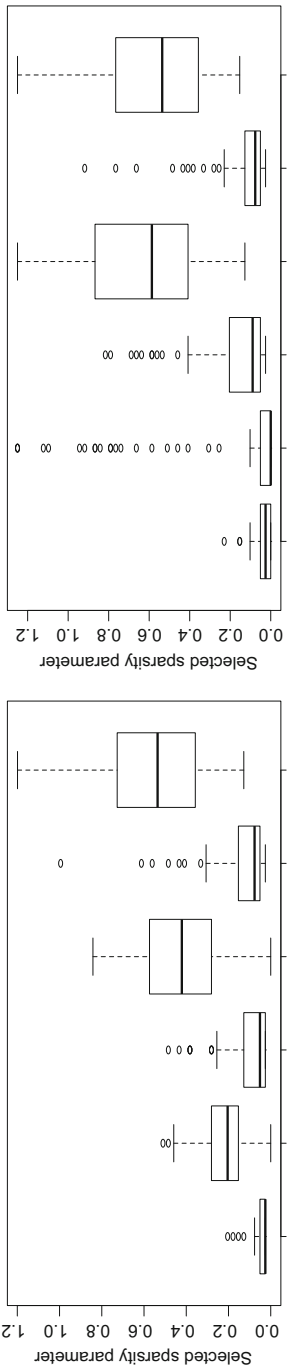

Fig. 5 Example 1. Distribution of the selected λ according to criterion (3) based on Q_n, for SPCA (*a*), SPCA2 (*b*), MM-SRPCA (*c*), MM-SRPCA2 (*d*), MCD-SRPCA (*e*), MCD-SRPCA2 (*f*), $n = 50$, $p = 10$, $q = 2$, $\epsilon = 0$ (*left*), $\epsilon = 0.20$ (*right*)

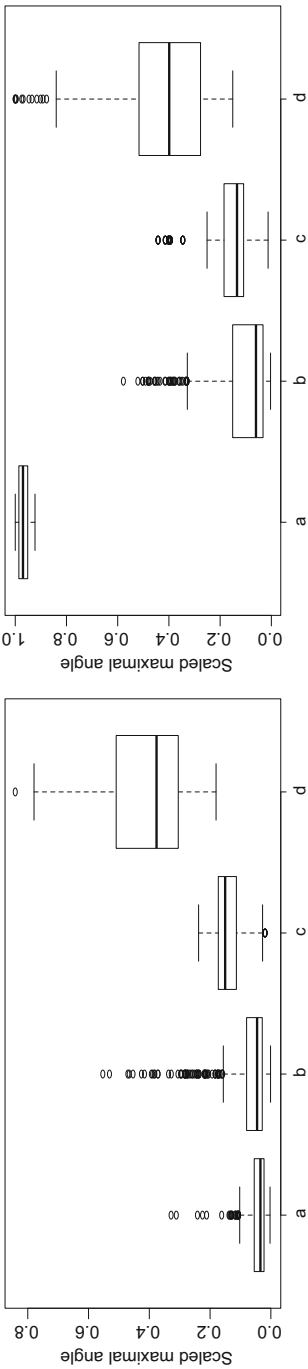

Fig. 6 Example 1. Distribution of the scaled maximal angle $2\phi/\pi$ by using SPCA (*a*), ROBPCA-SRPCA (*b*), ROSPCA (*c*) and Q_n-SPP (*d*), for $n = 50$, $p = 200$, $q = 2$, $\epsilon = 0$ (*left*), $\epsilon = 0.20$ (*right*). Algorithm 2 has been used

Table 3 Example 2

	SPCA		MCD-SRPCA		MM-SRPCA	
	PC1	PC2	PC1	PC2	PC1	PC2
$\epsilon = 0\ \%$						
Block 1						
X_1	0 (0)	0.49 (0.07)	0 (0)	0.49 (0.11)	0(0)	0.49 (0.12)
X_2	0 (0)	0.50 (0.08)	0 (0)	0.50 (0.09)	0 (0)	0.50 (0.13)
X_3	0 (0)	0.48 (0.07)	0 (0)	0.49 (0.10)	0 (0)	0.48 (0.13)
X_4	0 (0)	0.51 (0.07)	0 (0)	0.50 (0.10)	0 (0)	0.49 (0.13)
Block 2						
X_5	0.42 (0.04)	0 (0)	0.42 (0.06)	0 (0)	0.41 (0.09)	0 (0)
X_6	0.41 (0.04)	0 (0)	0.41 (0.06)	0 (0)	0.40 (0.07)	0 (0)
X_7	0.41 (0.04)	0 (0)	0.42 (0.05)	0 (0)	0.41 (0.10)	0 (0)
X_8	0.41 (0.04)	0 (0)	0.41 (0.06)	0 (0)	0.39 (0.10)	0 (0)
Block 3						
X_9	0.38 (0.04)	0 (0)	0.38 (0.06)	0 (0)	0.37 (0.08)	0 (0)
X_{10}	0.38 (0.05)	0 (0)	0.38 (0.06)	0 (0)	0.37 (0.09)	0 (0)
EV_2^{r2}	0.59 (0.10)	0.33 (0.08)	0.59 (0.13)	0.32 (0.12)	0.54 (0.14)	0.31 (0.11)
$\epsilon = 10\ \%$						
Block 1						
X_1	0 (0)	0 (0)	0 (0)	0.45 (0.13)	0 (0)	0.44 (0.18)
X_2	0.12 (0.08)	0 (0)	0 (0)	0.46 (0.13)	0 (0)	0.46 (0.15)
X_3	0.51 (0.32)	0 (0)	0 (0)	0.47 (0.14)	0 (0)	0.45 (0.16)
X_4	0 (0)	0 (0)	0 (0)	0.48 (0.14)	0 (0)	0.49 (0.18)
Block 2						
X_5	0 (0)	0 (0)	0.40 (0.11)	0 (0)	0.42 (0.17)	0 (0)
X_6	0.51 (0.26)	0 (0)	0.40 (0.09)	0 (0)	0.37 (0.15)	0 (0)
X_7	0.31 (0.24)	0.22 (0.33)	0.40 (0.08)	0 (0)	0.40 (0.16)	0 (0)
X_8	0 (0)	0 (0)	0.38 (0.12)	0 (0)	0.39 (0.13)	0 (0)
Block 3						
X_9	0 (0)	0.17 (0.26)	0.36 (0.11)	0 (0)	0.39 (0.12)	0 (0)
X_{10}	0 (0)	0 (0)	0.37 (0.10)	0 (0)	0.37 (0.16)	0 (0)
EV_2^{r2}	0.37 (0.16)	0.17 (0.07)	0.58 (0.14)	0.30 (0.14)	0.61 (0.16)	0.30 (0.13)

