Supplementary material for Sparse Principal Component Analysis Based on Least Trimmed Squares

Yixin Wang and Stefan Van Aelst*
Department of Mathematics, KU Leuven, Belgium

September 16, 2019

Abstract

This supplementary material provides more background and additional empirical results for the sparse principal component analysis method. Some results regarding the estimation of the center are provided. Conditions on the eigenvalues are stated for successful recovery of the sparse principal components. The additional simulation results include results concerning outlier detection performance. The analysis of the escalator video is discussed in more detail. A toy example illustrating the advantages of sequential estimation of the principal components is given. Finally, the performance of the method is evaluated when estimating a five-dimensional PC subspace.

Keywords: dimension reduction, penalization, robustness

*The authors gratefully acknowledge the support of the International Funds KU Leuven grant C16/15/068 and COST Action IC1408 CRoNoS.

Estimation of the center

We compare the performance of (reweighted) LTS-SPCA when updating or not updating the estimate of the center in the iterations. When the center is not updated, we center the data with the estimate of $\boldsymbol{\mu}$ given by LTS-PCA before running Algorithm 2 in the manuscript and remove step 11 from the algorithm. From the results in Table S.1 it can be seen that updating the center estimate leads to slightly better results in most cases.

Table S.1: The average angles between estimated and true PC subspace given by (reweighted) LTS-SPCA when updating or not updating the center estimate for the data generated from setting 3 ($\eta = 25$) with $n = 200$, $p = 20$ ($b = 4$) or 500 ($b = 20$) and $\varepsilon = 0\%$, 20% or 40%.

p	update $\boldsymbol{\mu}$	ε (%)	LTS-SPCA	LTS-SPCA-RW2	LTS-SPCA-RW5
20	yes	0	0.0878	0.0560	0.0573
		20	0.0837	0.0665	0.0649
		40	0.0706	0.0787	0.0698
	no	0	0.0905	0.0556	0.0551
		20	0.0846	0.0706	0.0686
		40	0.0833	0.0975	0.1016
500	yes	0	0.0760	0.0616	0.0580
		20	0.0755	0.0683	0.0661
		40	0.0732	0.0786	0.0771
	no	0	0.0760	0.0615	0.0590
		20	0.0781	0.0683	0.0658
		40	0.0933	0.0783	0.0812

Conditions on the eigenvalues

Let us define the largest and smallest s-sparse eigenvalue for any matrix $A \in \mathbb{R}^{p\times p}$ by

$$\lambda_{\max}(A, s) = \max_{\boldsymbol{v}\in\mathbb{R}^p, \|\boldsymbol{v}\|_2=1} \boldsymbol{v}^{\mathrm{T}} \boldsymbol{A}\boldsymbol{v} \qquad s.t. \qquad \|\boldsymbol{v}\|_0 \leqslant s, \tag{S.1}$$

and

$$\lambda_{\min}(A, s) = \min_{\boldsymbol{v}\in\mathbb{R}^p, \|\boldsymbol{v}\|_2=1} \boldsymbol{v}^{\mathrm{T}} \boldsymbol{A}\boldsymbol{v} \qquad s.t. \qquad \|\boldsymbol{v}\|_0 \leqslant s, \tag{S.2}$$

respectively. Now, define $\rho(\boldsymbol{A}, s) = \max\{|\lambda_{\max}(\boldsymbol{A}, s)|, |\lambda_{\min}(\boldsymbol{A}, s)|\}$. We assume that the regular observations follow model (3) in the manuscript and the corresponding empirical covariance matrix $\hat{\boldsymbol{\Sigma}}$ can be decomposed into

$$\hat{\boldsymbol{\Sigma}} = \boldsymbol{\Sigma} + \boldsymbol{E} \tag{S.3}$$

where $\boldsymbol{\Sigma}$ is the population covariance matrix. Denote by $\lambda_j(\boldsymbol{\Sigma})$ the jth eigenvector of $\boldsymbol{\Sigma}$, where $\lambda_1(\boldsymbol{\Sigma}) > \lambda_2(\boldsymbol{\Sigma}) > \ldots > \lambda_k(\boldsymbol{\Sigma})$. It can be seen that $\lambda_j(\boldsymbol{\Sigma}) = \rho_j^2$ in the multi-spike model. Let $\Delta\lambda = \lambda_1(\boldsymbol{\Sigma}) - \max_{j>1}(\lambda_j(\boldsymbol{\Sigma}))$ denote the gap between the largest eigenvalue and the remaining eigenvalues, and $\bar{s} = \|\boldsymbol{v}_1\|_0$ the true sparsity of the first principal component. For the single spike model it holds that $s = O(\bar{s})$ and with large probability $\rho(E, s) = O(\sqrt{s \log p/n})$ when the $\boldsymbol{z}_i$'s are random Gaussian noise (see Theorem 4 in Yuan and Zhang (2013) for more details). Under the assumption that $\Delta\lambda \geqslant 2\rho(E, s')$, with $s' = 2s + \bar{s}$ for some $s \geqslant \bar{s}$, Yuan and Zhang (2013) have shown that for the hard-thresholding penalty the estimation error is of order $\|\tilde{\boldsymbol{v}} - \boldsymbol{v}\| = O(\bar{s} \log(p)/n)$ and the upper bound decreases with larger $\Delta\lambda$. For the multiple spike model, the assumption needs to hold for the successive deflated matrices to ensure that the PCs are estimated with an acceptable error bound.

Additional simulation results

Here we present additional results for the simulation study in Section 3 of the manuscript. These results confirm the conclusions in the manuscript.

Estimation performance

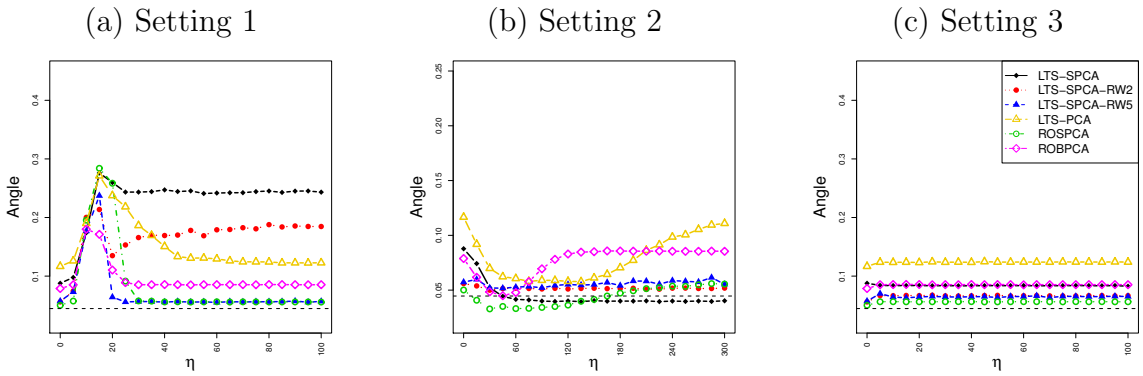

Figure S.1: Average angles for the three contamination settings with $n = 200$, $p = 20$, $b = 4$, $\varepsilon = 20\%$ and varying η. The dashed horizontal line is the result of sPCA-rSVD on clean data.

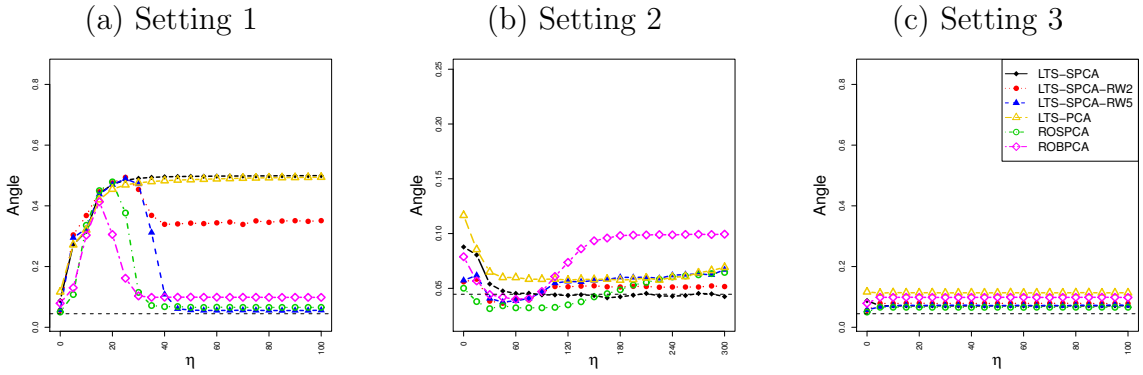

Figure S.2: Average angles for the three contamination settings with $n = 200$, $p = 20$, $b = 4$, $\varepsilon = 40\%$ and varying η. The dashed horizontal line is the result of sPCA-rSVD on clean data.

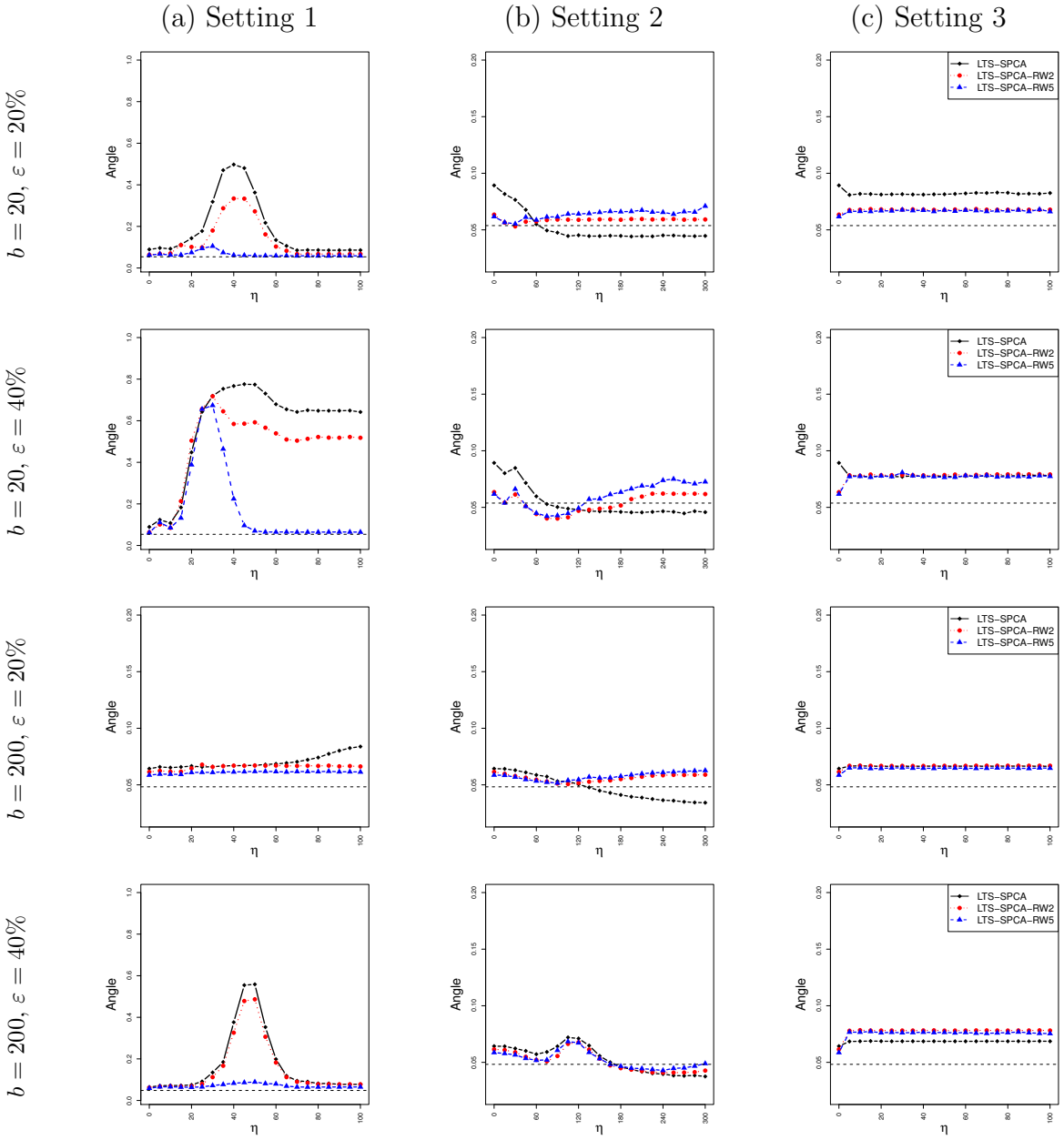

Figure S.3: Average angles computed for the three contamination settings corresponding to the three columns for $p = 2000$ with different b and ε, and varying η. The horizontal dashed line is the angle given by sPCA-rSVD on clean data.

Variable selection performance

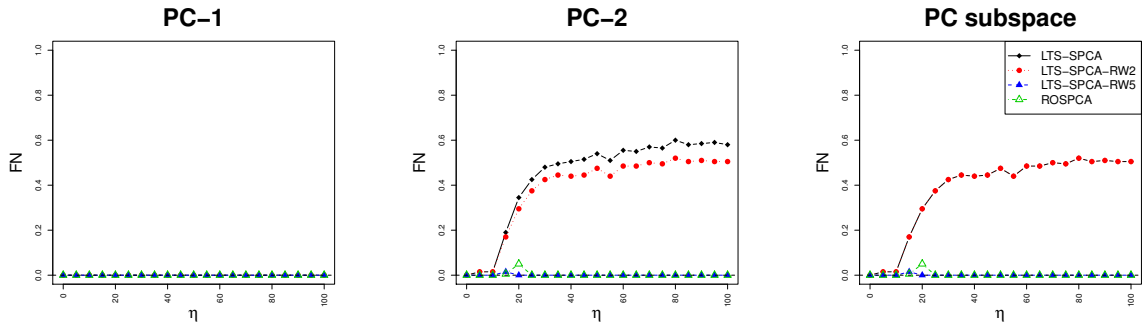

Figure S.4: Average false negatives (FN) for setting 1 with $n = 200$, $p = 20$, $b = 4$, $\varepsilon = 20\%$ and varying η. The dashed horizontal line is the result of sPCA-rSVD on clean data.

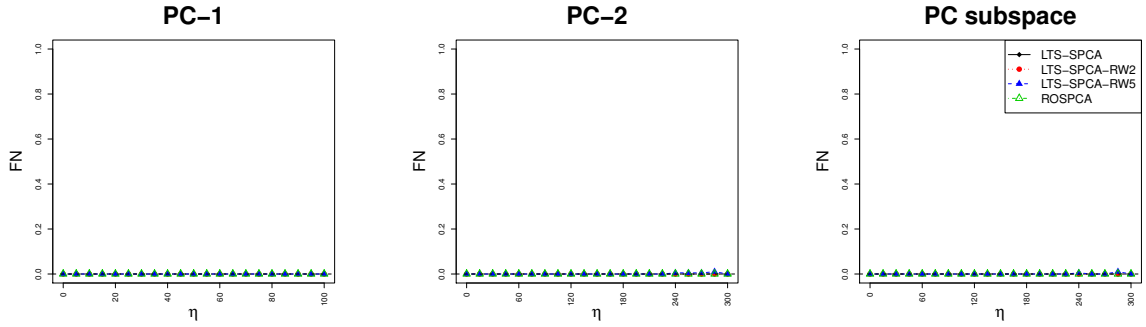

Figure S.5: Average false negatives (FN) for setting 2 with $n = 200$, $p = 20$, $b = 4$, $\varepsilon = 20\%$ and varying η. The dashed horizontal line is the result of sPCA-rSVD on clean data.

