Electronic Journal of Statistics
Vol. 5 (2011) 1123–1160
ISSN: 1935-7524
DOI: 10.1214/11-EJS636

Two proposals for robust PCA using semidefinite programming

Michael McCoy[*] **and Joel A. Tropp**[*]

Computing & Mathematical Sciences, MC 305-16
California Institute of Technology
1200 E. California Blvd.
Pasadena, CA 91125
e-mail: mccoy@cms.caltech.edu; jtropp@cms.caltech.edu

Abstract: The performance of principal component analysis suffers badly in the presence of outliers. This paper proposes two novel approaches for robust principal component analysis based on semidefinite programming. The first method, *maximum mean absolute deviation rounding*, seeks directions of large spread in the data while damping the effect of outliers. The second method produces a *low-leverage decomposition* of the data that attempts to form a low-rank model for the data by separating out corrupted observations. This paper also presents efficient computational methods for solving these semidefinite programs. Numerical experiments confirm the value of these new techniques.

AMS 2000 subject classifications: Primary 60H25, 62G35; secondary 90C22.
Keywords and phrases: Robustness, principal component analysis, semidefinite relaxation, leverage, duality.

Received December 2010.

1. Introduction

Principal component analysis (PCA), proposed in 1933 by Hotelling [30], is a common technique for summarizing high-dimensional data. Principal components are designed to identify directions in which the observations vary most. As a consequence, PCA is often used to reduce the dimension of the data.

Statistics based on variance, such as principal components, are highly sensitive to outliers [53]. The literature on robust statistics contains a wide variety of techniques that attempt to correct this shortcoming [32]. Unfortunately, many of these approaches are based on intractable optimization problems or lack a principled foundation.

Our focus in this work is to develop new formulations for robust PCA that can be solved efficiently using convex programming algorithms. Our first proposal, which we call *maximum mean absolute deviation rounding* (MDR), exchanges

[*]This work has been supported in part by ONR awards N00014-08-1-0883 and N00014-11-1-0025, AFOSR award FA9550-09-1-0643, and a Sloan Fellowship. This research was performed in part while the authors were in residence at the Institute for Pure and Applied Mathematics at the University of California, Los Angeles.

the variance in the definition of PCA with a function less sensitive to outliers known as the mean absolute deviation. Although this formulation leads to a non-convex optimization problem, we demonstrate that it is possible to approximate the optimum by relaxing to a semidefinite program and randomly rounding the solution. This method can be viewed as the first provably good approximation algorithm for a nontrivial instance of projection-pursuit PCA [35].

Our second proposal uses a different semidefinite program to split the input data into the sum of a low-leverage matrix and a matrix of corrupted observations. We refer to this dissection as a *low-leverage decomposition* (LLD) of the data. This method is similar in spirit to the rank-sparsity decomposition of Chandrasekaran et al. [9]. While preparing this manuscript, we learned of an independent investigation into this formulation of robust PCA by Xu et al. [56, 57].

We describe algorithms that solve these semidefinite programs efficiently, and we provide numerical experiments that confirm the effectiveness of these new techniques. We begin with a brief overview of our proposals before laying out the details in Sections 2 and 3.

1.1. The data model

Suppose that we have a family $\{\boldsymbol{x}_i\}_{i=1}^n$ of n observations in p dimensions. We form an $n \times p$ data matrix $\boldsymbol{X}$ whose rows are the observations. We assume the observations are centered; that is, $\frac{1}{n}\sum_i \boldsymbol{x}_i \approx \boldsymbol{0}$. While our methods do not explicitly require centered data, this hypothesis allows us to interpret principal components as directions of high variance in the data. We discuss practical centering approaches in Section 5.

1.2. Maximizing the mean absolute deviation

Our first method is designed to mitigate a source of sensitivity in classical principal component analysis. The top principal component $\boldsymbol{v}_{\mathrm{PCA}}$ is defined as a direction of maximum variance in the data:

$$\boldsymbol{v}_{\mathrm{PCA}} = \operatorname*{arg\,max}_{\|\boldsymbol{v}\|_2=1} \sum_{i=1}^n |\langle \boldsymbol{x}_i, \boldsymbol{v}\rangle|^2 . \tag{1}$$

The squared inner products in (1) may give outlying points an outsized influence. Simply put, squaring a large number results in a huge number that may drag the principal component away from the bulk of the data. We can reduce this effect by replacing the squared inner product with a measure of spread that is less sensitive to deviations outside the bulk of the data. We propose the use of the absolute value of the inner product:

$$\boldsymbol{v}_{\mathrm{MD}} = \operatorname*{arg\,max}_{\|\boldsymbol{v}\|_2=1} \sum_{i=1}^n |\langle \boldsymbol{x}_i, \boldsymbol{v}\rangle| , \tag{2}$$

where we have added the subscript MD to indicate that we have exchanged the variance in equation (1) with a measure of spread known as the *mean absolute deviation* (MD) [32, p. 2].

This revision results in some complications. The formulation (1) is an eigenvector problem that can be solved efficiently. In contrast, it is NP-hard to compute $\boldsymbol{v}_{\mathrm{MD}}$ (see Section 2.3). Nevertheless, we develop an efficient randomized algorithm that provably computes an approximate solution to (2). We call this approach *maximum mean absolute deviation rounding* (MDR).

Our main result, Theorem 2.2, states that, for any failure probability $\delta > 0$ and loss factor $\varepsilon > 0$, our algorithm produces a unit-norm vector $\boldsymbol{v}_{\mathrm{MDR}}$ such that

$$\sum_{i=1}^{n} |\langle \boldsymbol{x}_i, \boldsymbol{v}_{\mathrm{MDR}} \rangle| \geq (1-\varepsilon) \sqrt{\frac{2}{\pi}} \max_{\|\boldsymbol{v}\|_2 = 1} \sum_{i=1}^{n} |\langle \boldsymbol{x}_i, \boldsymbol{v} \rangle| \,.$$

We find additional robust principal components by restricting the data to a subspace perpendicular to the previous components and solving (2) again, repeating this process as necessary.