Median (with MAD) of the loadings on the first two components using SPCA and SRPCA based on the MCD and the MM-estimator of covariance, with $n = 20$, $p = 10$. The last line gives the median (with MAD) percentage of explained adjusted robust variance based on the Q_n estimate of scale

5.1 The car data

The first dataset we analyze concerns $n = 195$ cars and $p = 14$ variables containing technical and insurance-related data. The example has also been considered in Croux et al. (2013). The data include 20 cars equipped with a diesel engine, that could be

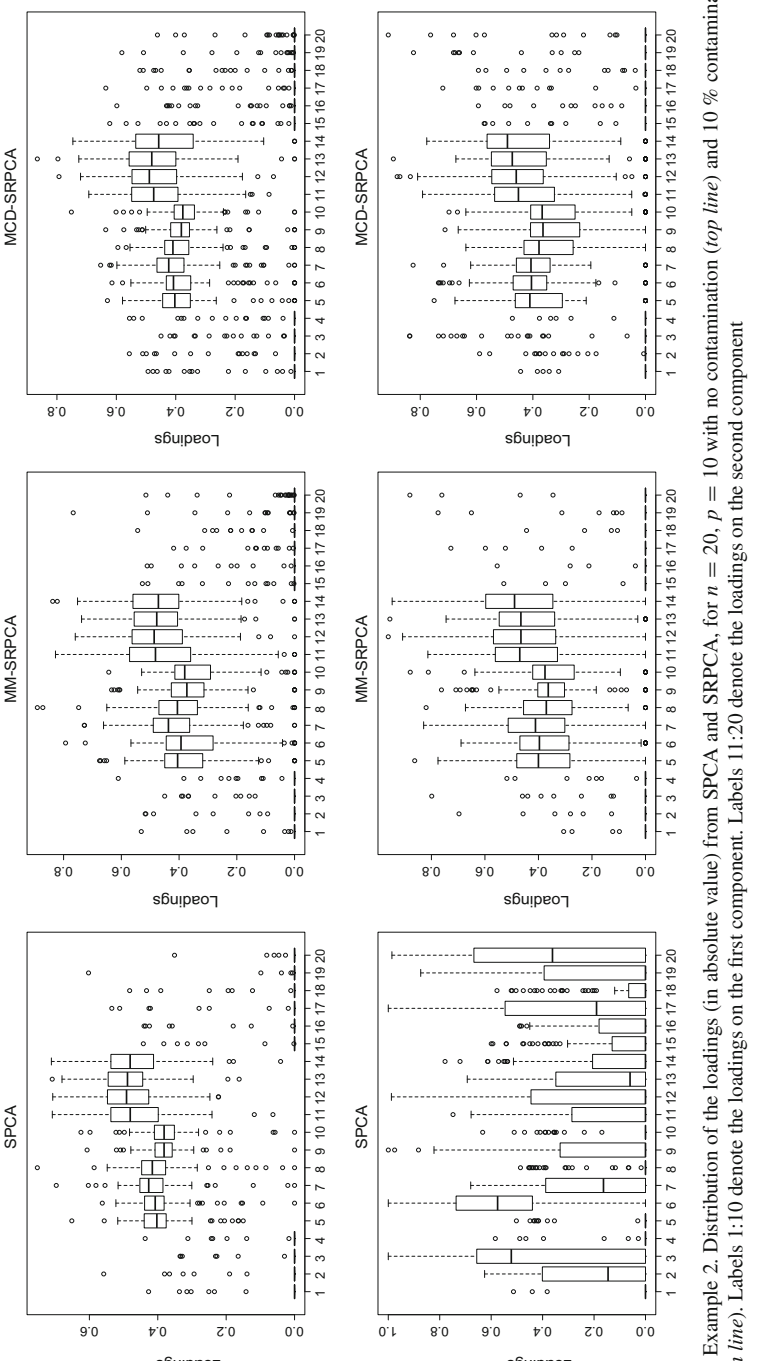

Fig. 7 Example 2. Distribution of the loadings (in absolute value) from SPCA and SRPCA, for $n = 20$, $p = 10$ with no contamination (*top line*) and 10 % contamination (*bottom line*). Labels 1:10 denote the loadings on the first component. Labels 11:20 denote the loadings on the second component