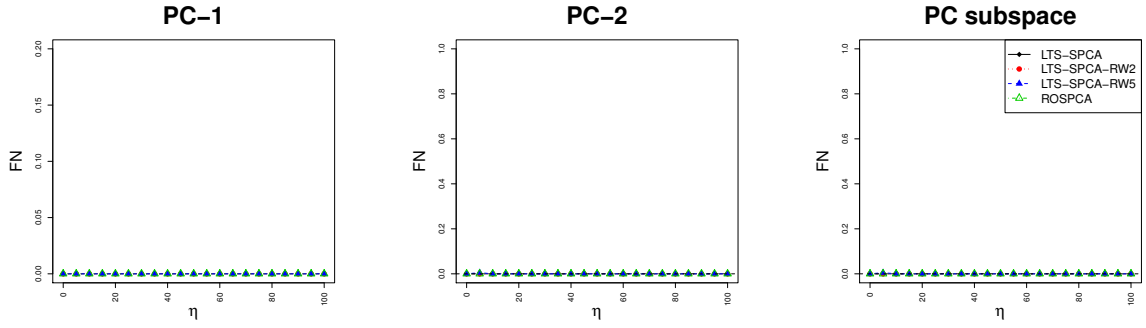

Figure S.6: Average false negatives (FN) for setting 3 with $n = 200$, $p = 20$, $b = 4$, $\varepsilon = 20\%$ and varying η. The dashed horizontal line is the result of sPCA-rSVD on clean data.

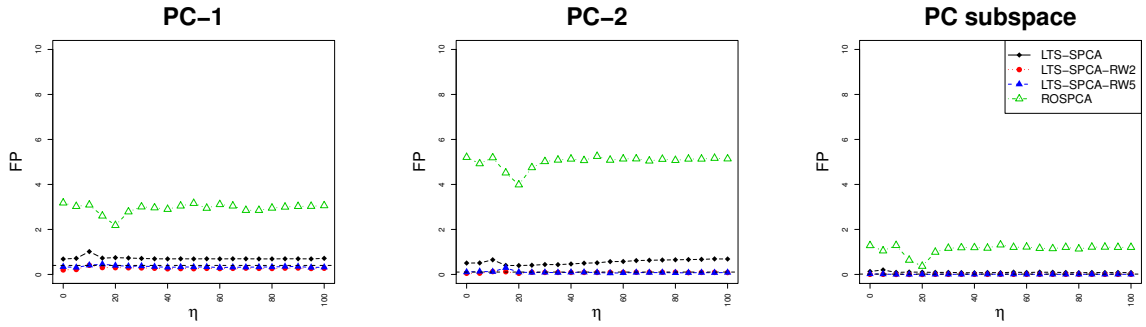

Figure S.7: Average false positives (FP) for setting 1 with $n = 200$, $p = 20$, $b = 4$, $\varepsilon = 20\%$ and varying η. The dashed horizontal line is the result of sPCA-rSVD on clean data.

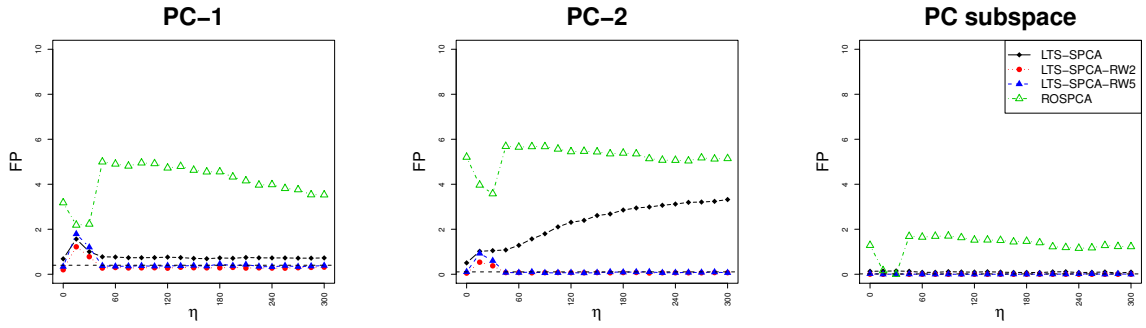

Figure S.8: Average false positives (FP) for setting 2 with $n = 200$, $p = 20$, $b = 4$, $\varepsilon = 20\%$ and varying η. The dashed horizontal line is the result of sPCA-rSVD on clean data.

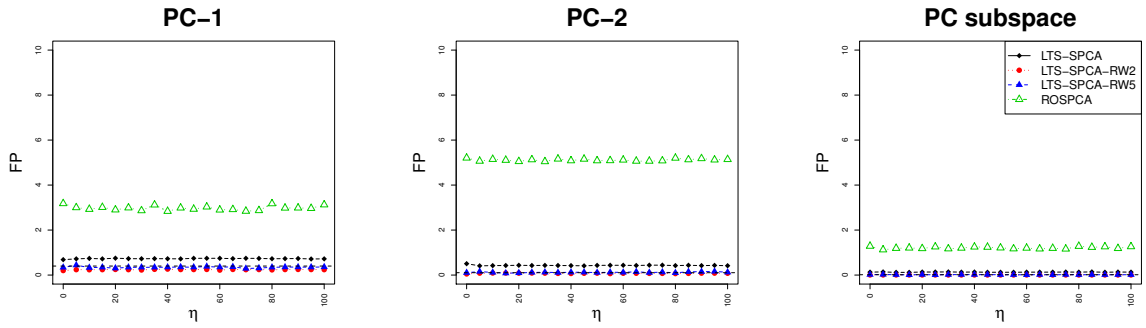

Figure S.9: Average false positives (FP) for setting 3 with $n = 200$, $p = 20$, $b = 4$, $\varepsilon = 20\%$ and varying η. The dashed horizontal line is the result of sPCA-rSVD on clean data.

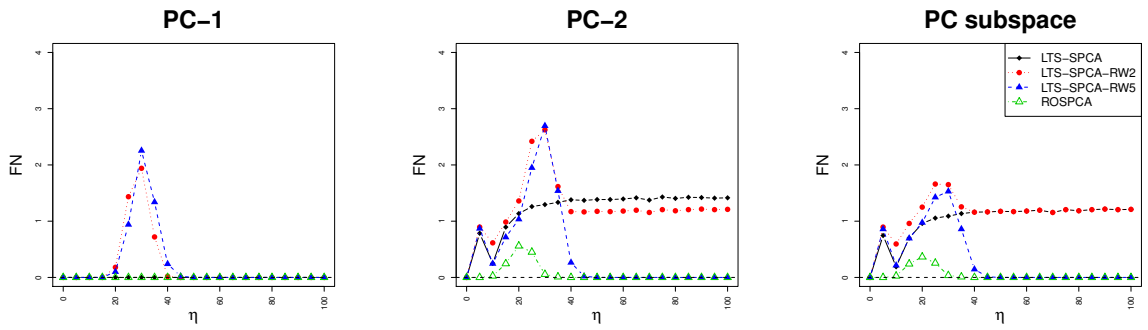

Figure S.10: Average false negatives (FN) for setting 1 with $n = 200$, $p = 20$, $b = 4$, $\varepsilon = 40\%$ and varying η. The dashed horizontal line is the result of sPCA-rSVD on clean data.

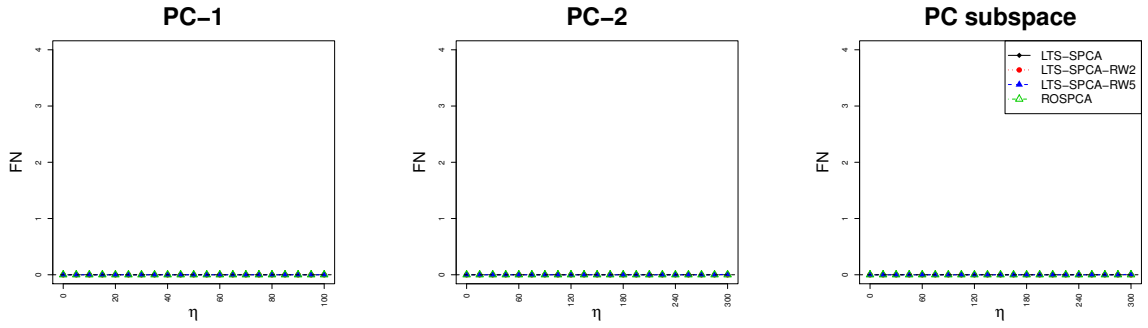

Figure S.11: Average false negatives (FN) for setting 2 with $n = 200$, $p = 20$, $b = 4$, $\varepsilon = 40\%$ and varying η. The dashed horizontal line is the result of sPCA-rSVD on clean data.

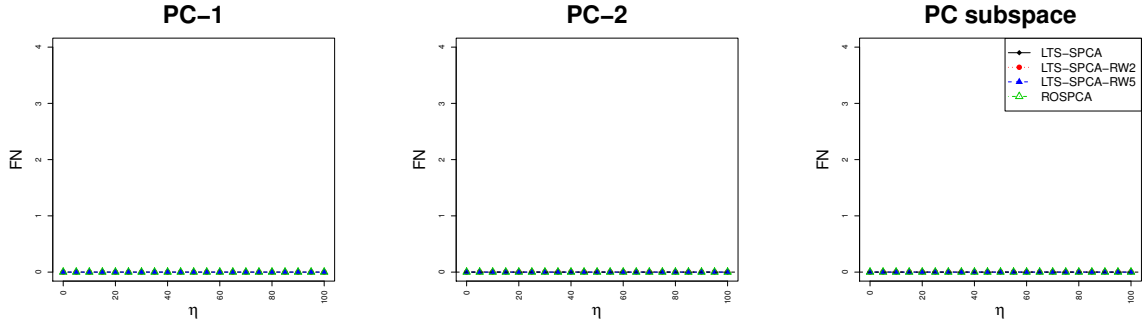

Figure S.12: Average false negatives (FN) for setting 3 with $n = 200$, $p = 20$, $b = 4$, $\varepsilon = 40\%$ and varying η. The dashed horizontal line is the result of sPCA-rSVD on clean data.

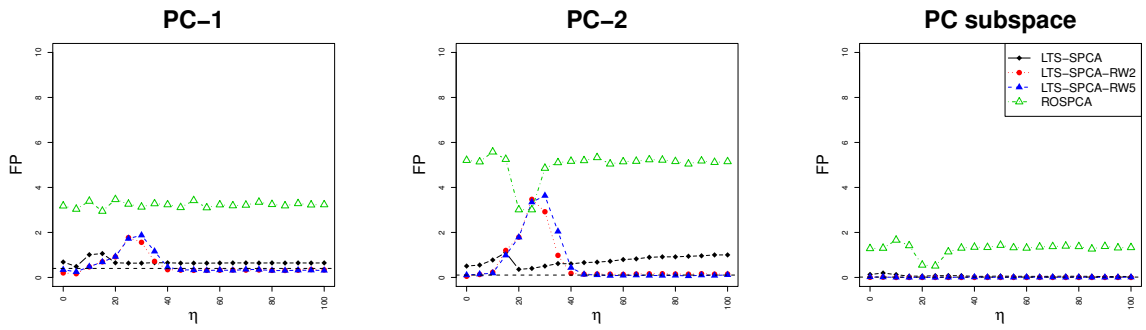

Figure S.13: Average false positives (FP) for setting 1 with $n = 200$, $p = 20$, $b = 4$, $\varepsilon = 40\%$ and varying η. The dashed horizontal line is the result of sPCA-rSVD on clean data.

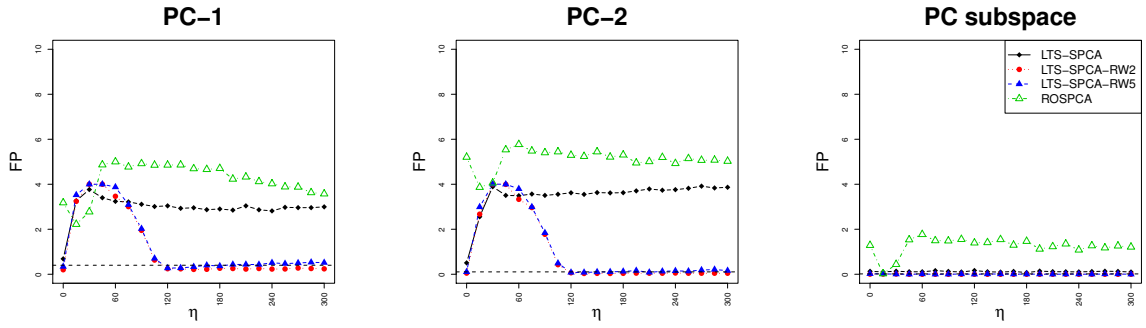

Figure S.14: Average false positives (FP) for setting 2 with $n = 200$, $p = 20$, $b = 4$, $\varepsilon = 40\%$ and varying η. The dashed horizontal line is the result of sPCA-rSVD on clean data.

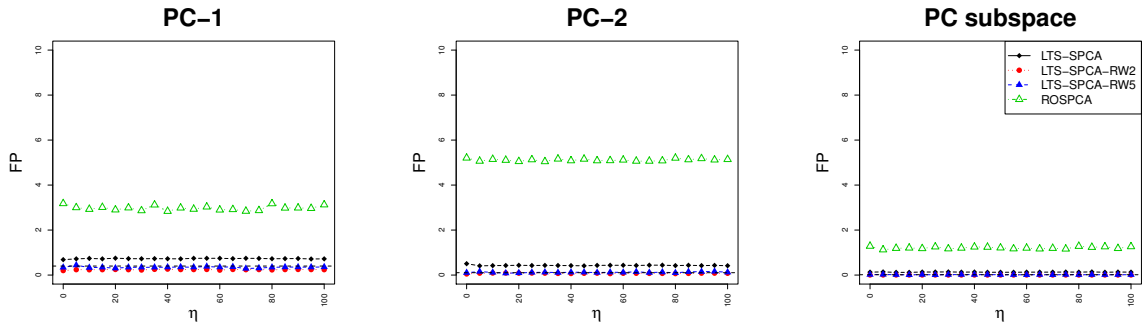

Figure S.15: Average false positives (FP) for setting 3 with $n = 200$, $p = 20$, $b = 4$, $\varepsilon = 40\%$ and varying η. The dashed horizontal line is the result of sPCA-rSVD on clean data.