The algorithm requires the solution to a semidefinite program whose size is polynomial in the number of observations. Since semidefinite programs are solvable in polynomial time using interior-point methods, our algorithm is theoretically tractable. In practice, solving semidefinite programs can be daunting even for moderately sized input data—say, more than 100 observations. To address this issue, we detail a technique of Burer and Monteiro [6, 7] that can usually solve our proposal efficiently. In Section 5 we provide some numerical evidence that this approach succeeds.

This proposal is not without precedent. A more general formulation appears in Huber's book [31, p. 203], and it is now known as *projection-pursuit PCA* (PP-PCA) [35]. The formulation (2) was rediscovered recently in [34], indicating contemporary interest in this approach. We provide further detail on PP-PCA in Section 2.2 and discuss the history of the method in Section 4.1.

1.3. A low-leverage decomposition

Our second proposal stems from a different interpretation of classical principal component analysis. Instead of viewing classical principal components as directions of maximum variance, we can view them as an optimal low-rank model for the data [8]. Suppose $\boldsymbol{P}_\star$ is a matrix that solves

$$\begin{array}{ll} \text{minimize} & \|\boldsymbol{X} - \boldsymbol{P}\|_{\mathrm{F}} \\ \text{subject to} & \operatorname{rank}(\boldsymbol{P}) = T. \end{array}$$

The dominant principal components of $\boldsymbol{X}$ are given by the T right singular vectors of $\boldsymbol{P}_\star$ corresponding to the largest T nonzero singular values of $\boldsymbol{P}_\star$.

With real data, one is often faced with the situation where entire observations are corrupted. Even when this is the case, we would still like to recover a low-rank model. A natural formulation for identifying a low-rank model is based on the well-known heuristics for rank sparsity [20] and group sparsity [47]. We propose to decompose the data matrix as $\boldsymbol{X} = \boldsymbol{P}_{\text{LLD}} + \boldsymbol{C}_{\text{LLD}}$ by solving the semidefinite program

$$\begin{array}{ll} \text{minimize} & \sum_i \sigma_i(\boldsymbol{P}) + \gamma \sum_j \|\boldsymbol{c}_j\|_2 \\ \text{subject to} & \boldsymbol{P} + \boldsymbol{C} = \boldsymbol{X}. \end{array} \tag{3}$$

Here, $\sigma_i(\boldsymbol{P})$ is the ith singular value of $\boldsymbol{P}$, and $\boldsymbol{c}_i$ is the ith row of $\boldsymbol{C}$.

We view the optimal matrix $\boldsymbol{P}_{\text{LLD}}$ as a surrogate for the low-rank approximation to the uncorrupted data, and the optimal matrix $\boldsymbol{C}_{\text{LLD}}$ as an approximation of the corrupted data. The formulation (3) has an interesting property even when $\boldsymbol{P}_{\text{LLD}}$ is not low-rank or $\boldsymbol{C}_{\text{LLD}}$ is not row-sparse: $\boldsymbol{P}_{\text{LLD}}$ is guaranteed to be a low-leverage set of observations in a sense we make precise in Section 3.1. As a result, we refer to $\boldsymbol{X} = \boldsymbol{P}_{\text{LLD}} + \boldsymbol{C}_{\text{LLD}}$ as a *low-leverage decomposition* (LLD) of the data. We define the LLD components as the right singular vectors of $\boldsymbol{P}_{\text{LLD}}$.

This optimization problem is similar to the rank-sparsity decomposition problem proposed in [9]; see also [8]. We discuss these ideas at more length in Section 4. As this manuscript was being prepared, we learned of an independent investigation of the program (3) for robust PCA by Xu et al. [56, 57] that provides conditions for recovery of the support of the corruption and the row-space of the uncorrupted observations.

1.4. Qualitative observations

Our experiments in Section 5 clearly demonstrate that our proposals are robust. Qualitatively, we find that MDR is less sensitive to outliers than PCA, while LLD is nearly insensitive to outliers. This suggests that MDR and LLD apply in different situations; we discuss this point and other conclusions in Section 6.

1.5. Roadmap

Sections 2 and 3 describe our proposals in more detail, including theoretical guarantees and practical algorithms. Section 4 offers an overview of previous work on robust PCA, while Section 5 gives numerical experiments illustrating the performance of our methods in various settings. We offer some conclusions in Section 6, and include a technical appendix with the proofs of some supporting results.

1.6. Notation

We work exclusively with real numbers. Bold capital letters refer to matrices while bold lower-case letters are vectors. We represent the ith row of a matrix $\boldsymbol{A}$ by $\boldsymbol{a}_i$ and the jth entry of a vector $\boldsymbol{a}$ by a_j. When clarity is required, the notation $[\boldsymbol{A}]_{ij}$ gives the (i, j)th matrix entry, while $[\boldsymbol{a}]_i$ gives the ith vector entry. The matrix $\boldsymbol{A}$ has adjoint $\boldsymbol{A}^*$ and Moore–Penrose pseudoinverse $\boldsymbol{A}^\dagger$.

Table 1
Summary of the norms used in this work

Norm	**Description**	**Description of Dual**
$\|\boldsymbol{A}\|_{2\to 2}$	Maximum singular value of $\boldsymbol{A}$	Sum of the singular values of $\boldsymbol{A}$
$\|\boldsymbol{A}\|_{2\to \infty}$	Maximum ℓ_2 row norm of $\boldsymbol{A}$	Sum of the ℓ_2 row norms of $\boldsymbol{A}$
$\|\boldsymbol{A}\|_{1\to \infty}$	Maximum absolute entry of $\boldsymbol{A}$	Sum of the absolute entries of $\boldsymbol{A}$

We use the compact convention for the singular value decomposition (SVD) of a matrix: when $\boldsymbol{A}$ has rank r, we write its SVD as $\boldsymbol{A} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^*$, where $\boldsymbol{U}$ and $\boldsymbol{V}$ have orthonormal columns, and $\boldsymbol{\Sigma}$ is a nonsingular diagonal matrix whose entries are positive and are arranged in weakly decreasing order. The notation $\boldsymbol{A} \succcurlyeq \boldsymbol{B}$ denotes that the difference $\boldsymbol{A} - \boldsymbol{B}$ is positive semidefinite.