Table 4 Example 2

	SPCA		SRPCA		SPP		ROSPCA	
	PC1	PC2	PC1	PC2	PC1	PC2	PC1	PC2
$\epsilon = 0$ %								
Block 1	0 (0)	0.25 (0.01)	0 (0)	0.25 (0.02)	0 (0)	0.23 (0.05)	0 (0)	0.24 (0.02)
Block 2	0.20 (0.01)	0 (0)	0.20 (0.03)	0 (0)	0.19 (0.04)	0 (0)	0.21 (0.02)	0 (0)
Block 3	0.20 (0.02)	0 (0)	0.19 (0.05)	0 (0)	0.18 (0.05)	0 (0)	0.19 (0.04)	0 (0)
EV_2^{r2}	0.58 (0.01)	0.38 (0.07)	0.57 (0.11)	0.37 (0.08)	0.56 (0.16)	0.31 (0.11)	0.59 (0.17)	0.35 (0.16)
$\epsilon = 10$ %								
Block 1	0.10 (0.14)	0 (0)	0 (0)	0.22 (0.05)	0 (0)	0.17 (0.14)	0 (0)	0.24 (0.02)
Block 2	0.16 (0.09)	0.06 (0.09)	0.20 (0.06)	0 (0)	0.18 (0.08)	0 (0)	0.21 (0.04)	0 (0)
Block 3	0.16 (0.09)	0.02 (0.03)	0.16 (0.08)	0 (0)	0.16 (0.08)	0 (0)	0.18 (0.05)	0 (0)
EV_2^{r2}	0.34 (0.08)	0.20 (0.07)	0.55 (0.12)	0.36 (0.12)	0.54 (0.13)	0.32 (0.11)	0.58 (0.15)	0.29 (0.13)

Median (with MAD) of the loadings in each block of variables with $n = 20$, $p = 40$ and $\epsilon = 0, 10$ % from SPCA, ROBPCA-SRPCA, SPP based on Q_n and ROSPCA. The last line gives the median percentage (with MAD) of explained (adjusted) robust variance based on the Q_n estimate of scale

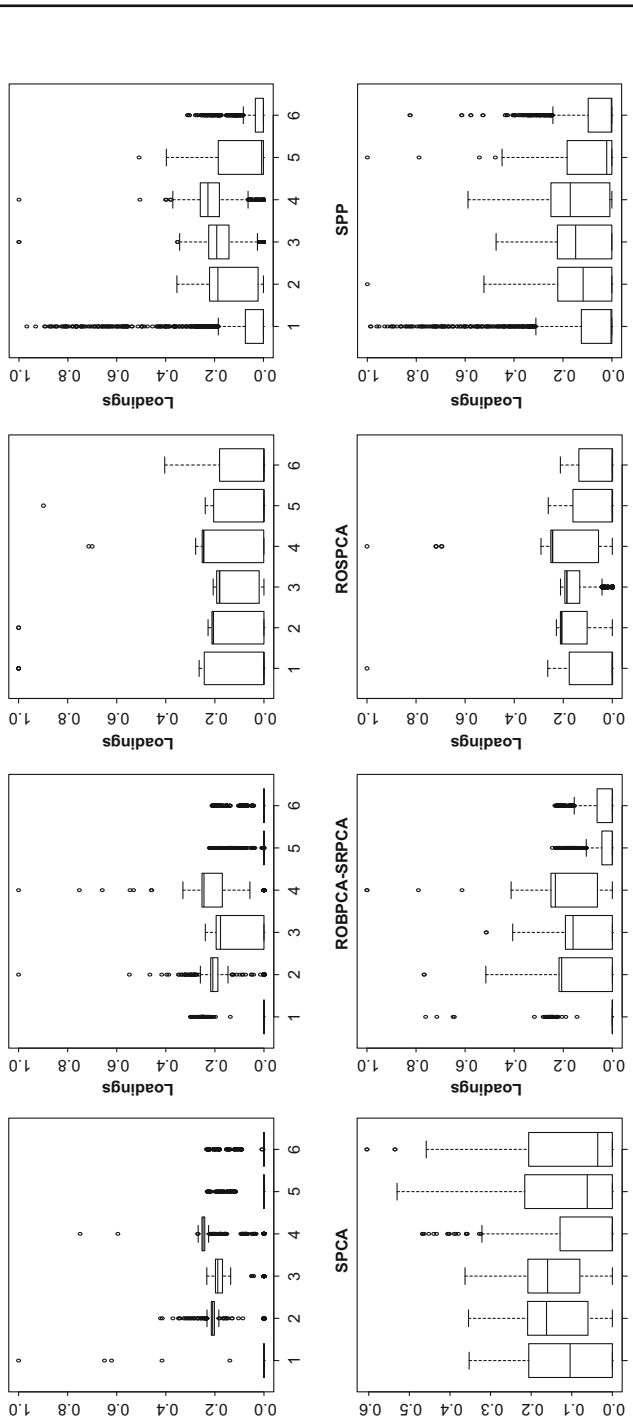

Fig. 8 Example 2. Distribution of the loadings (in absolute value) in each block from SPCA, SRPCA, ROSPCA and SPP for $n = 20$, $p = 40$ with no contamination (*top line*) and 10 % contamination (*bottom line*). Labels 1:3 denote the loadings in the three blocks on the first component. Labels 4:6 denote the loadings in the three blocks on the second component

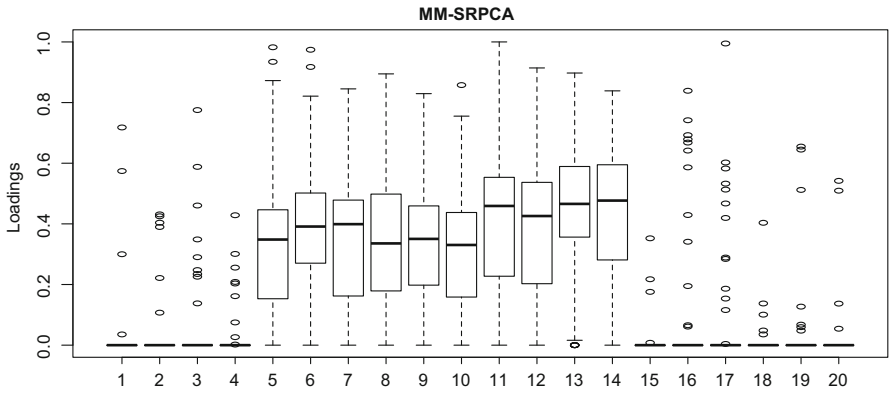

Fig. 9 Example 2. Loadings (in absolute value) from MM-SRPCA, $n = 20$, $p = 10$, $\epsilon = 20$ %. Labels 1:10 denote the loadings of the first component. Labels 11:20 denote the loadings of the second component