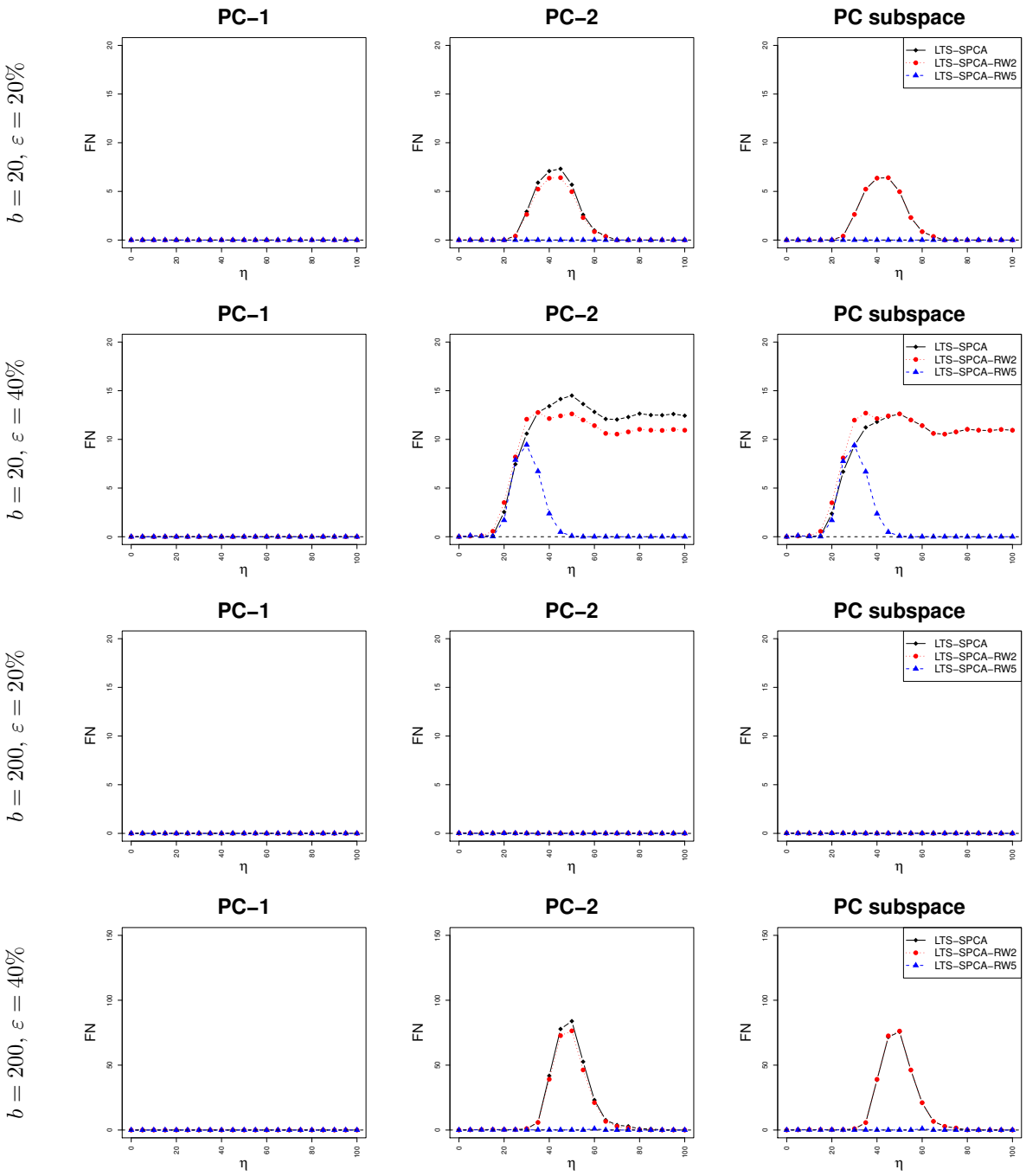

Figure S.16: Average false negatives (FN) for setting 1 with $n = 200$, $p = 2000$, $b = 20$ or 200, $\varepsilon = 20\%$ or 40%, and varying η. The dashed horizontal line is the result of sPCA-rSVD on clean data.

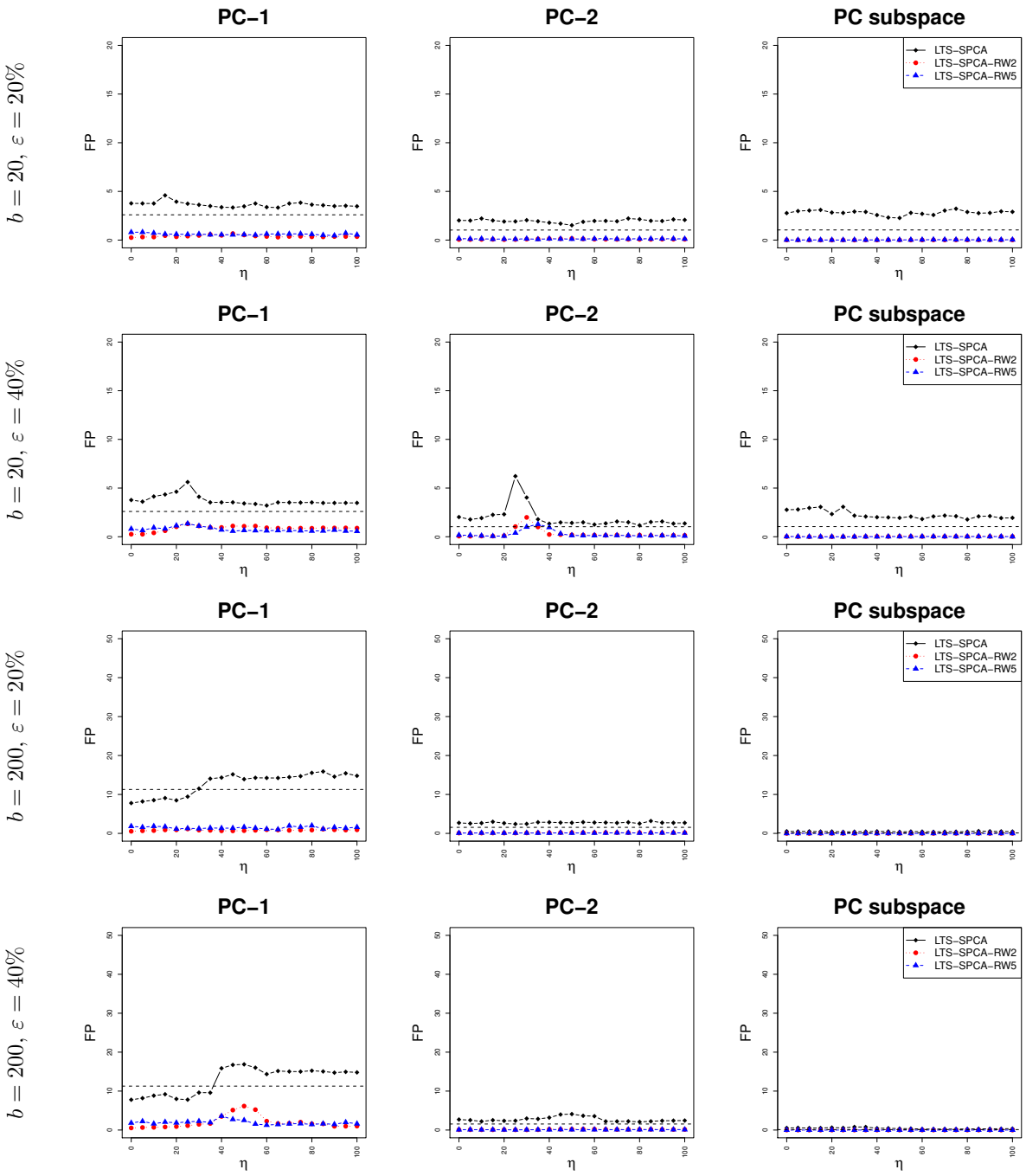

Figure S.17: Average false positives (FP) for setting 1 with $n = 200$, $p = 2000$, $b = 20$ or 200, $\varepsilon = 20\%$ or 40%, and varying η. The dashed horizontal line is the result of sPCA-rSVD on clean data.

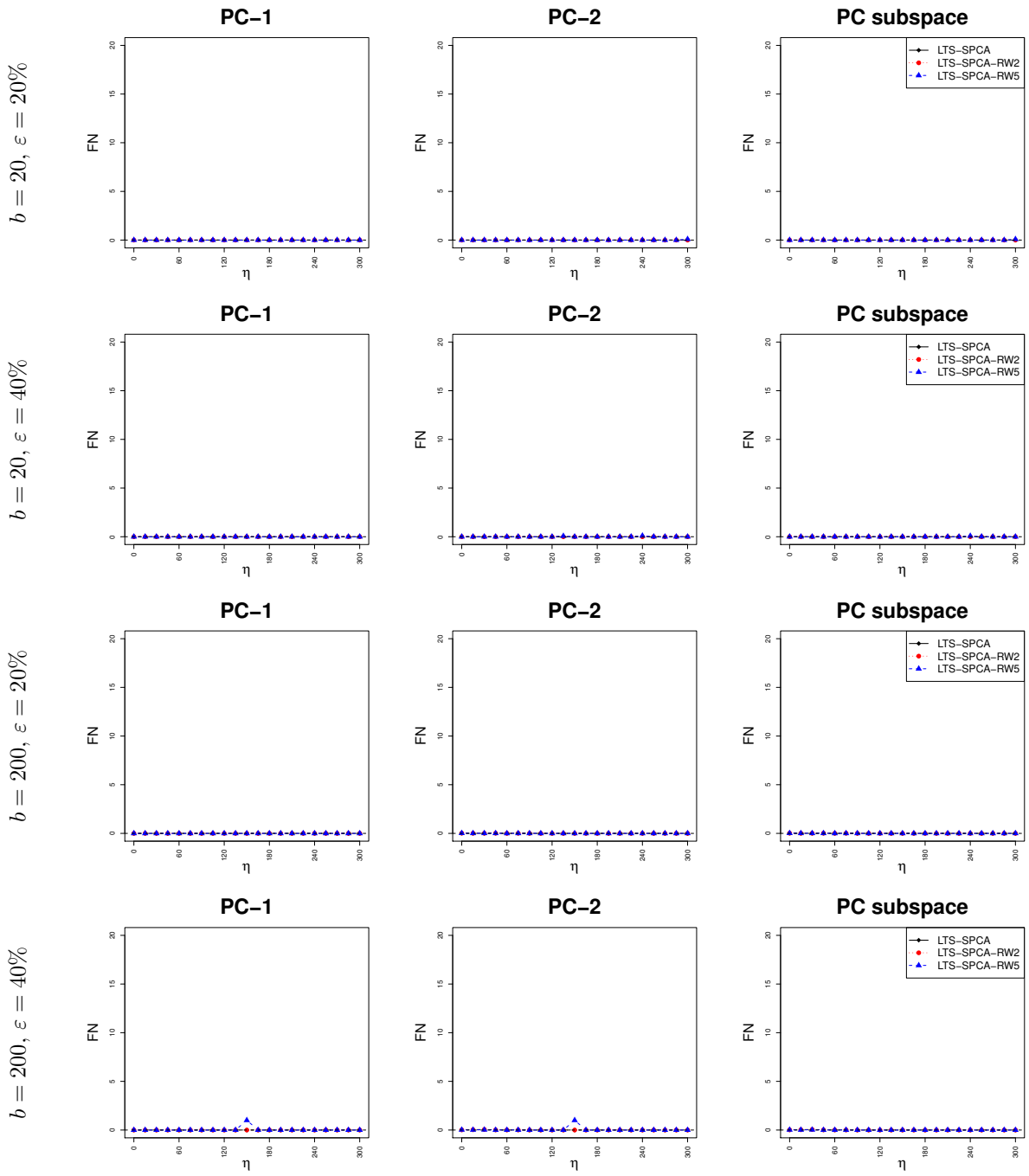

Figure S.18: Average false negatives (FN) for setting 2 with $n = 200$, $p = 2000$, $b = 20$ or 200, $\varepsilon = 20\%$ or 40%, and varying η. The dashed horizontal line is the result of sPCA-rSVD on clean data.

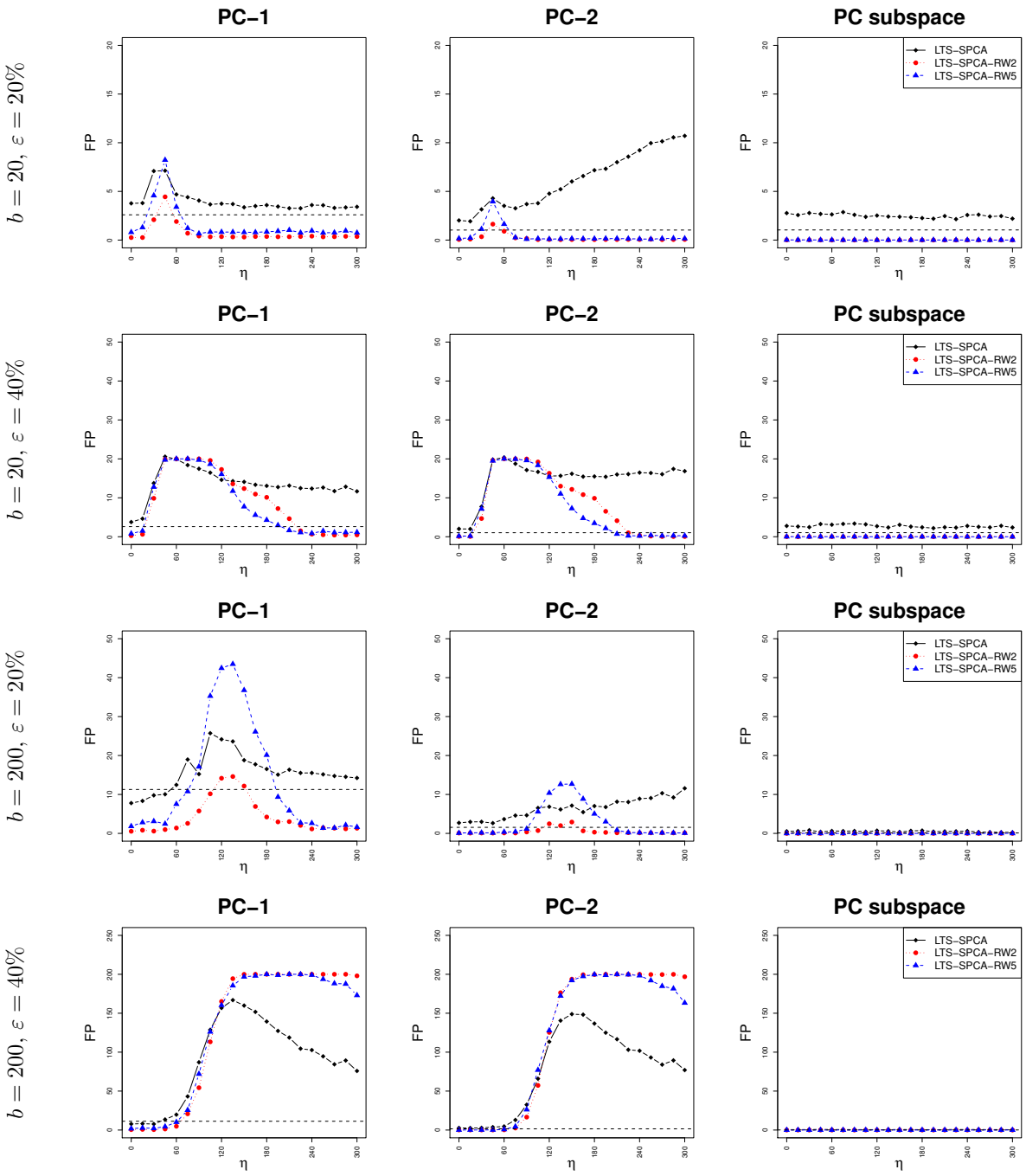

Figure S.19: Average false positives (FP) for setting 2 with $n = 200$, $p = 2000$, $b = 20$ or 200, $\varepsilon = 20\%$ or 40%, and varying η. The dashed horizontal line is the result of sPCA-rSVD on clean data.