We use ∂ for the subgradient map of a convex function. For background on subgradients, and convex analysis in general, we refer to the book [48].

The symbols $\mathbb{P}$ and $\mathbb{E}$ denote probability and expectation, respectively.

1.6.1. Norms

For $1 \le p < \infty$, the ℓ_p norm of $\boldsymbol{u}$ is $\|\boldsymbol{u}\|_p = \left(\sum_i |u_i|^p\right)^{1/p}$, while the ℓ_∞ norm of $\boldsymbol{u}$ is $\|\boldsymbol{u}\|_\infty = \max_i |u_i|$. The Frobenius norm of a matrix is defined by $\|\boldsymbol{A}\|_{\mathrm{F}}^2 = \langle \boldsymbol{A}, \boldsymbol{A}\rangle$, where $\langle\cdot,\cdot\rangle$ represents the standard inner product.

We define the $\ell_p \to \ell_q$ operator norm and its dual respectively by

$$\|\boldsymbol{A}\|_{p\to q} = \sup_{\|\boldsymbol{u}\|_p = 1} \|\boldsymbol{A}\boldsymbol{u}\|_q, \quad \text{and} \quad \|\boldsymbol{B}\|_{p\to q}^* = \sup_{\|\boldsymbol{A}\|_{p\to q}=1} \langle \boldsymbol{B}, \boldsymbol{A}\rangle .$$

Table 1 describes some of the specific operator norms used in this work. We also use the norms $\|\boldsymbol{A}\|_{2\to 1}$ and $\|\boldsymbol{A}\|_{\infty\to 1}$ and their duals, which lack such simple descriptions; see Sections 2.3 and 2.4.

2. Maximum mean absolute deviation rounding

Our first method is based on the classical interpretation of the top principal component as the direction of maximum empirical variance in multidimensional data. It is well known that the variance is highly sensitive to outliers in the data [53]. The field of robust statistics has reacted by developing and analyzing robust measures of spread known as robust scales; see [32, Ch. 5] or [40, Sec. 2.5]. This literature describes a generic method for determining robust principal components by replacing the variance with a robust measure of scale. Li and Chen [35] published the first investigation of this proposal under the name *projection-pursuit PCA* (PP-PCA). Our proposal is a specific instance of PP-PCA with the mean absolute deviation scale (4). We show that this formulation is computationally intractable, but we develop an algorithm that provably approximates its solution. To our knowledge, this is the first rigorous algorithm for PP-PCA with a robust scale.

2.1. Scales

A *scale* is a function that measures the spread of one-dimensional data [32, Ch. 5]. The most common scale is the empirical standard deviation, defined[1] as

$$\operatorname{std}(\boldsymbol{y}) = \left(\sum_i y_i^2 \right)^{1/2} = \|\boldsymbol{y}\|_2,$$

where we assume the data $\boldsymbol{y}$ is centered. Of course, the standard deviation is not the only way to measure the spread of the data. An alternative proposal [32, p. 2] is the *mean absolute deviation* (MD). For centered data $\boldsymbol{y}$, the MD scale is defined as

$$\operatorname{MD}(\boldsymbol{y}) = \sum_i |y_i| = \|\boldsymbol{y}\|_1. \tag{4}$$

More generally, a scale is a function $S : \mathbb{R}^n \to \mathbb{R}$ such that $S(\alpha\boldsymbol{y}) = |\alpha|\, S(\boldsymbol{y})$. Scales are typically chosen so that they are less sensitive to outliers than the standard deviation. The robust statistics literature focuses on scales that have a positive breakdown point: the value of the scale cannot be arbitrarily corrupted by nefariously chosen observations, so long as the fraction of bad observations in the entire data set is small. Although the mean absolute deviation has a breakdown point of zero, it exhibits more efficient behavior than the standard deviation under contaminated distributions [53].

2.1.1. Scales for multivariate data

We extend the definition of scales to multivariate observations by considering the scale of the data in a given direction. For a unit Euclidean norm vector $\boldsymbol{u}$, the entries of the product $\boldsymbol{X}\boldsymbol{u}$ give the projections of the rows of $\boldsymbol{X}$ onto the direction $\boldsymbol{u}$. Note that if $\boldsymbol{X}$ is centered in the sense of Section 1.1, then the projection $\boldsymbol{X}\boldsymbol{u}$ is also centered by linearity.

We then define the scale of $\boldsymbol{X}$ in the direction $\boldsymbol{u}$ to be the scale of the projected data $S(\boldsymbol{X}\boldsymbol{u})$. As noted in [31], this definition is equivariant under an orthogonal change of basis: for any orthogonal matrix $\boldsymbol{Q}$, the scale of $\boldsymbol{X}$ in the direction $\boldsymbol{u}$ is equal to the scale of $\boldsymbol{X}\boldsymbol{Q}^*$ in the direction $\boldsymbol{Q}\boldsymbol{u}$.

2.2. Projection-pursuit PCA

Classically, the top principal component is defined as the direction where the empirical standard deviation in the data is largest:

$$\boldsymbol{v}_{\text{PCA}} = \underset{\|\boldsymbol{v}\|_2 = 1}{\arg\max}\ \operatorname{std}(\boldsymbol{X}\boldsymbol{v}). \tag{5}$$

[1] One usually defines scales so that they are unbiased estimates of the sample standard deviation when the data is drawn from a normal distribution. We are interested in the direction of maximal scale, not the value of the scale in this direction, so we can safely ignore the normalization factor.