considered as outliers compared to the majority of cars running on gasoline. In particular, cars running on diesel exhibit larger compression ratios than those running on gasoline. In the following, we perform PCA and sparse PCA by omitting the information concerning the type of engine. The data have been pre-processed by standardizing each column by the Q_n estimate of scale. We compare the results of SRPCA based on both the 25 % breakdown point MCD and the 95 % shape efficient MM estimate of the covariance matrix. We also included in the analysis SPP based on Q_n. We retain the first three components, which account for about 75 % of the total robust variance, evaluated as in the denominator of (7), regardless of the robust estimator used. Figure 10 shows (3), based on Q_n, over a grid of 50 values. It leads to select $\lambda = 0.214$ when both for the reweighed MCD and the MM-estimator. Tables 5 and 6 report the unconstrained and sparse robust loadings based on the two plug-in approaches. The last lines give the percentage of explained adjusted robust variance based on Q_n, computed according to (6) and (7), respectively, and obtained by using LTS regression. Entries in parenthesis give the unadjusted explained robust variance. The occurrence of zero loadings clearly helps in interpretation. The first sparse component contrasts some characteristics of the vehicle and of the engine (positive sign) with fuel efficiency (negative sign). The price appears to be related to the former set of variables. The second component is mainly dominated by `height` and `compression ratio`, the third by `stroke` and `peak-rpm`. By introducing sparseness into robust PCA based on the reweighed MCD and the MM-estimator of covariance, the cardinality of the loadings matrix is reduced from 42 to 17. The percentage of explained robust variance only drops slightly at the selected penalty parameter. Figure 11 shows the proportion of cumulative explained adjusted variance as a function of λ. This is the so called trade-off curve proposed in Croux et al. (2013). The selected penalty parameter is located right before the curve drops, therefore we conclude that this is an acceptable value and the trade-off between sparsity and information loss is tolerable.

Table 7 also gives the output from SPP based on the Q_n. In the constrained setting, (3) lead us to fix $\lambda = 1.65$. The loadings from SPP are characterized by the same sparse configuration for the first component, but the results for the second and third are rather different. In particular, it turns out that the second component is dom-

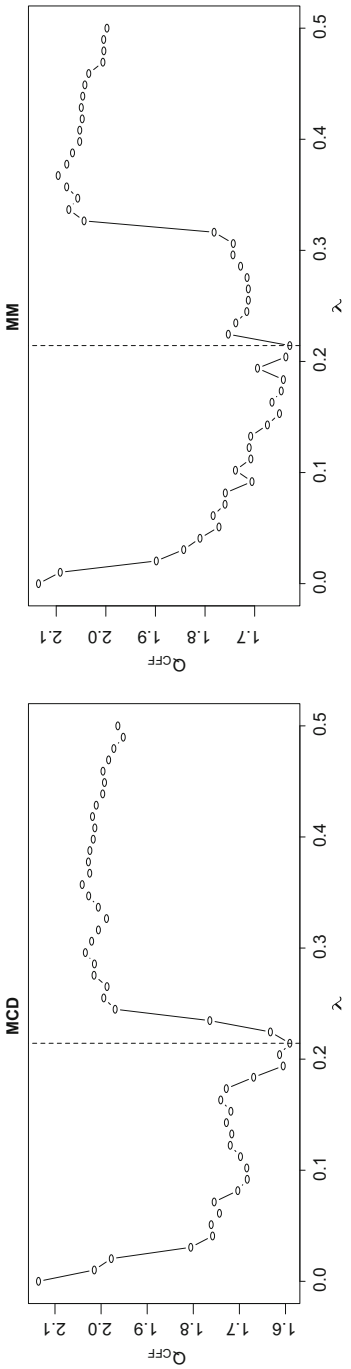

Fig. 10 Car data. Selection of λ for SRPCA based on Q_{CFF}. The *dotted lines* give the selected sparsity parameter

Table 5 Car data

	Unconstrained MCD			Sparse MCD		
	PC1	PC2	PC3	PC1	PC2	PC3
Symboling	0.05	0.19	0.13	0	0.07	0
Wheel-base	−0.33	−0.19	−0.25	0.39	−0.28	0
Length	−0.33	−0.12	−0.12	0.39	0	0
Width	−0.31	−0.02	−0.11	0.23	0	0
Height	−0.15	−0.41	−0.31	0	−0.63	0
Curb-weight	−0.32	0.02	−0.02	0.32	0	0
Bore	−0.31	−0.20	0.29	0.25	0	−0.16
Stroke	−0.04	0.61	−0.31	0	0	0.81
Compression-ratio	0.16	−0.42	−0.46	−0.17	−0.72	0
Horsepower	−0.35	0.22	0.01	0.40	0	0
Peak-rpm	0.07	0.28	−0.63	0	0	0.57
City-mpg	0.30	−0.13	−0.07	−0.22	0	0
Highway-mpg	0.31	−0.12	−0.07	−0.36	0	0
Price	−0.36	0.01	−0.08	0.33	0	0
$\%EV^{r_1}$	68.22	16.85	12.31	65.41	7.14 (7.75)	6.37 (7.40)
$\%EV^{r_2}$	52.98	13.08	9.56	50.80	5.55 (6.02)	4.95 (5.75)

Loadings on the first $q = 3$ non-sparse and sparse components from MCD estimation. The last lines give the percentage of explained adjusted variance. The rate of unadjusted variance is given in parenthesis

inated by `peak-rpm` and the only other non-null loading is the one corresponding to `symboling`. The third component is entirely determined by `stroke`. SPP leads to a more sparse solution than SRPCA, but at the cost of a slight reduction in the percentage of explained robust variance. According to (7), the percentage of explained robust variance is 62.57 % for MCD-SRPCA and 63.18 % for MM-SRPCA but only 55.71 % for Q_n-SPP.