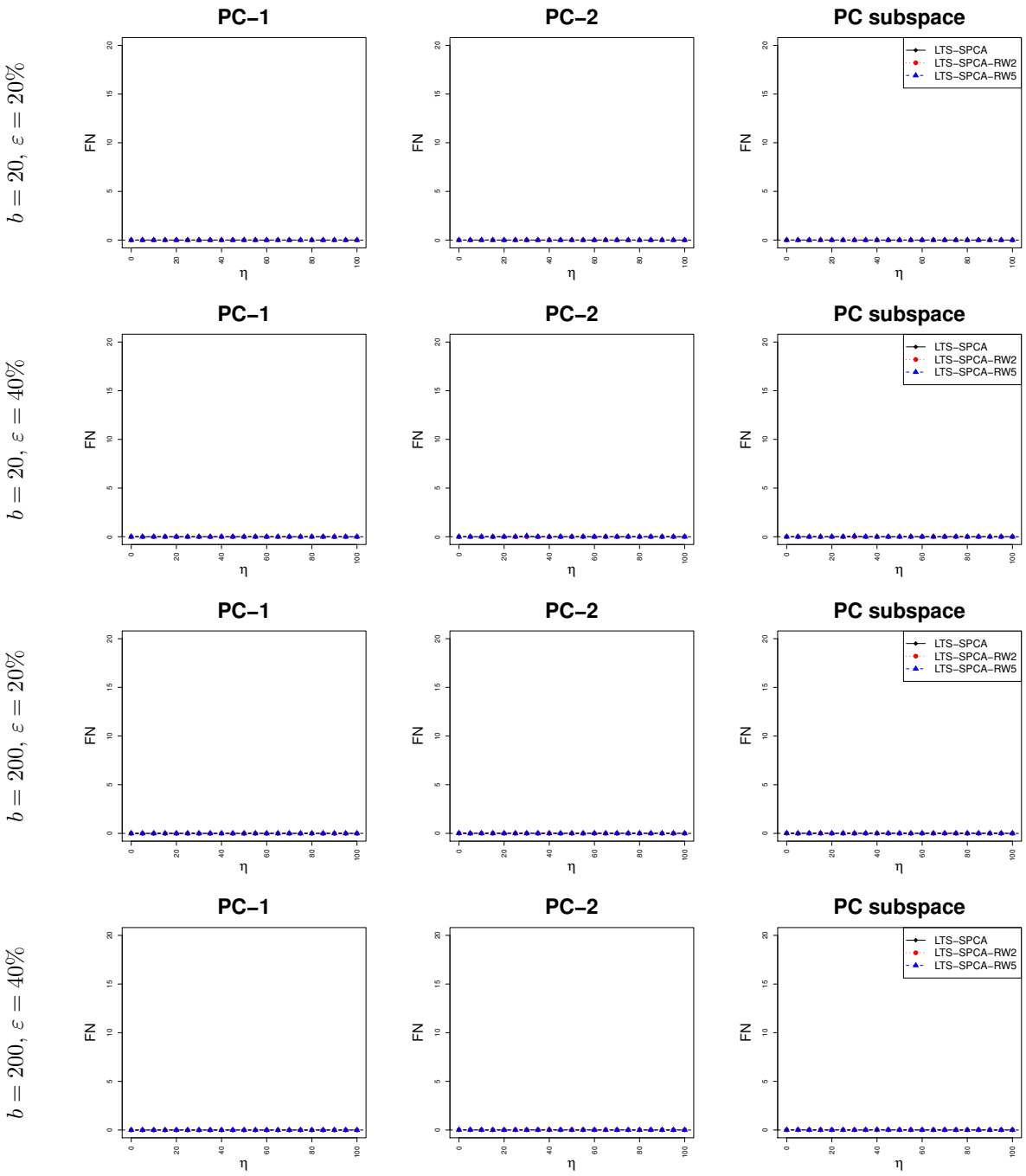

Figure S.20: Average false negatives (FN) for setting 3 with $n = 200$, $p = 2000$, $b = 20$ or 200, $\varepsilon = 20\%$ or 40%, and varying η. The dashed horizontal line is the result of sPCA-rSVD on clean data.

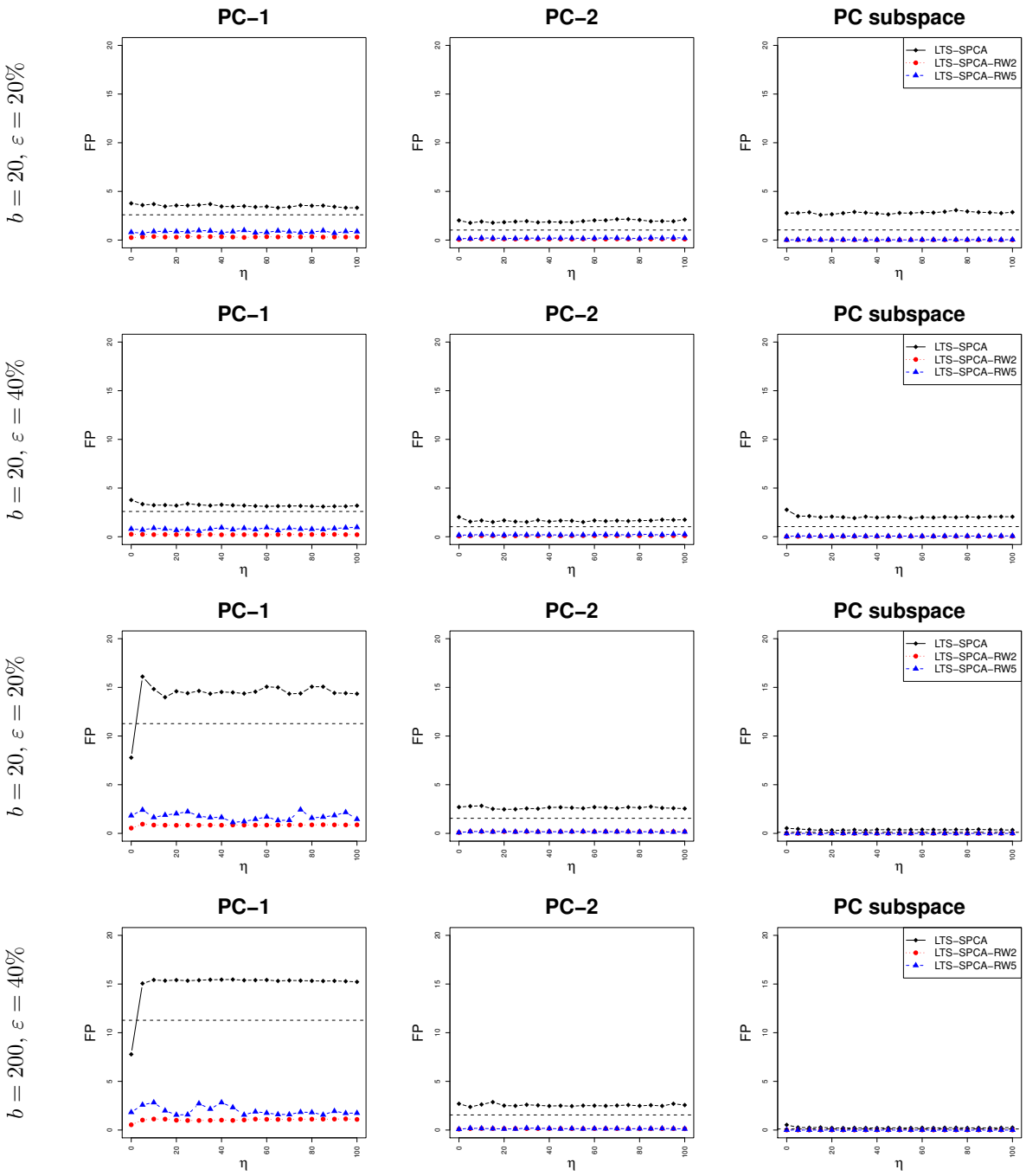

Figure S.21: Average false positives (FP) for setting 3 with $n = 200$, $p = 2000$, $b = 20$ or 200, $\varepsilon = 20\%$ or 40%, and varying η. The dashed horizontal line is the result of sPCA-rSVD on clean data.

Outlier detection performance

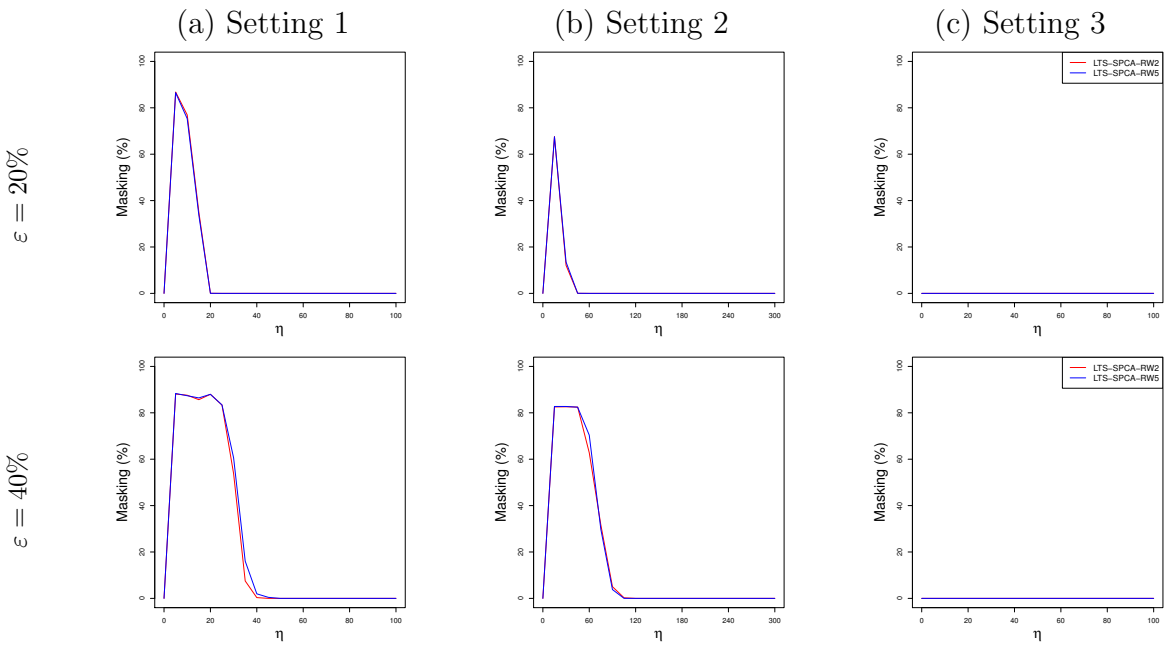

Figure S.22: The masking effect for the three contamination settings with $n = 200$, $p = 20$, $b = 4$, $\varepsilon = 20\%$ or 40%, and varying η.

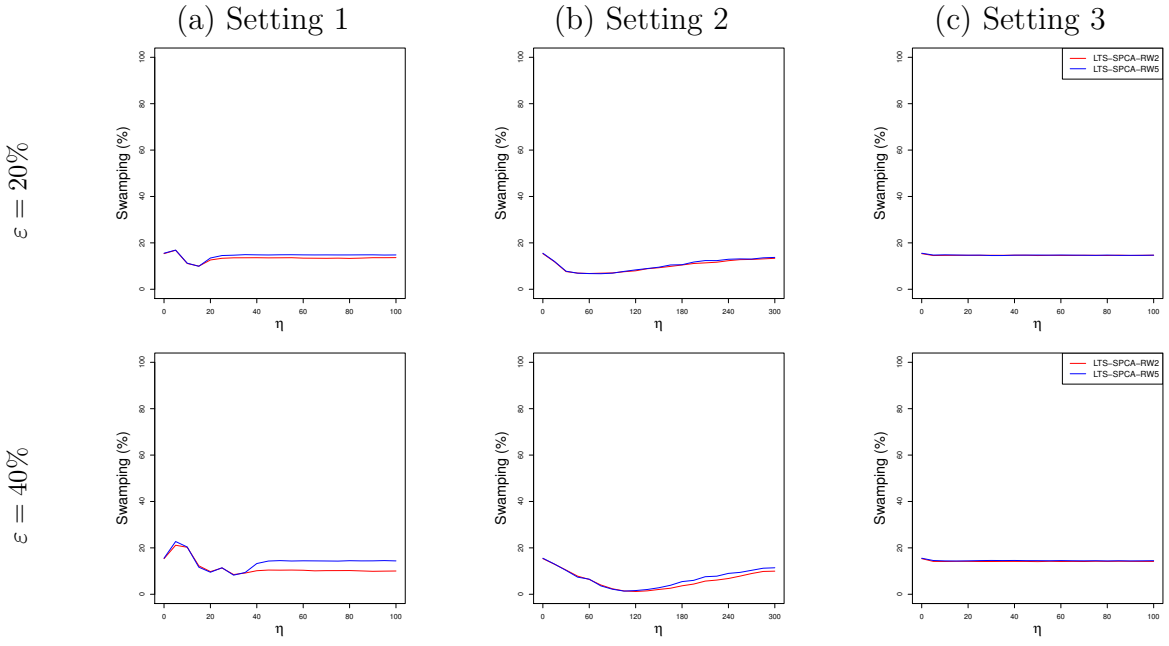

Figure S.23: The swamping effect for the three contamination settings with $n = 200$, $p = 20$, $b = 4$, $\varepsilon = 20\%$ or 40%, and varying η.

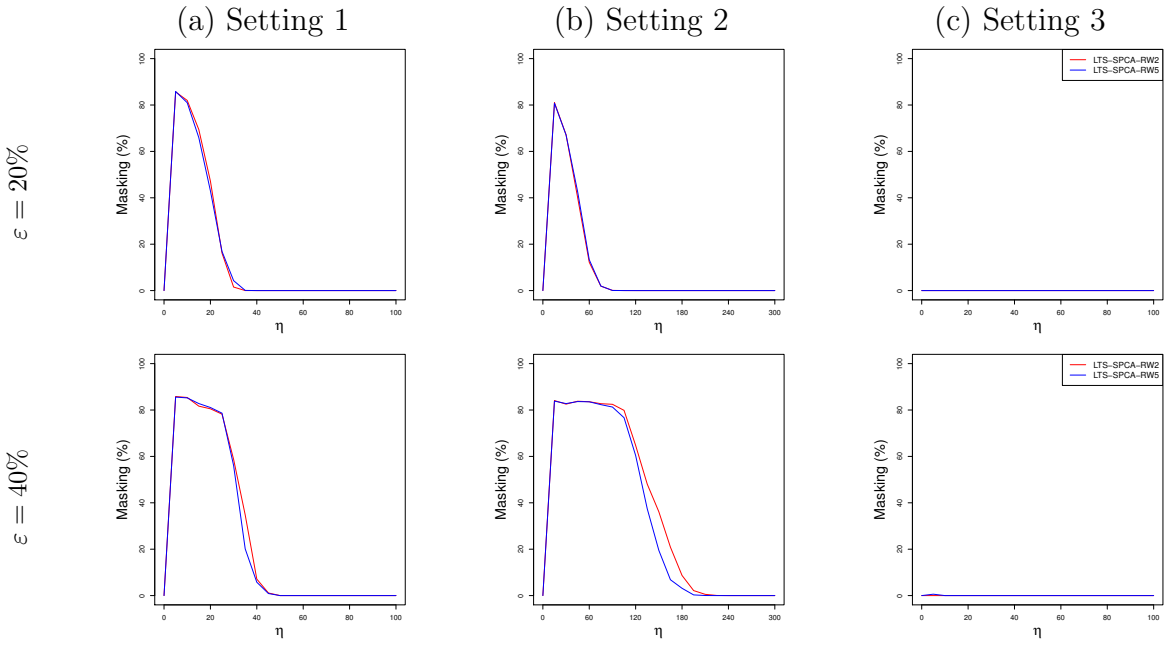

Figure S.24: The masking effect for the three contamination settings with $n = 200$, $p = 2000$, $b = 20$, $\varepsilon = 20\%$ or 40%, and varying η.