One natural approach for finding robust components replaces the standard deviation in (5) with a robust scale $S(\cdot)$. The robust component is then defined as the direction of maximum robust scale

$$\boldsymbol{v}_{\mathrm{PP}} = \operatorname*{arg\,max}_{\|\boldsymbol{v}\|_2=1} S(\boldsymbol{X}\boldsymbol{v}).$$

We define further robust components inductively by adding orthogonality constraints:

$$\boldsymbol{v}_{\mathrm{PP}}^{(k)} = \operatorname*{arg\,max}_{\substack{\|\boldsymbol{v}\|_2=1 \\ \boldsymbol{v}\perp\boldsymbol{v}_{\mathrm{PP}}^{(j)} \text{ for } j<k}} S(\boldsymbol{X}\boldsymbol{v}). \tag{6}$$

This greedy method for constructing orthogonal components based on robust scales goes by the name projection-pursuit PCA. While this scheme was originally proposed by Huber [31, p. 203], it was first studied in detail by Li and Chen [35]. The fact that PP-PCA reduces to PCA when the scale is given by the standard deviation is a consequence of the variational characterization of eigenvectors by Courant and Fischer.

As we discuss in Section 2.6.1, the orthogonal restriction step is straightforward to implement using standard techniques from numerical linear algebra. Therefore, the crux of any PP-PCA method involves the computation of a single direction of maximum scale. As we will see immediately, this computation is a rather thorny issue in the case of the MD scale.

2.3. PP-PCA with the MD scale is NP-hard

Finding the top classical principal component is an eigenvector problem that amounts to computing the direction where the $\ell_2 \to \ell_2$ norm is achieved. Similarly, PP-PCA with the MD scale amounts to finding a vector that achieves an operator norm. Indeed, the problem $\boldsymbol{v}_{\mathrm{MD}} = \arg\max_{\|\boldsymbol{v}\|_2=1} \|\boldsymbol{X}\boldsymbol{v}\|_1$ is equivalent to the problem

$$\text{Find } \boldsymbol{v}_{\mathrm{MD}} \text{ such that } \|\boldsymbol{X}\boldsymbol{v}_{\mathrm{MD}}\|_1 = \|\boldsymbol{X}\|_{2\to1} \text{ and } \|\boldsymbol{v}_{\mathrm{MD}}\|_2 = 1. \tag{7}$$

Unfortunately, exchanging the ℓ_2 norm for the ℓ_1 norm leads to an NP-hard computational problem. To see this, we require the following result, which we establish in the Appendix.

Fact 2.1. *For each matrix $\boldsymbol{X}$, the identity $\|\boldsymbol{X}\|_{2\to1}^2 = \|\boldsymbol{X}\boldsymbol{X}^*\|_{\infty\to1}$ holds.*

Rohn [49] shows that there exists a class of positive matrices $\mathcal{M}$ such that the existence of a polynomial-time algorithm for accurately computing $\|\boldsymbol{M}\|_{\infty\to1}$ for all $\boldsymbol{M} \in \mathcal{M}$ implies $\mathsf{P} = \mathsf{NP}$. Since we can factor positive matrices $\boldsymbol{M} = \boldsymbol{R}\boldsymbol{R}^*$ in polynomial time using, for example, a Cholesky factorization, the existence of an accurate polynomial-time algorithm that computes $\|\boldsymbol{R}\|_{2\to1}^2$ for any matrix $\boldsymbol{R}$ implies that $\mathsf{P} = \mathsf{NP}$.

The observation that Equation (6) is NP-hard to solve for the specific choice $S(\cdot) = \|\cdot\|_1$ has grave implications for existing PP-PCA algorithms. The algorithms available in the literature for PP-PCA [11, 13, 35] are general schemes that operate with any choice of scale S. As a result, it is unlikely that these algorithms provide both accurate and efficient solutions to the PP-PCA problem. This issue is not merely theoretical because these algorithms tend to perform poorly in practice. We discuss this point further in Section 4.1, and refer the interested reader to a comparative study of Maronna [39].

2.4. Approximating the $\ell_2 \to \ell_1$ norm using randomized rounding

Although it is NP-hard to compute the $\ell_2 \to \ell_1$ norm, it is possible to approximate its value efficiently. This fact is a consequence of the little Grothendieck theorem [46, Sec. 5b], but our procedure depends on ideas of Nesterov [44] and a new factorization step.

2.4.1. The semidefinite relaxation of the $\ell_2 \to \ell_1$ norm

Before describing our algorithm, we show how the computation of $\ell_2 \to \ell_1$ operator norm can be relaxed to a semidefinite program. First, apply Fact 2.1 to change the computation of the $\ell_2 \to \ell_1$ norm to the computation of the $\ell_\infty \to \ell_1$ norm:

$$\|\boldsymbol{X}\|_{2\to 1}^2 = \|\boldsymbol{X}\boldsymbol{X}^*\|_{\infty\to 1} = \max_{\|\boldsymbol{y}\|_\infty = 1} \boldsymbol{y}^*\boldsymbol{X}\boldsymbol{X}^*\boldsymbol{y}. \tag{8}$$

The second identity above follows from the proof of Fact 2.1; see also [49, Prop. 1]. Interpreting the quadratic form on the right hand side of (8) as a trace implies that $\|\boldsymbol{X}\|_{2\to 1}^2$ is the optimal value of the (nonconvex) program

$$\begin{array}{ll} \text{maximize} & \operatorname{trace}(\boldsymbol{X}\boldsymbol{X}^*\boldsymbol{Z}) \\ \text{subject to} & \boldsymbol{Z} = \boldsymbol{y}\boldsymbol{y}^*, \quad [\boldsymbol{Z}]_{ii} = 1 \text{ for all } i. \end{array}$$

Relaxing the rank one constraint $\boldsymbol{Z} = \boldsymbol{y}\boldsymbol{y}^*$ to the positive-semidefinite constraint $\boldsymbol{Z} \succcurlyeq \mathbf{0}$ leads to the semidefinite program

$$\begin{array}{ll} \text{maximize} & \operatorname{trace}(\boldsymbol{X}\boldsymbol{X}^*\boldsymbol{Z}) \\ \text{subject to} & \boldsymbol{Z} \succcurlyeq \mathbf{0}, \quad [\boldsymbol{Z}]_{ii} = 1 \text{ for all } i. \end{array} \tag{9}$$

It follows that $\|\boldsymbol{X}\|_{2\to 1} \le \alpha_\star$, where $\alpha_\star^2$ is the optimal value of (9). Moreover, Grothendieck's inequality for positive-semidefinite matrices implies that

$$\alpha_\star^2 \le \frac{\pi}{2}\|\boldsymbol{X}\boldsymbol{X}^*\|_{\infty\to 1}, \tag{10}$$

where this inequality is asymptotically the best possible [2, Sec. 4.2]. Thus, $\alpha_\star$ is within a factor of $\sqrt{\pi/2} < 1.26$ of the true value of the norm $\|\boldsymbol{X}\|_{2\to 1}$.