Robust PCA can be used as a tool for reliable outlier detection by using appropriate outlier maps (e.g., Hubert et al. 2005; Cerioli and Farcomeni 2011; Farcomeni and Greco 2015). Outlier maps are obtained by plotting the score distance (SD) and the orthogonal distance (OD) for each observation. Anomalous observations are characterized by large score or orthogonal distances. Figure 12 indicates that the sparsity constraints do not affect the diagnostic power of the robust methodologies. Outlier maps from both MCD- and MM-SRPCA detect the group of outliers in the right top corner, corresponding to the cars running on diesel. The 20 cars running on diesel exhibit both large distances and can be classified as *bad leverages*.The solid lines give the cut-off values aimed at detecting outliers (see Hubert et al. 2005; Farcomeni and Greco 2015 for more details).

5.2 Octane data

The Octane data consist of $n = 39$ gasoline samples. For each gasoline sample the near-infrared absorbance spectrum over $p = 226$ wavelengths is measured, hence

Table 6 Car data

	Unconstrained MM			Sparse MM		
	PC1	PC2	PC3	PC1	PC2	PC3
Symboling	0.05	0.19	0.13	0	0.01	0
Wheel-base	−0.32	−0.21	−0.25	0.24	−0.53	0
Length	−0.33	−0.14	−0.12	0.34	0	0
Width	−0.33	−0.03	−0.13	0.32	0	0
Height	−0.14	−0.42	−0.33	0	−0.66	0
Curb-weight	−0.32	0.02	0	0.30	0	0
bore	−0.29	−0.21	0.33	0.21	0	−0.27
Stroke	−0.05	0.60	−0.33	0	0	0.80
Compression-ratio	0.16	−0.40	−0.43	−0.24	−0.53	0
Horsepower	−0.34	0.23	0.05	0.44	0	0
Peak-rpm	0.05	0.28	−0.60	0	0	0.93
City-mpg	0.30	−0.13	−0.08	−0.25	0	0
Highway-mpg	0.31	−0.12	−0.07	−0.37	0	0
Price	−0.37	0.03	−0.09	0.36	0	0
$\%EV^{r_1}$	68.67	17.10	12.29	67.60	8.36 (13.76)	6.70 (7.48)
$\%EV^{r_2}$	52.49	13.07	9.39	51.67	6.39 (10.52)	5.12 (5.72)

Loadings on the first $q = 3$ non-sparse and sparse components from MM-estimation. The last lines give the percentage of explained adjusted variance. The rate of unadjusted variance is given in parenthesis

$p \gg n$. A sparse representation of the loadings may be useful for identifying relevant spectral ranges. Six samples contain added alcohol and are outliers. Their spectra clearly deviate from the others as they present larger values after the 145th wavelength. This example has been discussed in Hubert et al. (2005).

Here, a sparse structure of the loadings has been recovered by using SRPCA based on the first two leading loadings vectors from ROBPCA (with 25 % breakdown point) and Algorithm 2. A different sparsity parameter λ_{1j}, $j = 1, 2$ has been selected for each component by maximizing the criterion (4) based on the MAD, as shown in Fig. 13, over a grid of 50 values. In order to run Algorithm 2 separately for each component, we proceeded as follows: first we applied Algorithm 2 only for the first component over a grid of values for λ_{11}; then we run it again but by looking for two components and imposing no restrictions on the first but only on the second one by varying λ_{12} on a different grid of 50 points.

The unconstrained and the sparse robust loadings are given in Fig. 14. Sparseness enhances the interpretation of the two components, since relevant spectral ranges are now clearly identified. The cardinality of the first sparse component is 90, wheres 57 is the number of non-null loadings for the second one. It is worth noting that the two sparse components have exact zero loadings in wavelengths on the right end of the spectrum, that is the region of the spectrum in which the six outlying samples exhibit anomalous large values. At the selected sparsity parameters, the percentage of explained (unadjusted) robust variance evaluated by using MAD for the first two

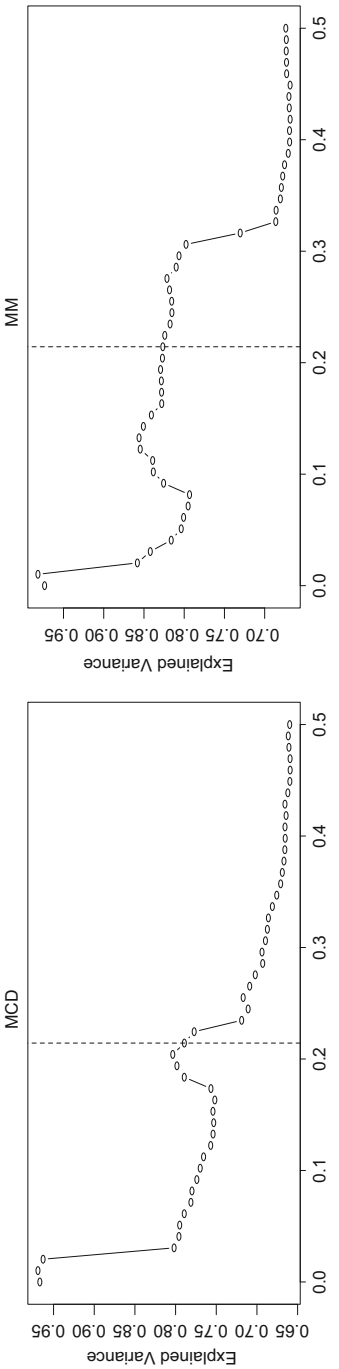

Fig. 11 Car data. Cumulative explained robust adjusted variance from sRPCA based on the reweighed MCD (*left*) and the MM estimator (*right*) with $q = 3$. The *dotted line* gives the selected sparsity parameter λ