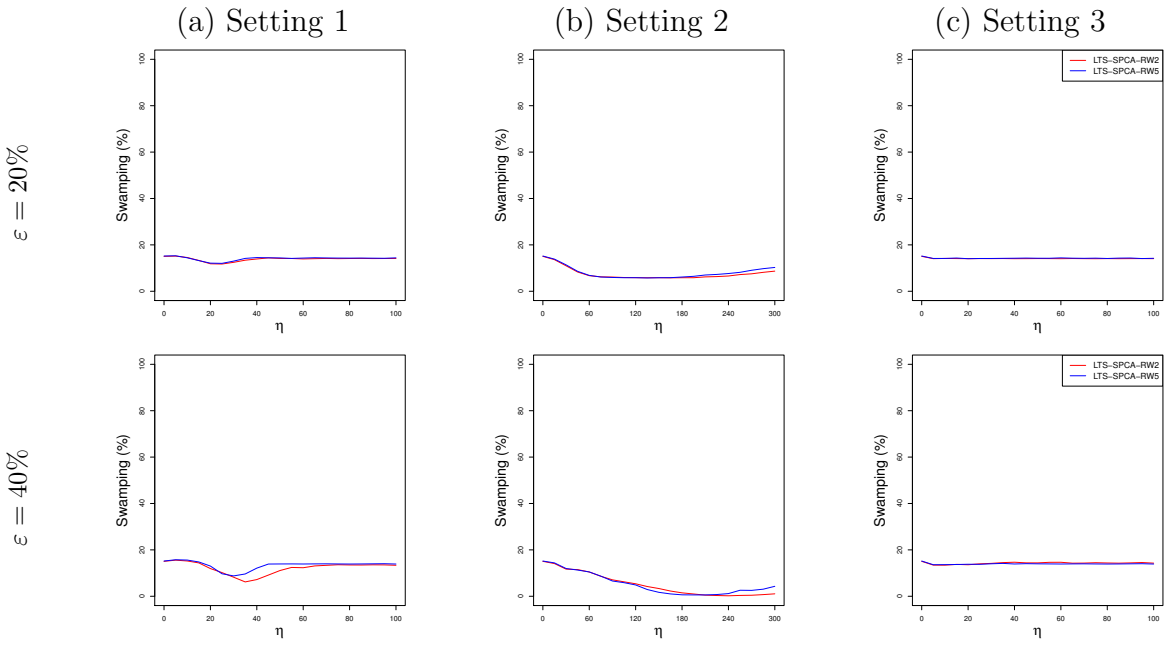

Figure S.25: The swamping effect for the three contamination settings with $n = 200$, $p = 2000$, $b = 20$, $\varepsilon = 20\%$ or 40%, and varying η.

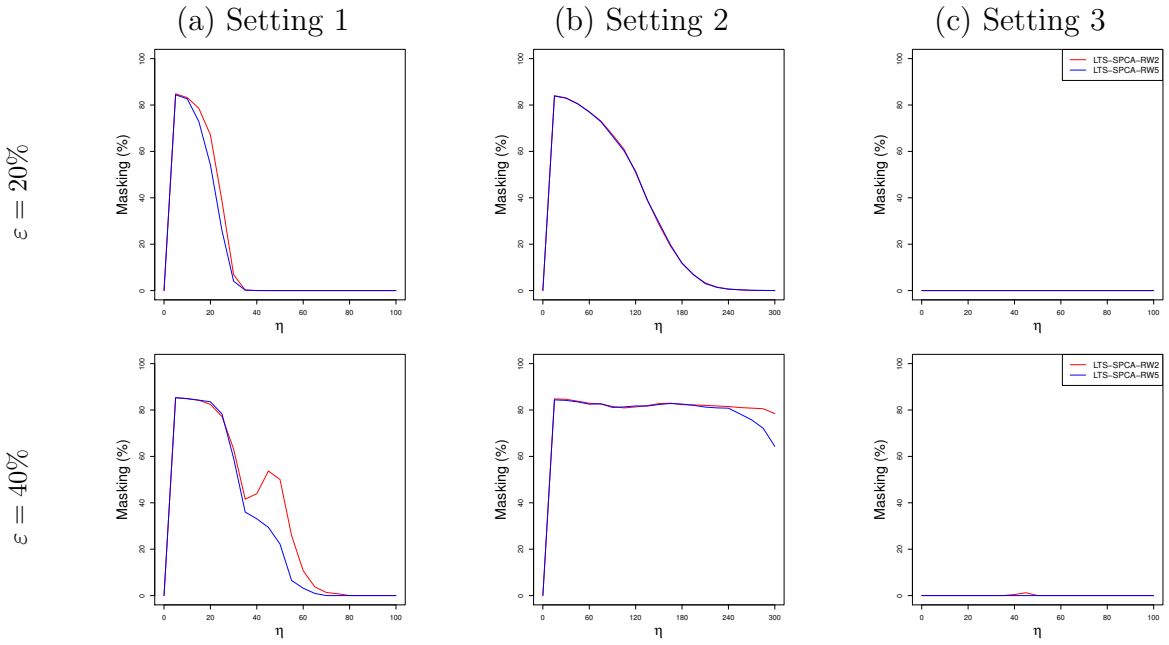

Figure S.26: The masking effect for the three contamination settings with $n = 200$, $p = 2000$, $b = 200$, $\varepsilon = 20\%$ or 40%, and varying η.

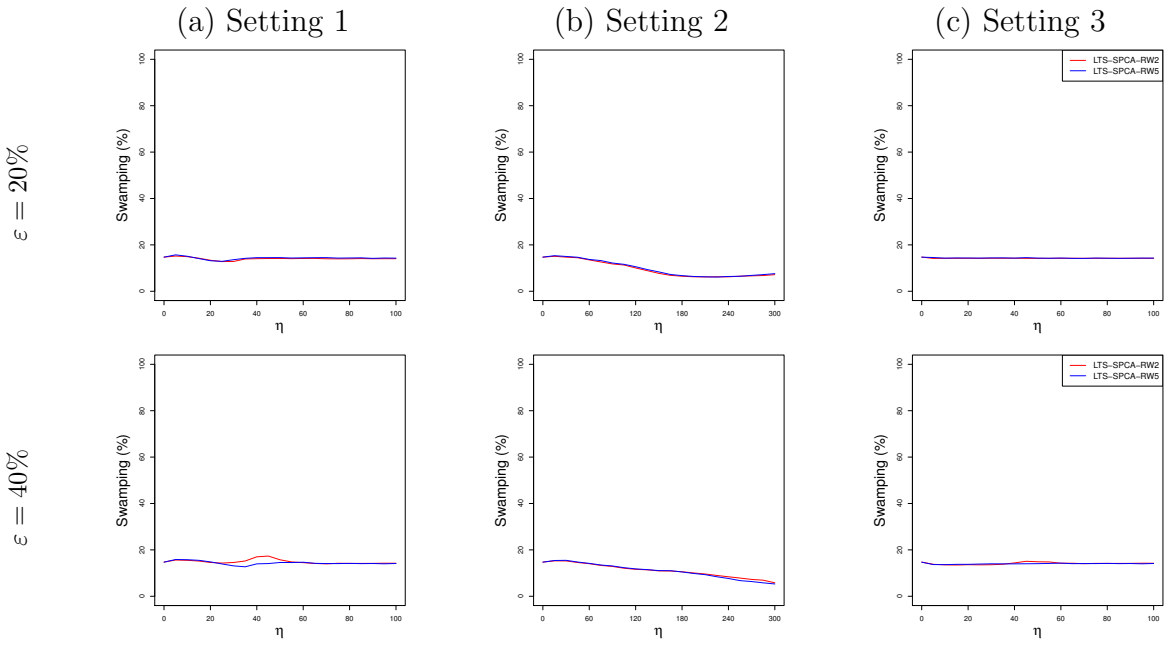

Figure S.27: The swamping effect for the three contamination settings with $n = 200$, $p = 2000$, $b = 200$, $\varepsilon = 20\%$ or 40%, and varying η.

Escalator video data continued

Figure S.28 displays images of the original video (left) and the diagnostic plots corresponding to LTS-SPCA (middle) and LTS-PCA (right). The four rows in the first column correspond to frames 1, 30, 70, and 100 of the video. On the diagnostic plots, the observation is then marked by a cross. The positioning information for all the frames can be found at `https://www.dropbox.com/s/ykngmddjw0o9gnh/video-out-detect-4pc.mp4?dl=0`. These plots show that LTS-SPCA is more consistent with the movement in the original video. When we only look at the scaled ODs on the vertical axis, the two methods deliver similar information. The emerging persons result in an increase of the OD. Indeed, the first 54 frames are flagged as orthogonal outliers by LTS-SPCA/LTS-PCA. However, within the PC subspace, the pictures are quite different. The score weights given by LTS-SPCA change consistently with the moving crowds on the escalator. In the beginning, when people start gathering above the escalator, the frames jump out as bad leverage points on the diagnostic plot. Finally, when the moving crowds disappear in the video, the corresponding frames become non-outlying. On the other hand, the score weights corresponding to LTS-PCA show a rather random variation without clear relation to the movement in the video.

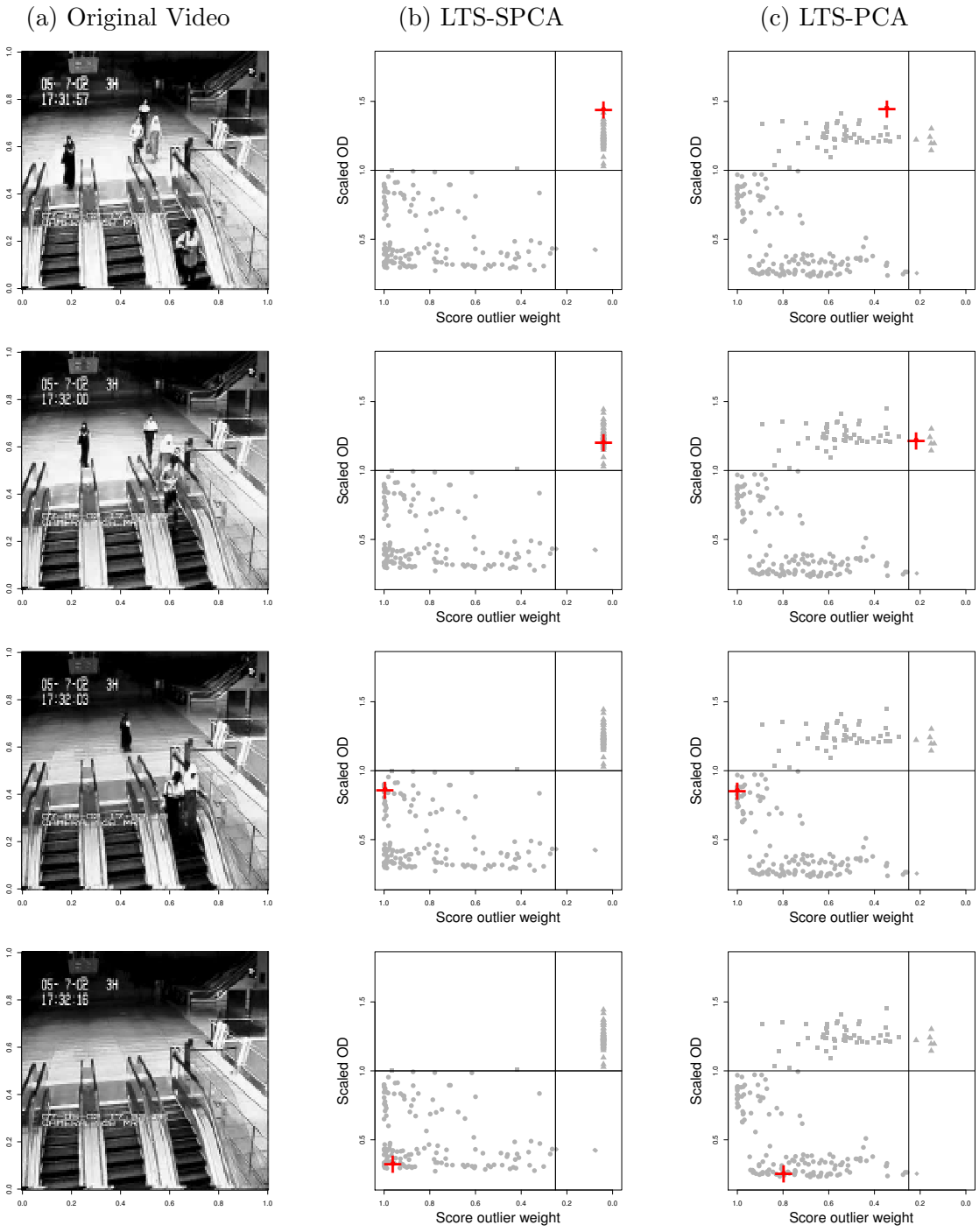

Figure S.28: The original video (a), and the diagnostic plots given by LTS-SPCA (b), and by LTS-PCA (c).

A toy example

Here, we use a toy example to illustrate potential problems that may occur when estimating the whole PC subspace at once with LTS-PCA. Then, we use the same example to show how the LTS-(S)PCA works with the deflation method.

To estimate the whole PC subspace at the same time, we need to pre-specify the dimension of the PC subspace. As discussed in the manuscript, the outlier detection performance depends on the dimension of the PC subspace. It is highly possible for an orthogonal outlier detected in the lower dimensional PC subspace to be a score outlier in an enlarged PC subspace. These score outliers may then affect all the PCs which span the PC subspace. This can lead to instability in the detection of the outliers. If some of the outliers are modeled by the PC subspace and fail to be detected, they will not only destroy the sparse structure within the PC subspace, but also inflate the estimate of the explained variance and lead to an erroneous estimation of the PC dimension. Moreover, due to the presence of outliers, the estimated PC subspaces will not be necessarily nested, i.e. lower dimensional PC subspaces may not be covered by higher dimensional subspaces. This further complicates outlier detection and the selection of the PC dimension.