2.5. The MDR algorithm

The fact that equation (9) gives us a good upper bound on the *value* of $\|\boldsymbol{X}\|_{2\to1}$ is of secondary importance. Rather, we seek an approximation for $\boldsymbol{v}_{\mathrm{MD}}$ defined in (7), that is, we desire a vector $\boldsymbol{v}_\star$ with $\|\boldsymbol{v}_\star\|_2 = 1$ and $\|\boldsymbol{X}\boldsymbol{v}_\star\| \approx \|\boldsymbol{X}\|_{2\to1}$. We accomplish this goal via a randomized procedure that rounds an optimal solution $\boldsymbol{Z}_\star$ of (9) back to a vector $\boldsymbol{v}_\star$. The entire procedure is detailed in Algorithm 1.

The first step of the algorithm solves the semidefinite relaxation (9), and we factor the optimal matrix in Step 2. In Step 3(a), we use a Gaussian rounding procedure determine a random sign vector $\boldsymbol{y} \in \{\pm1\}^n$ such that the equation $\mathbb{E}\,\|\boldsymbol{X}\boldsymbol{X}^*\boldsymbol{y}\|_1 = 2\alpha_\star^2/\pi$ holds. This procedure is well understood [44]. The method in Step 3(b) that we use to compute $\boldsymbol{v}$ from $\boldsymbol{y}$ is novel; the proof of correctness appears in the appendix. By choosing the best random outcome, Step 4 controls the probability that our method fails to provide a reasonable approximation.

The following theorem describes the behavior of Algorithm 1.

Theorem 2.2. *Suppose that $\boldsymbol{X}$ is an $n \times p$ matrix, and let K be the number of rounding trials. Let $(\boldsymbol{v}_\star, \alpha_\star)$ be the output of Algorithm 1. Then $\alpha_\star \geq \|\boldsymbol{X}\|_{2\to1}$. Moreover, for $\theta < 1$, the inequality*

$$\|\boldsymbol{X}\boldsymbol{v}_\star\|_1 > \theta\sqrt{\frac{2}{\pi}}\alpha_\star$$

holds except with probability $\mathrm{e}^{-2K(1-\theta^2)/\pi}$.

In Theorem 2.2, it may be more natural to specify a failure probability $\delta > 0$ and approximation loss $\varepsilon = 1 - \theta > 0$ instead of a repetition number K. In this case, simple algebra shows that $\|\boldsymbol{X}\boldsymbol{v}_\star\|_1 > (1-\varepsilon)\sqrt{2/\pi}\,\|\boldsymbol{X}\|_{2\to1}$ except with

```
Algorithm 1: Maximum Mean Absolute Deviation Rounding
INPUT: An $n \times p$ matrix $\boldsymbol{X}$; repetition count $K$.
OUTPUT: A $p \times 1$ unit Euclidean norm vector $\boldsymbol{v}_\star$ and an optimal value $\alpha_\star$.
  1. Find $\boldsymbol{Z}_\star$ that solves the semidefinite program

         maximize     $\mathrm{trace}(\boldsymbol{X}\boldsymbol{X}^*\boldsymbol{Z})$
         subject to   $\boldsymbol{Z} \succcurlyeq \boldsymbol{0}, \quad [\boldsymbol{Z}]_{ii} = 1$ for $i = 1, \dots, n$      (11)

     Set $\alpha_\star$ to be the square root of the optimal value: $\alpha_\star = \sqrt{\mathrm{trace}(\boldsymbol{X}\boldsymbol{X}^*\boldsymbol{Z}_\star)}$.
  2. Factor $\boldsymbol{Z}_\star = \boldsymbol{R}_\star\boldsymbol{R}_\star^*$.
  3. For each $k = 1, \dots, K$, do
     (a) Set $\boldsymbol{y}^{(k)} = \mathrm{sgn}(\boldsymbol{R}_\star\boldsymbol{g}^{(k)})$, where $\boldsymbol{g}^{(k)}$ is an $n \times 1$ standard normal random
         vector.
     (b) Set $\boldsymbol{v}^{(k)} = \boldsymbol{X}^*\boldsymbol{y}^{(k)}/\|\boldsymbol{X}^*\boldsymbol{y}^{(k)}\|_2$.
  4. Set $\boldsymbol{v}_\star = \arg\max_{k=1,\dots,K} \|\boldsymbol{X}\boldsymbol{v}^{(k)}\|_1$.
```

probability δ, so long as

$$K \geq \frac{\pi}{2} \cdot \frac{\log(1/\delta)}{\varepsilon(2-\varepsilon)} = \mathcal{O}\left(\varepsilon^{-1} \log(\delta^{-1})\right).$$

In particular, we achieve $\|\boldsymbol{X}\boldsymbol{v}_\star\|_1 > 0.75\|\boldsymbol{X}\|_{2\to1}$ with probability at least 0.999 by the choice $K = 94$.

2.5.1. Implications of approximation guarantees

The approximation guarantee of Theorem 2.2 shows that, with high probability, the unit vector $\boldsymbol{v}_\star$ nearly maximizes the MD scale over the unit sphere. It does not, however, guarantee[2] that the vector $\boldsymbol{v}_\star$ is close (in, say, the Euclidean norm) to a unit vector that achieves $\|\boldsymbol{X}\|_{2\to1}^2$. Put another way, Theorem 2.2 ensures that our component $\boldsymbol{v}_\star$ achieves a near-optimal direction for the MD scale, but it says nothing about the orientation of this near-optimal direction with respect to some truly optimal direction.

This distinction should not be unnerving: our original motivation was the identification of a direction with large spread in the MD scale, and this is precisely the assurance Theorem 2.2 provides.