Table 7 Car data

	Unconstrained PP			Sparse PP		
	PC1	PC2	PC3	PC1	PC2	PC3
Symboling	0.03	−0.10	0.25	0	0.12	0
Wheel-base	−0.32	0.16	−0.20	−0.05	0	0
Length	−0.30	0.21	0.01	−0.26	0	0
Width	−0.32	0.10	0.09	−0.22	0	0
Height	−0.13	0.16	−0.29	0	0	0
Curb-weight	−0.29	0.13	0.07	−0.27	0	0
Bore	−0.13	0.37	−0.02	−0.27	0	0
Stroke	−0.01	−0.26	0.49	0	0	1.00
Compression−ratio	0.47	0.57	0.52	0.44	0	0
Horsepower	−0.31	0.00	0.31	−0.41	0	0
Peak-rpm	−0.15	−0.53	0.29	0	0.99	0
City-mpg	0.28	−0.01	−0.13	0.30	0	0
Highway-mpg	0.27	−0.06	−0.14	0.33	0	0
Price	−0.32	0.23	0.28	−0.42	0	0
$\%EV^{r_1}$	83.22	28.69	17.81	80.01	8.90	7.13
$\%EV^{r_2}$	48.28	16.64	10.33	46.41	5.16	4.14

Loadings on the first $q = 3$ non-sparse and sparse components from SPP based on Q_n. The last lines give the percentage of explained adjusted variance

components is 86.77 and 29.40, respectively. Figure 15 shows the trade-off curve for each component based on (7): parameters λ_{11} and λ_{12} are well before the sharpest decline of the curve. After adjusting with LTS regression, the explained variance for the second components drops to 9.73 and the cumulative explained variance is 96.50. SPP based on the MAD leads to a close number of non-null loadings on the first two components, that is 130, but accounting for only about 80 % of the total robust variance. The total variance has been computed on the first two components from the corresponding unconstrained analyses, respectively, according to (7), with $p = q = 2$. The criterion (6) here is not appropriate, since in both cases the robust variance of the components is larger than the total robust variance $\sum_{j=1}^{p} \hat{\sigma}^2(X_j)$. From Fig. 16 we can see that enforcing sparseness does not affect the diagnostic power of the robust procedure. The outlier map based on $q = 2$ is still able to flag the six outlying samples, as does unconstrained ROBPCA. As one referee pointed out, the score distances of the six outliers became smaller in the sparse analysis. Actually, SRPCA gives many zero loadings to those wavelengths in which the six outlying samples exhibit anomalous larger values than the other clean samples. Hence, their scores on both the first and the second sparse component decrease in absolute value with respect to their unconstrained counterparts and the same happens to the corresponding score distances.

In this example, the CPU time for obtaining the SRPCA solution (including ROBPCA and the search for the sparseness parameters) was 0.461 seconds. The time

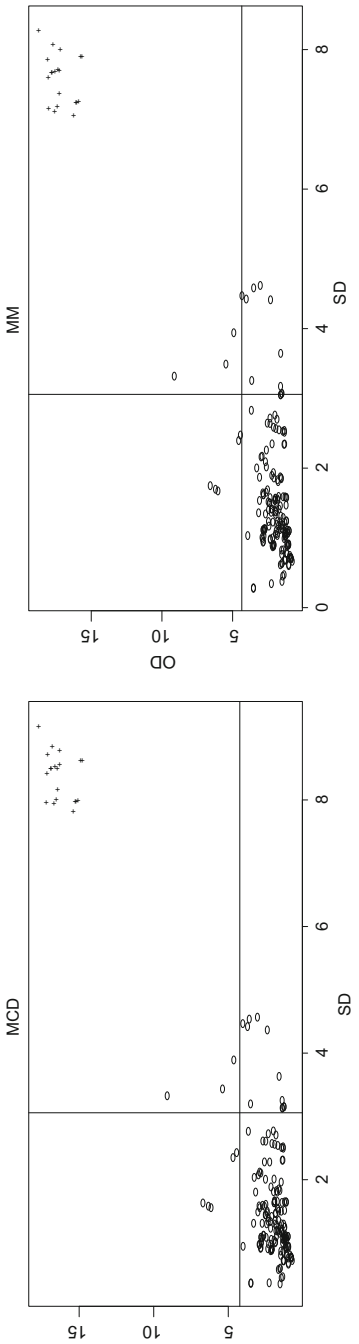

Fig. 12 Car data. Outlier map from MCD-SRPCA (*left*) and MM-SRPCA (*right*). Cars running on diesel are denoted by +

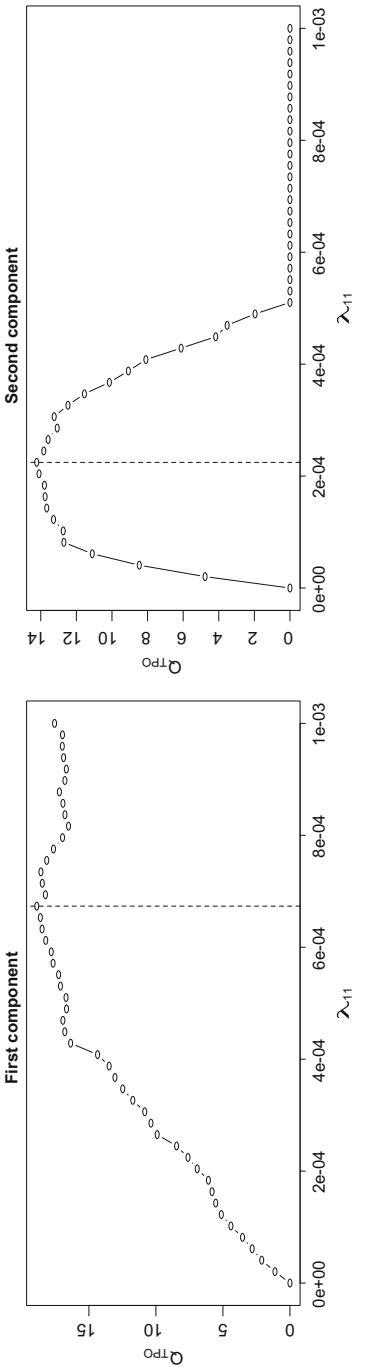

Fig. 13 Octane data. Selection of $\lambda_{1j}, j = 1, 2$ based on Q_{TPO}. The *dotted lines* give the selected parameters, respectively