We use a toy example to illustrate this phenomenon. We simulate a set of regular observations from a multivariate Gaussian distribution with block diagonal covariance matrix as in the simulation study, but with three correlated blocks of important variables. More explicitly, we generate $\boldsymbol{X} = \boldsymbol{X}_u + \boldsymbol{X}_{\text{noise}}$ with $\boldsymbol{X}_u \sim N_p(\boldsymbol{0}, \boldsymbol{\Sigma})$, $\boldsymbol{X}_{\text{noise}} \sim N_p(\boldsymbol{0}, \boldsymbol{I}_p)$, $\boldsymbol{\Sigma} = \boldsymbol{D}^{\frac{1}{2}}\boldsymbol{R}\boldsymbol{D}^{\frac{1}{2}}$ with $\boldsymbol{D}$ a diagonal matrix, and

$$\boldsymbol{R} = \begin{pmatrix} \boldsymbol{R}(a_1) & \boldsymbol{0}_{b\times b} & \boldsymbol{0}_{b\times b} & \boldsymbol{0}_{b\times(p-3b)} \\ \boldsymbol{0}_{b\times b} & \boldsymbol{R}(a_2) & \boldsymbol{0}_{b\times b} & \boldsymbol{0}_{b\times(p-3b)} \\ \boldsymbol{0}_{b\times b} & \boldsymbol{0}_{b\times b} & \boldsymbol{R}(a_3) & \boldsymbol{0}_{b\times(p-3b)} \\ \boldsymbol{0}_{(p-3b)\times b} & \boldsymbol{0}_{(p-3b)\times b} & \boldsymbol{0}_{(p-3b)\times b} & \boldsymbol{I}_{(p-3b)} \end{pmatrix}$$

where $\boldsymbol{R}(y)$ ($|y| < 1$) is a $b \times b$ matrix with ones on its diagonal and y as the off-diagonal elements. We take $n = 200$, $p = 20$, $b = 4$, $a_1 = 0.9$, $a_2 = 0.7$, $a_3 = 0.5$ and $\boldsymbol{D} = \text{diag}(\underbrace{10^2, \ldots, 10^2}_{b}, \underbrace{7^2, \ldots, 7^2}_{b}, \underbrace{5^2, \ldots, 5^2}_{b}, \underbrace{2^2, \ldots, 2^2}_{b})$. For this setting, the regular data has three principal components which are the three leading eigenvectors of the covariance

matrix $\boldsymbol{\Sigma}$:

$$\begin{aligned}
\boldsymbol{v}_1 &= (\underbrace{0.5,\ldots,0.5}_{4},\underbrace{0,\ldots,0}_{4},\underbrace{0,\ldots,0}_{4},\underbrace{0,\ldots,0}_{8})^{\mathrm{T}} \\
\boldsymbol{v}_2 &= (\underbrace{0,\ldots,0}_{4},\underbrace{0.5,\ldots,0.5}_{4},\underbrace{0,\ldots,0}_{4},\underbrace{0,\ldots,0}_{8})^{\mathrm{T}} \\
\boldsymbol{v}_3 &= (\underbrace{0,\ldots,0}_{4},\underbrace{0,\ldots,0}_{4},\underbrace{0.5,\ldots,0.5}_{4},\underbrace{0,\ldots,0}_{8})^{\mathrm{T}}
\end{aligned}$$

Then, we add 20% of observations which are outlying in two variables in two blocks of correlated variables and have larger variances in the unimportant variables. In particular, we set $(\boldsymbol{X}_{\text{out}})_j \sim N(15,3)$ for $j \in \{1,2,5,6\}$, $(\boldsymbol{X}_{\text{out}})_j \sim N(0,3)$ for $j \in \{13,\ldots,20\}$, and $(\boldsymbol{X}_{\text{out}})_j = \boldsymbol{X}_j$ otherwise. The data matrix is shown in Figure S.29, where each line represents one observation, the grey lines represent the clean observations, and the red lines represent outliers. It can be seen that the outliers are very close to the regular observations, and are hard to detect if we inspect each variable individually.

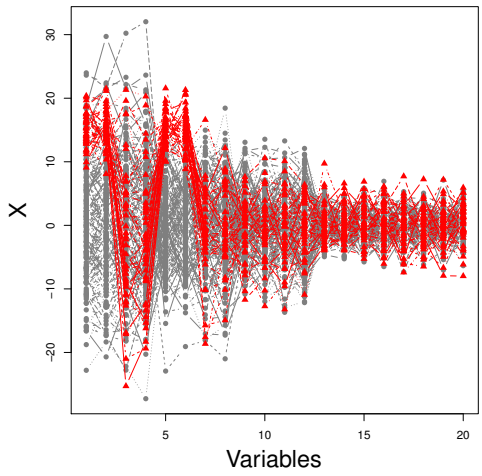

Figure S.29: The plot of the original data matrix for the toy example.

We first apply LTS-PCA to estimate a k-dimensional PC subspace for different choices of k. The resulting loadings for different dimensions k are displayed in Figure S.30. It can be seen that for $k = 1$, the single PC mainly contains information from the first block of important variables. For $k = 2$, LTS-PCA produces two PCs which model information from the first and the second block of important variables, respectively. We can thus conclude

that the first PC or the first two PCs are successfully recovered by LTS-PCA when $k = 1$ or $k = 2$, respectively (ignoring the small loadings for the other variables due to the lack of sparsity). However, for $k = 3$, not only the third PC but also the first one is estimated erroneously. In this case, the three PCs fail to recover the block structure within the PC subspace.

Similar phenomena can be seen when estimating more PCs. For $k \geqslant 3$, the LTS-PCA PC subspace is affected by the outliers. Indeed, several outliers have small orthogonal distances as can be seen from Figure S.31. When $k = 3$ or 4, most of the outliers also have small score weights and thus cannot be detected. From Figure S.32 (a) and (b), we can see that they indeed lie very close to the regular data in the k-dimensional PC subspace. If we estimate a 5-dimensional PC subspace, the fifth PC provides a separation between the outliers and the regular data which makes the outliers detectable as score outliers (see Figure S.32 (c)). However, if we enlarge the dimension to $k = 6$ the estimates of all PCs change (see Figure S.30 (f)) and the outliers become undetectable again (see Figure S.31 (f) and Figure S.32 (d)). Any unflagged outliers unduly inflate the estimated variance contained in the PC subspace, and deteriorate the estimation of the PC dimension.

On the other hand, if we search for $k_{\max} = 6$ PCs sequentially by running LTS-PCA with $k = 1$ on a set of deflated matrix (i.e. the non-sparse analog of LTS-SPCA), we can get a set of nested PC subspaces, which means that lower dimensional PC subspaces are now fully covered by higher dimensional PC subspaces. The resulting PC loadings are displayed in Figure S.33 (a). It can be seen that the first two PCs are both well estimated (again ignoring small loadings on other variables). In contrast to the previous approach, the inclusion of a higher order PC does not influence the estimation of its leading PCs. The diagnostic plots for different dimensions k are provided in Figure S.34. The diagnostic plots for $k = 1$ and $k = 2$ are very similar to those in Figure S.31. The outliers can be flagged as orthogonal outliers with respect to the k-dimensional PC subspace whenever $k \leqslant 4$. When the 5th PC is included, some outliers become score outliers. From the scatter plot of the scores in Figure S.33 (b), we can clearly observe the separation of the regular observations and outliers on the 5th PC. By assigning a large weight to the 5th PC according to its kurtosis, the outliers will be flagged as score outliers within the PC

subspace (see Figure S.34 (e)). Furthermore, after modeling the outlying information by the 5th PC, those outliers will have non-outlying projections on the subsequent PCs. The extra PCs model the remaining information in the data, and bear very limited information. Hence, including the extra PCs will not have much influence on the detection of outliers.

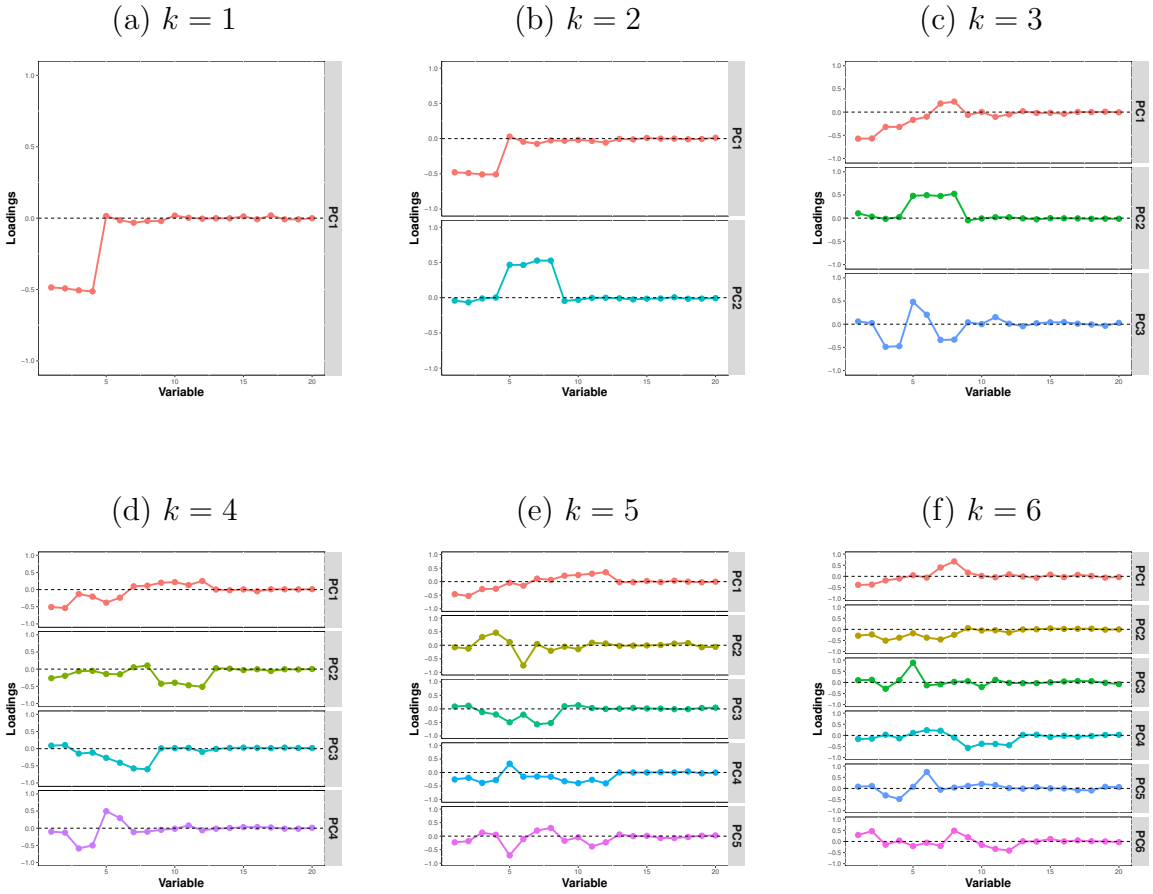

Figure S.30: Plots of the PC loadings when estimating the whole PC subspace at once using LTS-PCA for different dimensions k on the toy example.

Now, we explain how LTS-SPCA works on the toy example. We search for the PCs sequentially using LTS-SPCA together with the deflation method. We start with searching for the first PCs using the original data matrix (see Figure S.29) in which the outliers are hard to detect if we inspect each variable individually. However, all the outliers are successfully excluded from the h-subset, and this leads to an accurate estimate of the first PC, see Figure S.35 (a1). The deflated matrix estimated from the first PC is shown in Figure S.35 (a2). We can observe that most of the variability in the first block of correlated variables is removed by deflation. Moreover, the outliers are more visible in the deflated

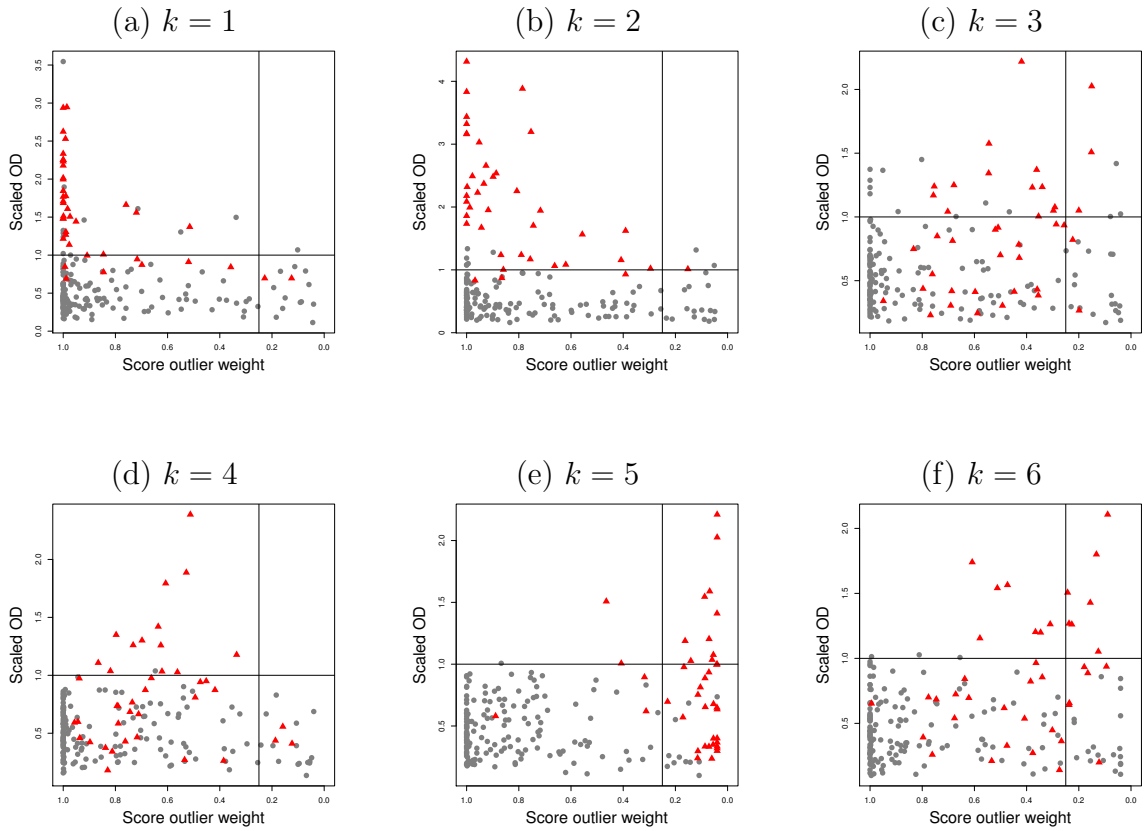

Figure S.31: Diagnostic plots when estimating the whole PC subspace using LTS-PCA for different dimensions k on the toy example: ● represents the regular data, and ▲ represents the outliers.

matrix than in the original matrix. By estimating the second and the third PCs (see Figure S.35 (b1) and (c1)), we sequentially remove most of the variability in the next two blocks of correlated variables (see Figure S.35 (b2) and (c2)). The first three PCs together give an accurate estimation of the true PC subspace, and the outliers become more visible in its orthogonal complement (see Figure S.35 (c2)). The subsequent PCs, especially the 6th PC, will model the extra variability introduced by the outliers, since that variability dominates the noise remaining in the regular data. Therefore, most of the outliers become score outliers (see Figure S.36 (f)) when the 6th PC is included, and they are not visible anymore after deflation (see Figure S.36 (f2)).

Starting from the 6-dimensional PC subspace, we run a reweighting step with all the detected outliers removed. The final loadings are shown in Figure S.37 (a). From the scree plot (see Figure S.37 (b)), we can select $k = 3$ PCs. We can see that the first three PCs

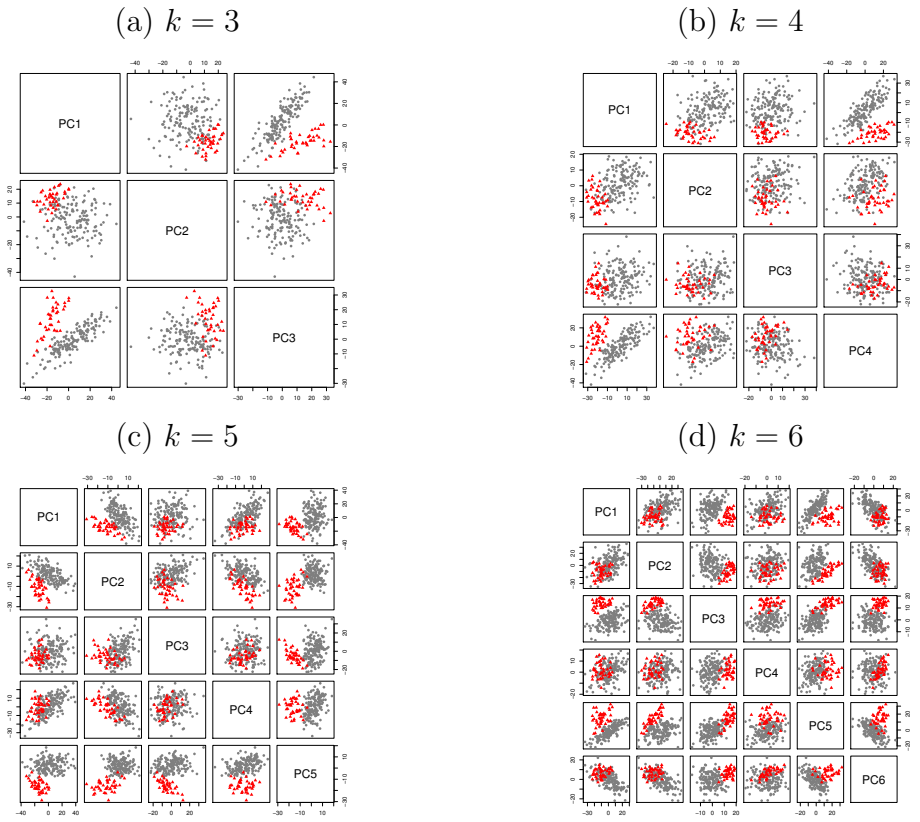

Figure S.32: Scatter plots of the scores when estimating the whole PC subspace using LTS-PCA for different dimensions k on the toy example: • represents the regular data, and ▲ represents the outliers.