Our algorithm can provide even stronger approximation bounds in some instances. By virtue of the relaxation, the constant $\alpha_\star$ generated by an instance of Algorithm 1 is an upper bound on the norm $\|\boldsymbol{X}\|_{2\to1}$. Therefore, we use the approximation ratio

$$\rho := \frac{1}{\alpha_\star}\|\boldsymbol{X}\boldsymbol{v}_\star\|_1 \leq \frac{\|\boldsymbol{X}\boldsymbol{v}_\star\|_1}{\|\boldsymbol{X}\boldsymbol{v}_{\mathrm{MD}}\|_1}$$

to measure the quality of the optimal solution in Section 5.

We clearly must have $\rho \leq 1$, but Theorem 2.2 only guarantees that we achieve $\rho \approx 0.79$ with high probability. In practice, however, we typically see $\rho > 0.95$, which certifies that $\boldsymbol{v}_\star$ is a very good proxy for the direction of maximum deviation in the observations. This empirical result does not indicate that the analysis of Algorithm 1 is loose: it follows directly from [2, Sec. 4.2] that this bound is asymptotically tight for a class of examples as $n \to \infty$.

2.6. Implementation of Algorithm 1

For a fixed iteration count K, the complexity of Algorithm 1 is typically dominated by Step 1. Modern interior-point methods applied to (11) are guaranteed to compute the optimal objective value $\alpha_\star$ and optimal point $\boldsymbol{Z}_\star$ accurately in polynomial time. The factor $\boldsymbol{R}_\star$ may then be determined using a Cholesky factorization of $\boldsymbol{Z}_\star$. In practice, interior-point methods are very slow for large-scale problems, so we prefer an algorithm of Burer and Monteiro [6, 7].

[2] There is an exception to this rule. If the optimal matrix $\boldsymbol{Z}_\star$ generated by (11) has rank one, it is easy to check that the output $\boldsymbol{v}_\star$ of Algorithm 1 satisfies $\|\boldsymbol{X}\boldsymbol{v}_\star\|_1 = \alpha_\star$ with probability one—that is, we have solved the PP-PCA problem exactly. This feature is typical of schemes that involve a semidefinite relaxation of a rank-one constraint.

The algorithm of Burer and Monteiro never forms the semidefinite matrix $\boldsymbol{Z}$; rather it operates directly with the Cholesky factor $\boldsymbol{R}$. We express the objective function of (11) in terms of $\boldsymbol{R}$ as $\operatorname{trace}(\boldsymbol{X}\boldsymbol{X}^*\boldsymbol{R}\boldsymbol{R}^*) = \|\boldsymbol{X}^*\boldsymbol{R}\|_{\mathrm{F}}^2$. The constraints $[\boldsymbol{Z}]_{ii} = 1$ are equivalent to constraints on the rows of $\boldsymbol{R}$ of the form $\|\boldsymbol{r}_i\|_2 = 1$.

We implicitly enforce these row constraints by incorporating them into the objective function (cf. [6, Sec. 4.2]). The resulting unconstrained, nonconvex optimization problem takes the form

$$\operatorname{maximize}_{\boldsymbol{R}} \|\boldsymbol{X}^*\mathcal{N}(\boldsymbol{R})\|_{\mathrm{F}}^2, \tag{12}$$

where $\mathcal{N}(\boldsymbol{R})$ denotes the operator that normalizes the rows of $\boldsymbol{R}$, that is, $[\mathcal{N}(\boldsymbol{R})]_{ij} = [\boldsymbol{r}_i]_j / \|\boldsymbol{r}_i\|_2$.

We then apply a conjugate gradient algorithm to maximize the unconstrained objective in (12). Our particular implementation uses the algorithm of Hager and Zhang [29], which we have found to work well in our experiments. We refer to our online code for the choice of parameters in this conjugate gradient algorithm [41].

This factorization technique for solving (11) is advantageous in part because it reduces the dimension of the problem. The paper [7] shows that restricting $\boldsymbol{R}$ to be an $n \times k$ matrix for $k = \mathcal{O}(\sqrt{n})$ suffices to solve this problem exactly. To be precise, when $k = \lfloor (1 + \sqrt{9 + 8n})/2 \rfloor$ any *local* minimum $\boldsymbol{R}_\star \in \mathbb{R}^{n \times k}$ of (12) gives a *global* minimum $\boldsymbol{Z}_\star$ of (11) via the map $\boldsymbol{Z}_\star = \boldsymbol{R}_\star \boldsymbol{R}_\star^*$, provided a mild technical condition[3] holds.

2.6.1. *Orthogonal restriction*

Algorithm 1 approximates only the first principal component in (2). In order to approximate the kth robust principal component for $k > 1$, we define a new matrix $\boldsymbol{X}_k$ by restricting the rows of $\boldsymbol{X}$ to the subspace perpendicular to the span of $\boldsymbol{v}_1, \dots, \boldsymbol{v}_{k-1}$. Ignoring numerical stability issues, we can inductively define

$$\boldsymbol{X}_k = \boldsymbol{X}_{k-1} - \boldsymbol{X}\boldsymbol{v}_{k-1}\boldsymbol{v}_{k-1}^* = \boldsymbol{X}\left(\mathbf{I} - \sum_{j=1}^{k-1} \boldsymbol{v}_j \boldsymbol{v}_j^*\right), \tag{13}$$

which ensures each row of $\boldsymbol{X}$ is orthogonal to the previous components $\boldsymbol{v}_j$ for $j < k$. We then apply Algorithm 1 to the restricted matrix $\boldsymbol{X}_k$ to produce the component $\boldsymbol{v}_k$. Since the output $\boldsymbol{v}_\star$ of Algorithm 1 is a linear combination of the rows of the input matrix by Step 3(b), this iterative procedure ensures that $\boldsymbol{v}_k$ is perpendicular to the previous components.

In practice, we implement the orthogonal restriction using Householder reflectors as in [13]; see [52] for further background on the implementation of Householder transformations. Householder reflectors are more numerically stable than the naïve method (13).