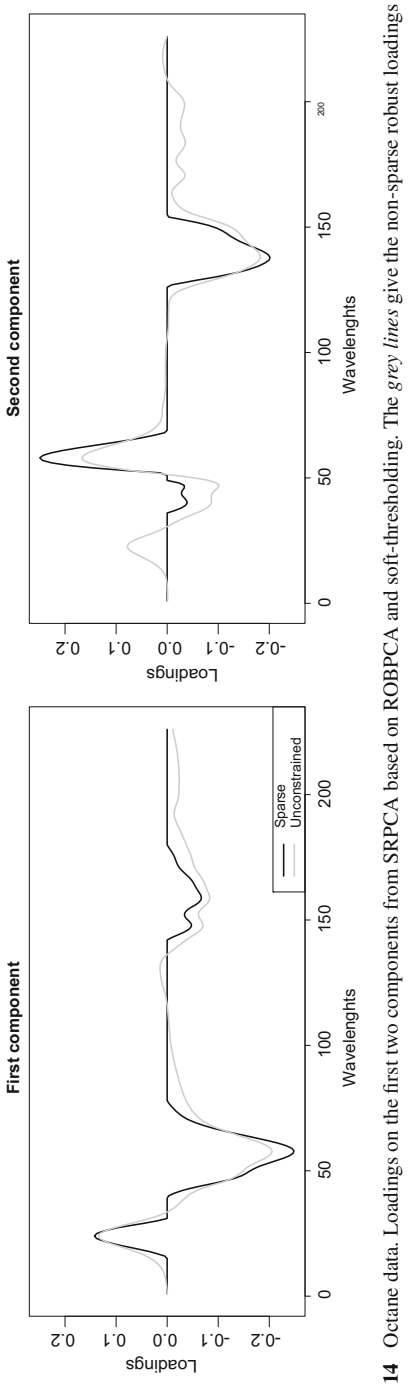

Fig. 14 Octane data. Loadings on the first two components from SRPCA based on ROBPCA and soft-thresholding. The *grey lines* give the non-sparse robust loadings

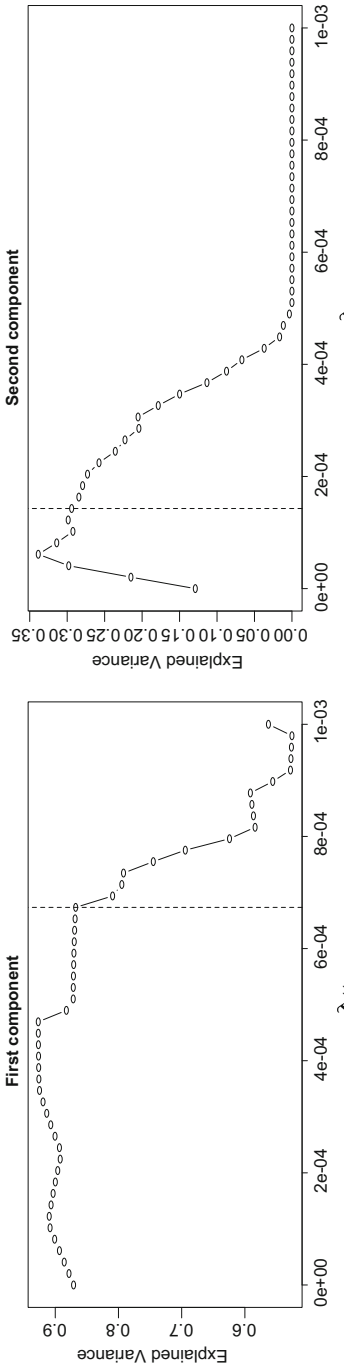

Fig. 15 Octane data. Explained robust variance for the first $q = 2$ components from sRPCA based on ROBPCA. The *dotted lines* give the selected sparsity parameter λ_{1j}, $j = 1, 2$, respectively

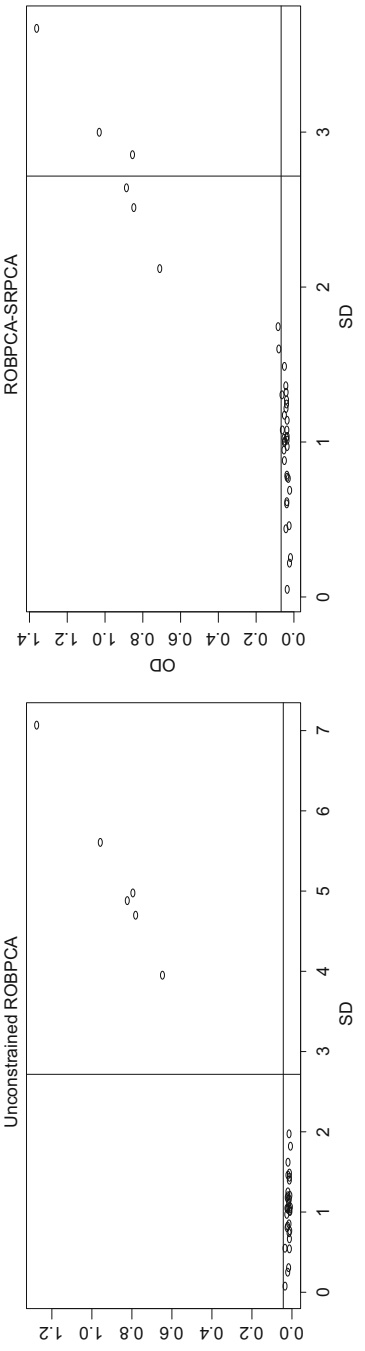

Fig. 16 Octane data. Outlier map from unconstrained ROBPCA (*left*) and ROBPCA-SRPCA (*right*), with $q = 2$

required by SPP was 0.161 seconds, but the comparison is not fair because the latter method is based on optimized R code whereas the former is not.