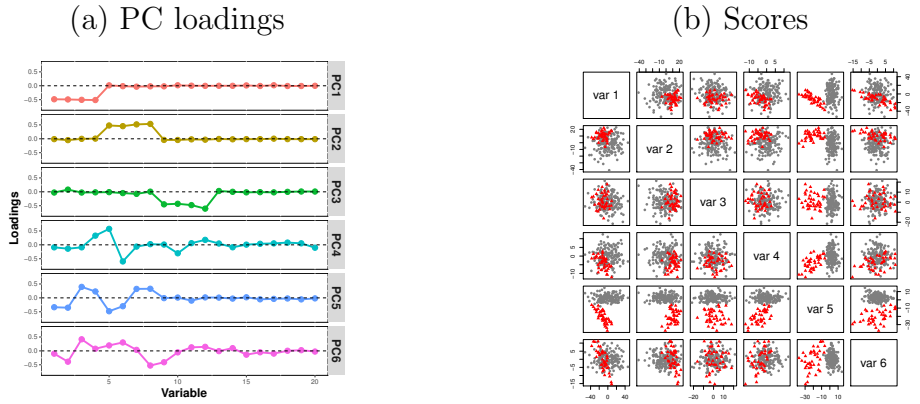

Figure S.33: (a) Plot of the PC loadings and (b) scatter plot of the scores when estimating a 6-dimensional PC subspace sequentially using LTS-PCA together with the deflation method on the toy example: • represents the regular data, and ▲ represents the outliers.

are well-estimated, and the last three PCs now mainly contain noise. The final diagnostic plot is shown in Figure S.37 (c).

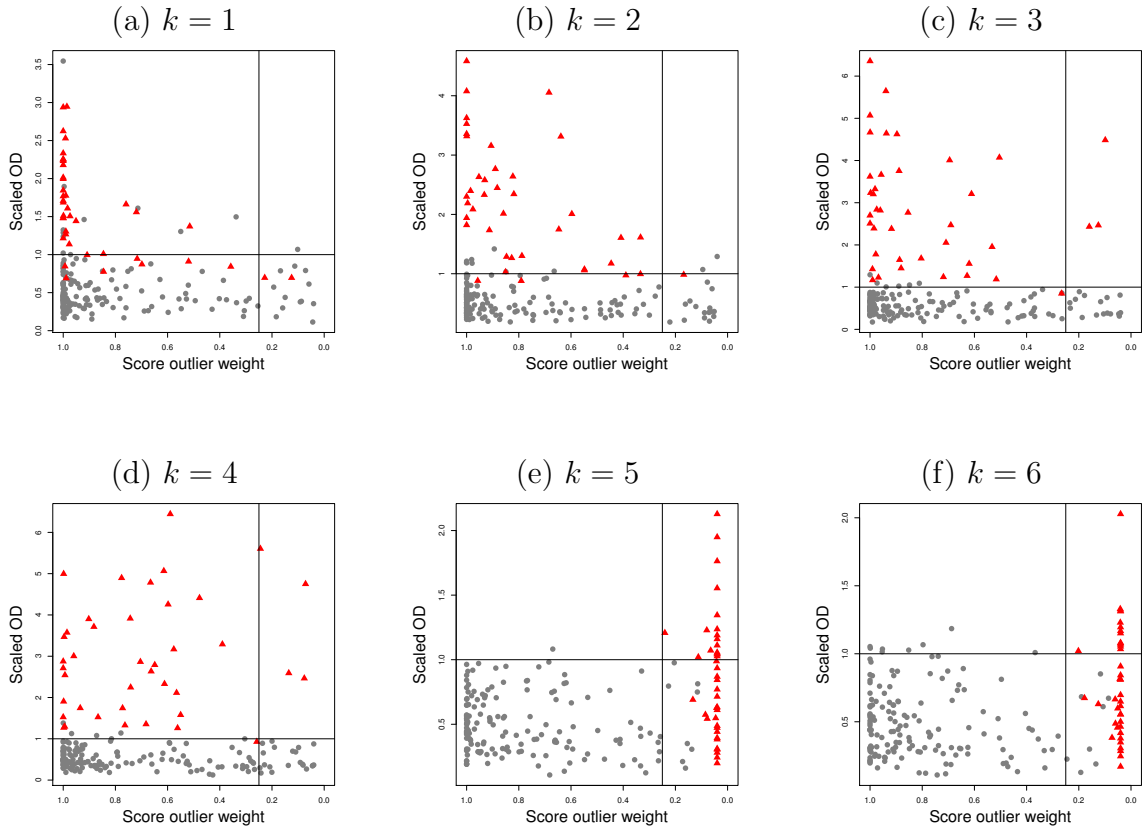

Figure S.34: Diagnostic plots given by estimating a 6-dimensional PC subspace sequentially using LTS-PCA together with the deflation method on the toy example: • represents the regular data, and ▲ represents the outliers.

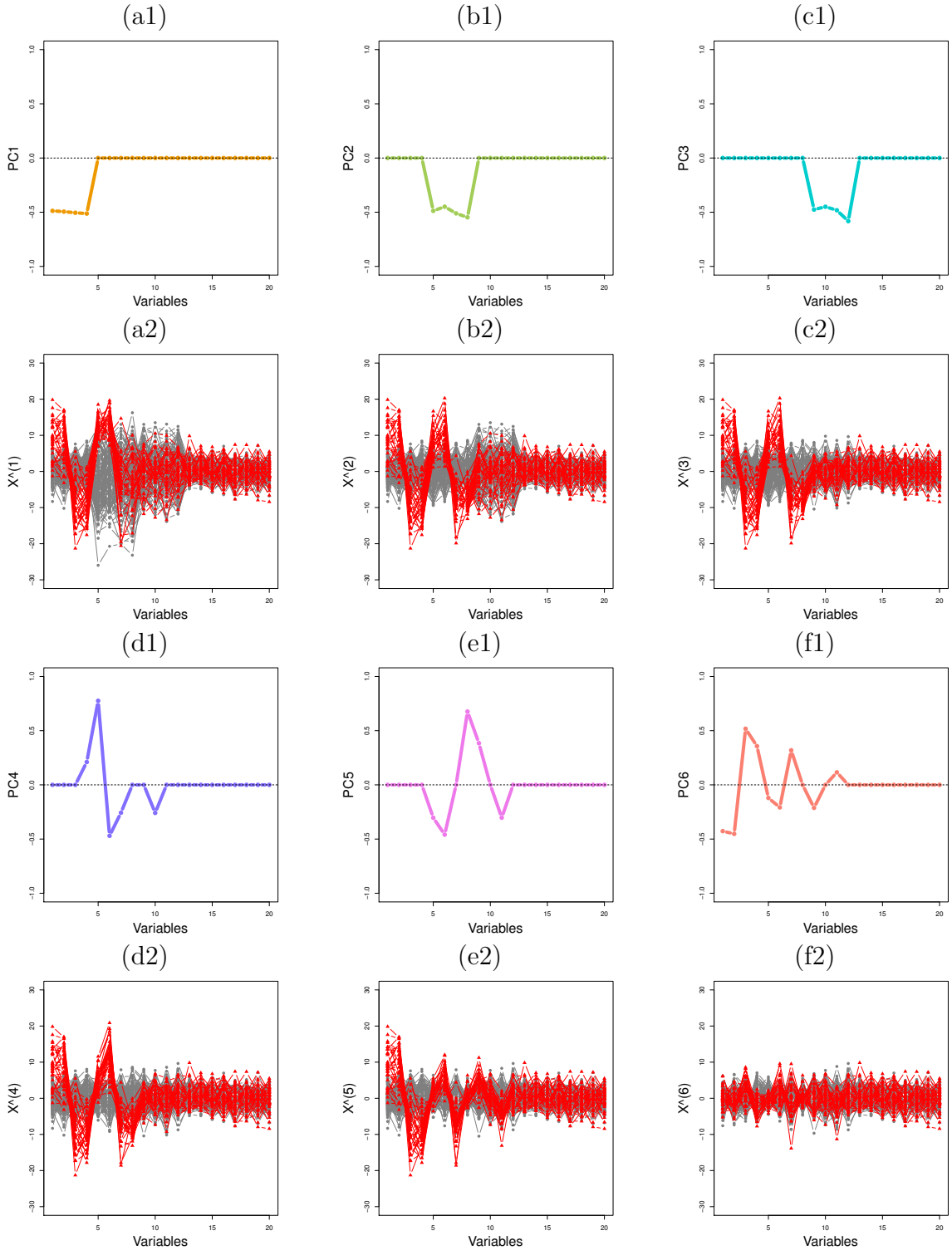

Figure S.35: Plots of deflated matrices and PC loadings when estimating 6 PCs sequentially using LTS-SPCA together with the deflation method on the toy example.

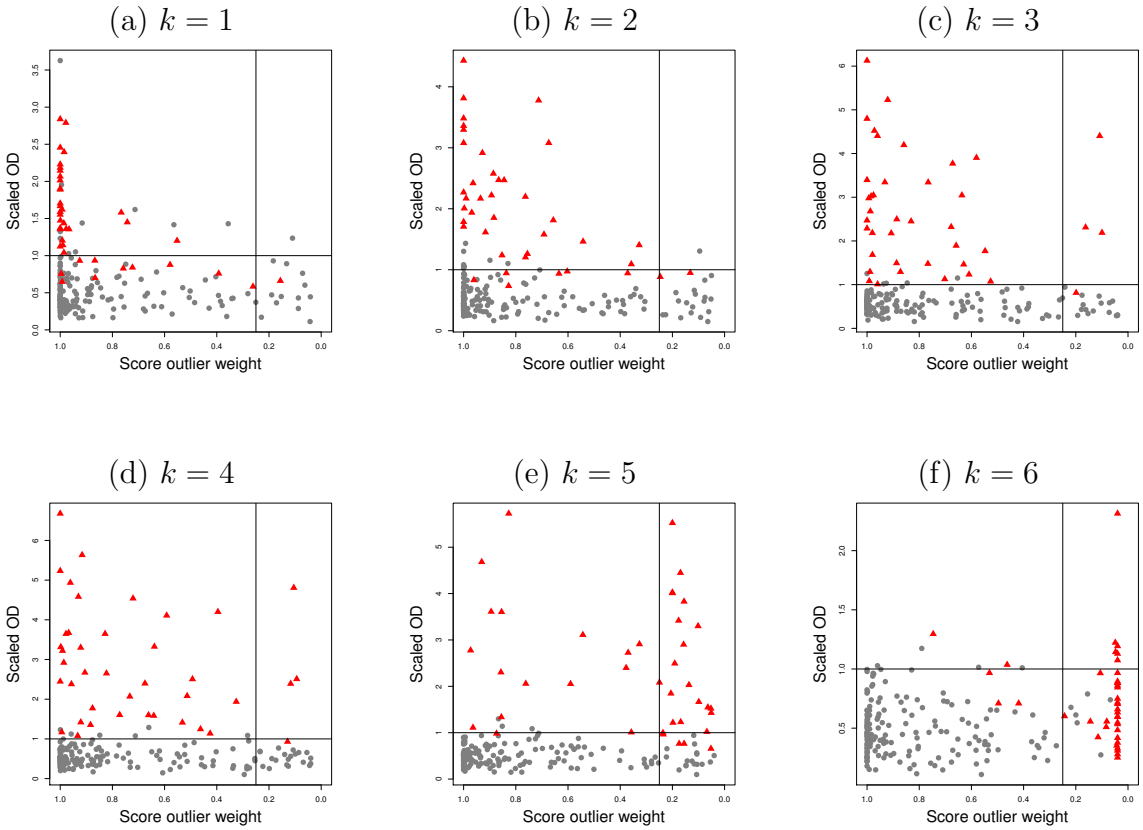

Figure S.36: Diagnostic plots given by estimating a 6-dimensional PC subspace sequentially using LTS-SPCA together with the deflation method on the toy example: • represents the regular data, and ▲ represents the outliers.

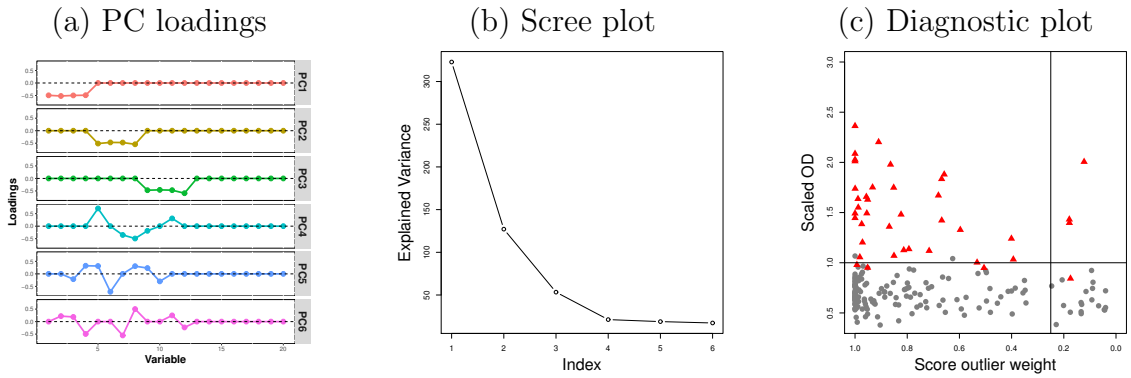

Figure S.37: (a) Plot of the PC loadings, (b) scree plot of the explained variance (c) diagnostic plot using Reweighted LTS-SPCA on the toy example.

Simulation study with 5 PCs

Here we consider a simulation study which contains 5 important PCs. The regular data are generated from a multivariate normal distribution with block diagonal covariance matrix plus additional noise as in the simulation study in the manuscript. Here, we take five blocks of correlated variables. The correlations between the variables within the five blocks are 0.9, 0.7, 0.8 0.6 and 0.5, respectively. The variances of the individual variables in the five blocks are 30^2, 25^2, 20^2, 15^2, and 10^2, respectively. Moreover, we take $n = 200$, $p = 2000$, $b = 20$. Then, we add $\varepsilon = 20\%$ or 40% of outliers. The outliers are generated according to setting 1 in the manuscript, which is the most difficult type of outliers. Hence, the first two variables in the second block are contaminated, i.e. $(\boldsymbol{X}_{\text{out}})_j \sim N(-\eta, 1)$ if $j = b+1$ or $b+2$, and $(\boldsymbol{X}_{\text{out}})_j = \boldsymbol{X}_j$ otherwise. Here, η varies from 25 to 500 with steps of length 25. We run LTS-SPCA with $k_{\max} = 5$ or 10 followed by a reweighting step, and then consider the first 5 PCs. As before, we denote by "LTS-SPCA" the initial LTS-SPCA without reweighting, and by "LTS-SPCA-RW5" or "LTS-SPCA-RW10" the reweighted LTS-SPCA using $k_{\max} = 5$ or $k_{\max} = 10$, respectively. The average angles between the estimated and true PC subspaces, and the FNs and FPs associated with each PC and the PC subspace over 200 replicates are reported in Figures S.38-S.42. Similar patterns can be observed as for the case of 2 PCs in the manuscript. The contamination in the important variables may lead to large false negatives for the PCs given by the initial LTS-SPCA. As long as the outliers remain detectable, reweighting based on a sufficient number of initial PCs significantly improves the subspace estimation. Moreover, it is obvious that the reweighting step can also largely reduce the number of false positives from the initially estimated PCs.