[3] Specifically, the objective function $\operatorname{trace}(\boldsymbol{X}\boldsymbol{X}^*\boldsymbol{Z})$ must not be constant along a face of the feasible set.

2.7. Extending the rounding to multiple components

We have also attempted to extract a collection of robust components simultaneously by solving a single semidefinite program. That is, we would like to simultaneously solve for T components by optimizing

$$\begin{array}{ll} \text{maximize} & \sum_{i=1}^{T} \|\boldsymbol{X}\boldsymbol{v}_i\|_1 \\ \text{subject to} & \langle \boldsymbol{v}_i, \boldsymbol{v}_j \rangle = \delta_{ij} \end{array}, \tag{14}$$

where δ_{ij} is the Kronecker delta function. When $T = 1$, equation (14) is equivalent to equation (2). When $T > 1$, the restriction $\langle \boldsymbol{v}_i, \boldsymbol{v}_j \rangle = \delta_{ij}$ ensures that the optimum occurs at an orthogonal set of unit vectors.

We rephrase this optimization problem as the equivalent quadratically constrained quadratic program

$$\begin{array}{ll} \text{maximize} & \sum_{i=1}^{n} \boldsymbol{w}_i^* \boldsymbol{X}\boldsymbol{v}_i \\ \text{subject to} & \operatorname{diag}(\boldsymbol{w}_i \boldsymbol{w}_i^*) = 1, \quad \langle \boldsymbol{v}_i, \boldsymbol{v}_j \rangle = \delta_{ij} \end{array} \tag{15}$$

The diagonal restrictions on $\boldsymbol{w}_i$ ensure that $\boldsymbol{w}_i \in \{\pm 1\}^n$ for each $i = 1, \dots, n$. The nonconvex problem (15) can be approximated via a semidefinite relaxation and rounding scheme proposed in [43]. The results of [51] imply that the optimal value of this relaxation is guaranteed to be larger than the optimal value of (14) by no more than a logarithmic factor. As the rounding procedure in [43] does not produce orthogonal vectors, we apply an additional orthogonalization step so that the output is feasible for (14). Empirically, we have found that the orthogonalization increases the objective value over the standard rounding, so it appears that there is no loss in applying a naïve orthogonalization procedure.

Unfortunately, this method does not appear to be competitive with the projection pursuit method. The vectors we find by coupling Algorithm 1 with the orthogonal pursuit of Section 2.6.1 are feasible for (14) and typically provide a larger objective value than rounding coupled with post-processing orthogonalization. A better rounding procedure for this type of relaxation may prove more effective than the projection-pursuit approach; this is a direction for further research.

3. The low-leverage decomposition

Our second method is derived from the interpretation of principal component analysis as a matrix approximation problem. When the observations are drawn from a highly correlated family, the singular values of the data matrix $\boldsymbol{X}$ tend to decay rapidly. When this is the case, the matrix $\boldsymbol{X}$ is well approximated by a low-rank matrix $\boldsymbol{P}$.

It is rare that a large data set is compiled without error, but the errors may only affect a subset of the observations. We can model these errors through a multi-population model: we assume the bulk of the observations is well explained

```
Algorithm 2: Low-Leverage Decomposition
INPUT: An $n \times p$ data matrix $\boldsymbol{X}$; desired number $T$ of principal components.
OUTPUT: A $p \times T$ matrix $\boldsymbol{V}_\star$ with orthogonal columns.
  1. Find $(\boldsymbol{P}_\star, \boldsymbol{C}_\star)$ that solve
         $\text{minimize}_{(\boldsymbol{P},\boldsymbol{C})} \quad \|\boldsymbol{P}\|_{2\to 2}^* + \gamma \|\boldsymbol{C}\|_{2\to\infty}^*$
         $\text{subject to} \quad \boldsymbol{P} + \boldsymbol{C} = \boldsymbol{X}$
  2. Compute the SVD $\boldsymbol{P}_\star = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^*$.
  3. Set $\boldsymbol{V}_\star$ to the first $T$ columns of $\boldsymbol{V}$, that is, set
         $[\boldsymbol{V}_\star]_{ij} = [\boldsymbol{V}]_{ij}$ for $i = 1, \dots, p,$ and $j = 1, \dots, T$.
```

by a low-rank model while the remainder come from another population or are corrupted by measurement noise. Under this model, it is prudent to separate the corrupted data from the uncorrupted data before attempting to recover a low-rank model. When the corrupted rows are unknown, this task may seem daunting.

To accomplish this goal, we propose a semidefinite program that decomposes the input $\boldsymbol{X}$ into two matrices:

$$\begin{array}{ll} \text{minimize}_{(\boldsymbol{P},\boldsymbol{C})} & \|\boldsymbol{P}\|_{2\to 2}^* + \gamma \|\boldsymbol{C}\|_{2\to\infty}^* \\ \text{subject to} & \boldsymbol{P} + \boldsymbol{C} = \boldsymbol{X}. \end{array} \tag{16}$$

The norm $\|\boldsymbol{P}\|_{2\to 2}^*$ is the sum of the singular values of $\boldsymbol{P}$ and is known to promote low-rank solutions [20], while $\|\boldsymbol{C}\|_{2\to\infty}^*$ is the sum of the ℓ_2 norms of the rows of $\boldsymbol{C}$ and promotes group sparsity of the rows [47]. The quantity γ is a positive number that reflects the relative importances of these two priorities.

We call the optimal matrix pair $(\boldsymbol{P}_\star, \boldsymbol{C}_\star)$ for the problem (16) the *low-leverage decomposition* (LLD) of $\boldsymbol{X}$; we can interpret $\boldsymbol{C}_\star$ as an identified corruption and $\boldsymbol{P}_\star$ as a surrogate for the uncorrupted observations. We define our robust components as the right singular vectors of the surrogate matrix $\boldsymbol{P}_\star$. The detailed procedure appears in Algorithm 2. We show in Section 3.1 that our recovered data matrix $\boldsymbol{P}_\star$ has the additional property of being a low-leverage set of observations.

The LLD formulation is related to recent proposals [8, 9], and we discuss this point more in Section 4.2.