6 Final remarks

We proposed a robust technique for sparse PCA that has been thought as a direct extension of robust PCA based on robust covariance estimation, but it also performs well in high dimensions and when $p > n$. SRPCA is based on combining a robust estimate (possibly not full rank) of the covariance matrix and SPCA. In principle any robust PCA method could be used to this end. In our experience not all methods give satisfactory solutions. For instance, spherical PCA does not seem to be a good choice. For our illustration, for reasons of space, we have focused only on few (good) options. SRPCA compares well with existing methods, such as sparse and robust projection pursuit and ROSPCA. In conclusion, SRPCA has been seen to be accurate and to lead to an acceptable trade-off between sparseness and efficiency loss. The use of computationally efficient algorithms, available through `R` functions, allows us to readily obtain solutions in reasonable time even in high-dimensional problems.

Acknowledgments The authors want to thank two anonymous reviewers whose stimulating comments were helpful in improving this work and the understanding of the problem. The authors are also grateful to Professor Mia Hubert who kindly shared the `R` code for ROSPCA.

References

Cadima J, Jolliffe I (1995) Loading and correlations in the interpretation of principal components. J Appl Stat 22(2):203–214

Cerioli A, Farcomeni A (2011) Error rates for multivariate outlier detection. Comput Stat Data Anal 55:544–553

Croux C, Haesbroeck G (2000) Principal component analysis based on robust estimators of the covariance or correlation matrix: influence functions and efficiencies. Biometrika 87(3):603–618

Croux C, Ruiz-Gazen A (2005) High breakdown estimators for principal components: the projection-pursuit approach revisited. J Multivar Anal 95(1):206–226

Croux C, Filzmoser P, Oliveira MR (2007) Algorithms for projection-pursuit robust principal component analysis. Chemometr Intell Lab 87(2):218–225

Croux C, Filzmoser P, Fritz H (2013) Robust sparse principal component analysis. Technometrics 55(2):202–214

Engelen S, Hubert M, Branden K (2005) A comparison of three procedures for robust PCA in high dimensions. Aust J Stat 34:117–126

Farcomeni A (2009) An exact approach to sparse principal component analysis. Comput Stat 24(4):583–604

Farcomeni A, Ventura L (2012) An overview of robust methods in medical research. Stat Med Res 21:111–133

Farcomeni A, Greco L (2015) Robust methods for data reduction. Chapman & Hall/CRC Press, Boca Raton

Friedman J, Hastie T, Höfling H, Tibshirani R et al (2007) Pathwise coordinate optimization. Ann Appl Stat 1(2):302–332

Guo J, James G, Levina E, Michailidis G, Zhu J (2010) Principal component analysis with sparse fused loadings. J Comput Graph Stat 19(4):930–946

Heritier S, Cantoni E, Copt S, Victoria-Feser MP (2009) Robust methods in biostatistics. Wiley, Chichester

Hubert M, Rousseeuw P, Branden K (2005) ROBPCA: a new approach to robust principal component analysis. Technometrics 47(1):64–79

Hubert M, Rousseeuw P, Van Aelst S (2008) High-breakdown robust multivariate methods. Stat Sci 23:92–119

Hubert M, Reynkens T, Schmitt E, Verdonck T (2015) Sparse PCA for high-dimensional data with outliers. Technometrics (to appear)
Jolliffe I (2005) Principal component analysis. Wiley Online Library, New York
Jolliffe I, Trendafilov N, Uddin M (2003) A modified principal component technique based on the LASSO. J Comput Graph Stat 12(3):531–547
Leng C, Wang H (2009) On general adaptive sparse principal component analysis. J Comput Graph Stat 18(1):201–215
Locantore N, Marron J, Simpson D, Tripoli N, Zhang J, Cohen K (1999) Robust principal component analysis for functional data. TEST 8(1):1–73
Maronna R (2005) Principal components and orthogonal regression based on robust scales. Technometrics 47(3):264–273
Maronna RA, Martin RD, Yohai VJ (2006) Robust statistics: theory and methods. Wiley, New York
Pison G, Van Aelst S, Willems G (2002) Small sample corrections for LTS and MCD. Metrika 55:111–123
Rousseeuw PJ (1984) Least median of squares regression. J Am Stat Assoc 79:851–857
Rousseeuw P, Croux C (1993) Alternatives to the median absolute deviation. J Am Statis Assoc 88(424):1273–1283
Rousseeuw P, Leroy A (1987) Robust regression and outlier detection. Wiley-Interscience, New York
Rousseeuw P, Van Driessen K (1999) A fast algorithm for the minimum covariance determinant estimator. Technometrics 41:212–223
Salibian-Barrera M, Yohai VJ (2006) A fast algorithm for S-regression estimates. J Comput Graph Stat 15:414–427
Salibian-Barrera M, Van Aelst S, Willems G (2006) Principal components analysis based on multivariate MM estimators with fast and robust bootstrap. J Am Stat Assoc 101(475):1198–1211
Tatsuoka K, Tyler D (2000) On the uniqueness of S-functionals and M-functionals under nonelliptical distributions. Ann Statist 28(4):1219–1243
Varmuza K, Filzmoser P (2008) Introduction to multivariate statistical analysis in chemometrics. Chapman & Hall/CRC Press, Boca Raton
Witten DM, Tibshirani R, Hastie T (2009) A penalized matrix decomposition, with applications to sparse principal components and canonical correlation analysis. Biostat 10(3):515–534
Zou H, Hastie T (2005) Regularization and variable selection via the elastic net. J R Stat Soc Ser B 67:301–320
Zou H, Hastie T, Tibshirani R (2006) Sparse principal component analysis. J Comput Graph Stat 15(2):265–286