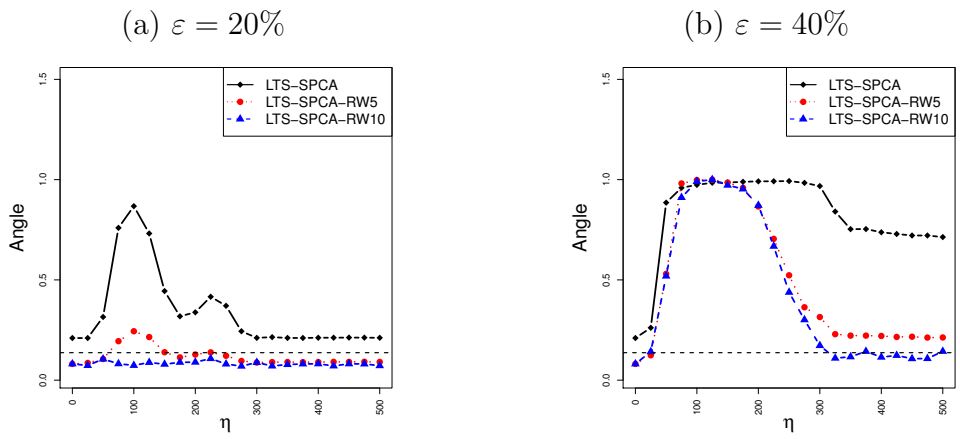

Figure S.38: Average angles for the simulation study with 5 PCs with $n = 200$, $p = 2000$, $b = 20$, $\varepsilon = 20\%$ and varying η. The dashed horizontal line is the result of sPCA-rSVD on clean data.

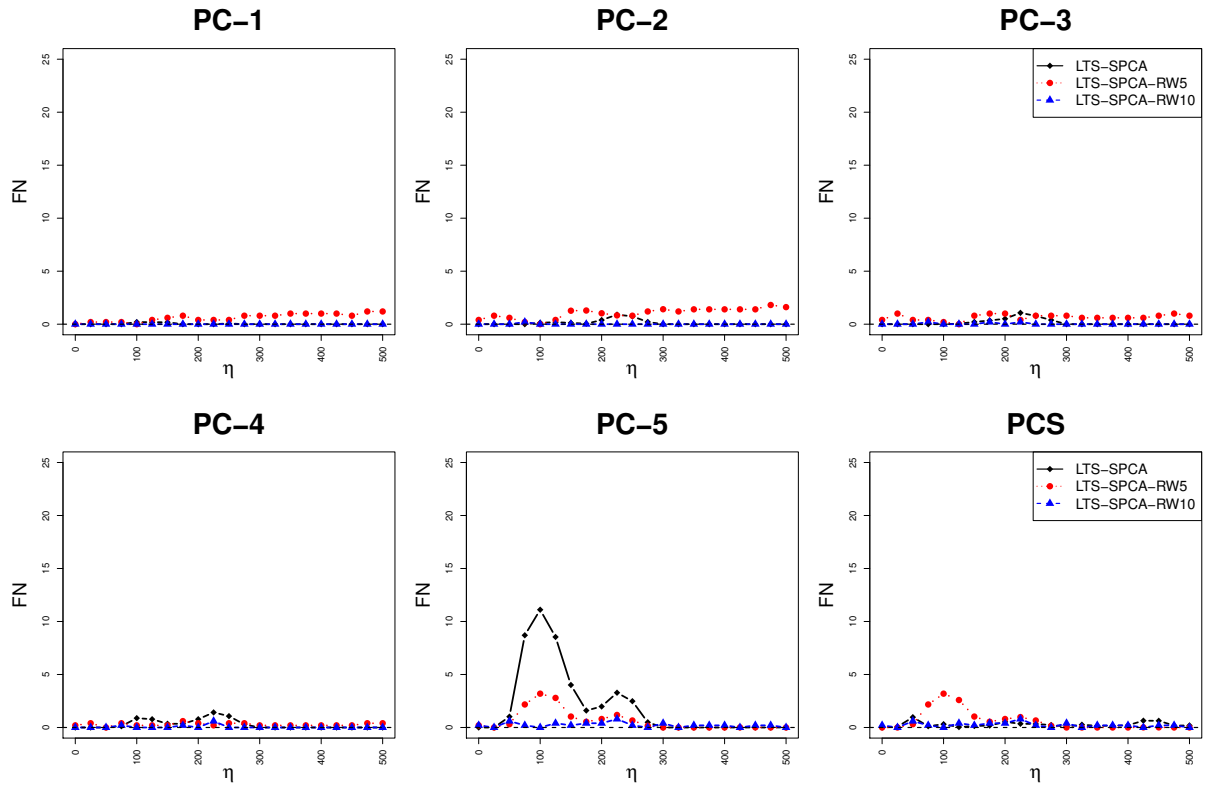

Figure S.39: Average false negatives (FN) for the simulation study with 5 PCs with $n = 200$, $p = 2000$, $b = 20$, $\varepsilon = 20\%$ and varying η. The dashed horizontal line is the result of sPCA-rSVD on clean data.

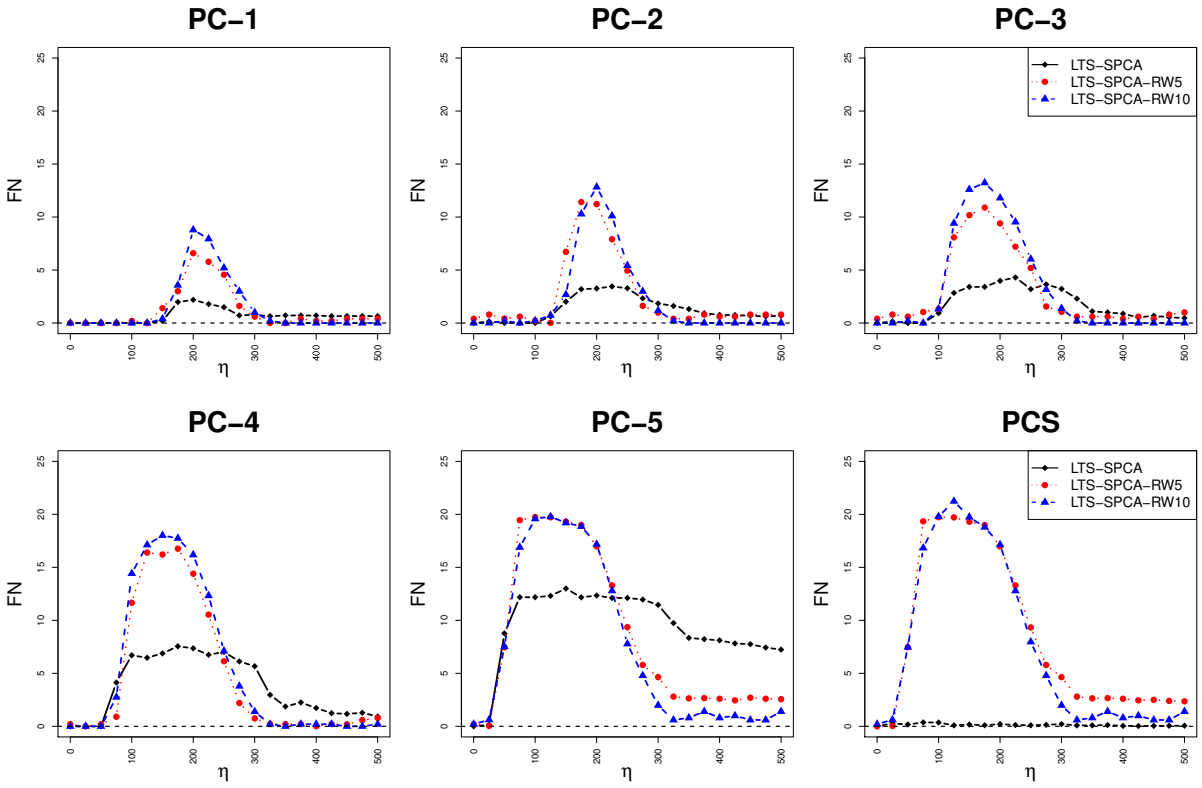

Figure S.40: Average false negatives (FN) for the simulation study with 5 PCs with $n = 200$, $p = 2000$, $b = 20$, $\varepsilon = 40\%$ and varying η. The dashed horizontal line is the result of sPCA-rSVD on clean data.

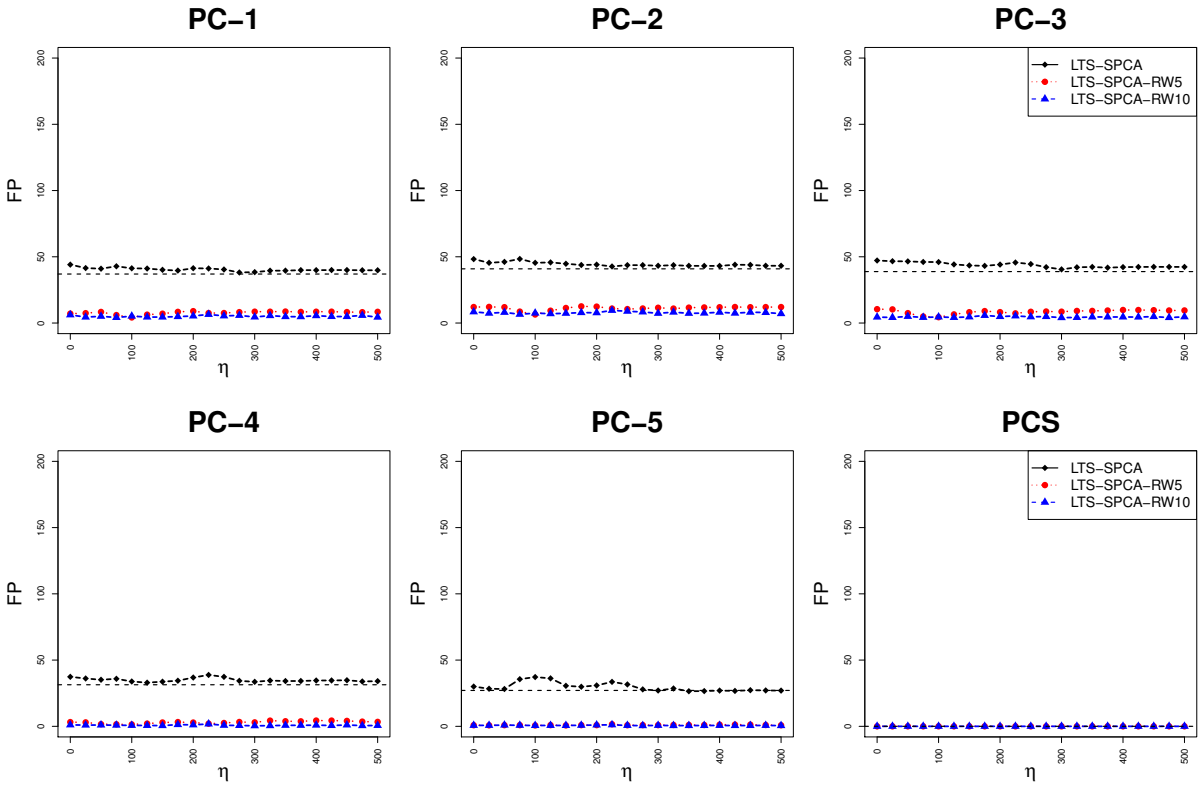

Figure S.41: Average false positives (FP) for the simulation study with 5 PCs with $n = 200$, $p = 2000$, $b = 20$, $\varepsilon = 20\%$ and varying η. The dashed horizontal line is the result of sPCA-rSVD on clean data.

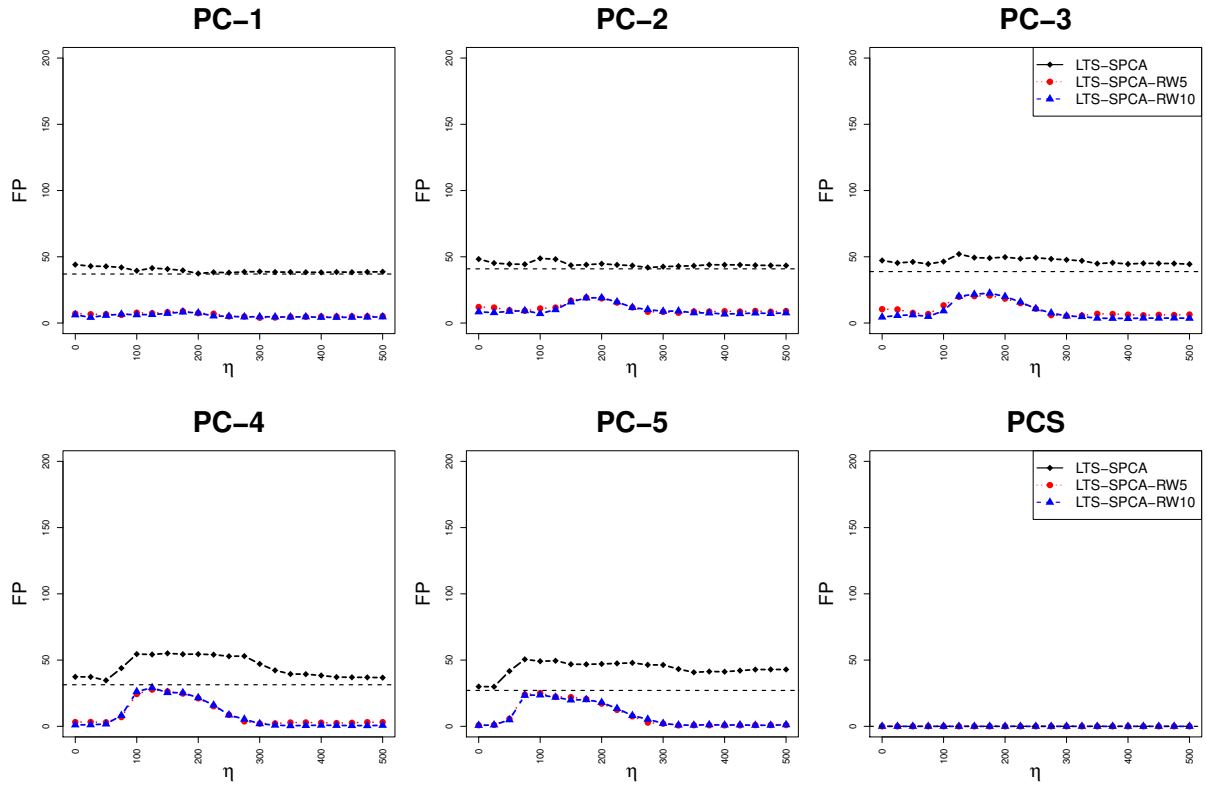

Figure S.42: Average false positives (FP) for the simulation study with 5 PCs with $n = 200$, $p = 2000$, $b = 20$, $\varepsilon = 40\%$ and varying η. The dashed horizontal line is the result of sPCA-rSVD on clean data.

References

Yuan, X. and Zhang, T. (2013). Truncated power method for sparse eigenvalue problems. *Journal of Machine Learning Research*, 14:899–925.