As we were preparing this manuscript, we became aware of independent work [56, 57] that also proposes (16) for the robust PCA problem. This work shows that, under certain hypotheses, the recovered low-rank data $\boldsymbol{P}_\star$ has the same row-space as the true data and the corrupted rows are correctly identified. While this paper was under review, we were further made aware of another similar proposal (under the name *low-rank representation*) used for a robust subspace segmentation problem [37].

3.1. Leverage scores and properties of the decomposition

In this section, we demonstrate that (16) extracts a low-leverage model for the data. This result follows from duality arguments that characterize the optimum of the convex program.

3.1.1. Leverage scores

The *leverage score* of the observation $\boldsymbol{x}_i$ corresponding to the ith row of $\boldsymbol{X}$ is given by the number $[\boldsymbol{H}]_{ii}$, where $\boldsymbol{H} = \boldsymbol{X}(\boldsymbol{X}^*\boldsymbol{X})^\dagger\boldsymbol{X}^*$ is the orthoprojector onto the column space of $\boldsymbol{X}$. In accord with common statistical parlance, we refer to $\boldsymbol{H}$ as the *hat matrix*.

A large leverage score tends to indicate that the corresponding observation lies outside of the bulk of the data, although it does not necessarily indicate that the point is influential in linear regression. This interpretation follows from the use of the hat matrix in least-squares regression. The hat matrix gains its name from the fact that it "puts the hat" on an observation—that is, it projects an observation onto a least-squares regression surface. A diagonal element $[\boldsymbol{H}]_{ii}$ is a measure of the influence that the ith observation has on the regression surface. We refer to [42, Ch. 6] for further discussion of leverage scores.

The following theorem shows that the leverage scores of our decomposition are bounded above by γ^2, justifying the name *low-leverage decomposition* for Algorithm 2.

Theorem 3.1. *Suppose* $(\boldsymbol{P}_\star, \boldsymbol{C}_\star)$ *is an optimal point of the program* (16). *Then the diagonal elements of the hat matrix* $\boldsymbol{H}_\star = \boldsymbol{P}_\star(\boldsymbol{P}_\star^*\boldsymbol{P}_\star)^\dagger\boldsymbol{P}_\star^*$ *are bounded above by* γ^2.

We establish Theorem 3.1 in Section 3.1.2 below. Theorem 3.1 demonstrates that our proposal provides a method for decomposing a data matrix $\boldsymbol{X}$ into a component with a (user-specified!) upper bound on the leverage plus an error term. Moreover, this result gives a statistical interpretation to the regularization parameter γ in (16).

We note that, while the program (16) guarantees a low-leverage decomposition, an assumption of suitably small leverage is a technical hypothesis in other works, e.g., [8, eq. (1.2)].

The reader should be warned that this method does not necessarily produce a low-leverage solution if we use our program to identify outlying data and then "prune" the rows. That is, suppose $(\boldsymbol{P}_\star, \boldsymbol{C}_\star)$ is an optimal point of (16) and $\boldsymbol{c}_i = \boldsymbol{0}$ for row indices $i \in I$. Then the corresponding matrix $\boldsymbol{X}_I = \boldsymbol{P}_I$ *does not* necessarily have leverage scores bounded above by γ^2. Rather, we interpret $\boldsymbol{P}_\star$ as a set of statistically "nice" observations that we use to build our principal components.

3.1.2. Proof of Theorem 3.1

We require some background results to prove the theorem.

Lemma 3.2 (First-order optimality conditions for (16))**.** *A feasible pair* $(\boldsymbol{P}, \boldsymbol{C})$ *is optimal for* (16) *if and only if there exists a matrix* $\boldsymbol{Q}$ *such that*

$$\langle \boldsymbol{Q}, \boldsymbol{P}\rangle = \|\boldsymbol{P}\|_{2\to 2}^{*}, \quad \|\boldsymbol{Q}\|_{2\to 2} \le 1 \tag{17a}$$

$$-\langle \boldsymbol{Q}, \boldsymbol{C}\rangle = \gamma\|\boldsymbol{C}\|_{2\to\infty}^{*}, \quad \|\boldsymbol{Q}\|_{2\to\infty} \le \gamma, \tag{17b}$$

Proof. It follows from standard subgradient conditions that a feasible point $(\boldsymbol{P}, \boldsymbol{C})$ minimizes the functional in (16) if and only if zero is in the subgradient $\partial(\|\boldsymbol{P}\|_{2\to 2}^{*} + \gamma\|\boldsymbol{X} - \boldsymbol{P}\|_{2\to\infty}^{*})$. By the additivity of subgradients [48, Thm. 23.8], this condition holds if and only if there exists a matrix $\boldsymbol{Q}$ such that the subgradient conditions $\boldsymbol{Q} \in \partial\|\boldsymbol{P}\|_{2\to 2}^{*}$ and $-\boldsymbol{Q} \in \partial(\gamma\|\boldsymbol{C}\|_{2\to\infty}^{*})$ are in force. We show that these subgradient conditions are equivalent to (17).

The rest of the proof demonstrates that $\boldsymbol{Q} \in \partial\|\boldsymbol{P}\|_{2\to 2}^{*}$ is equivalent to (17a). We omit the analogous proof of equivalence between $-\boldsymbol{Q} \in \partial(\gamma\|\boldsymbol{C}\|_{2\to\infty}^{*})$ and relation (17b).

By definition of the subdifferential, $\boldsymbol{Q} \in \partial\|\boldsymbol{P}\|_{2\to 2}^{*}$ if and only if for every perturbation $\boldsymbol{\Delta}$ the subgradient inequality

$$\langle \boldsymbol{Q}, \boldsymbol{\Delta}\rangle \le \|\boldsymbol{P} + \boldsymbol{\Delta}\|_{2\to 2}^{*} - \|\boldsymbol{P}\|_{2\to 2}^{*} \tag{18}$$

holds. Suppose first that (17a) holds. Then, for all $\boldsymbol{\Delta}$, we have

$$\langle \boldsymbol{Q}, \boldsymbol{\Delta}\rangle = \langle \boldsymbol{Q}, \boldsymbol{P} + \boldsymbol{\Delta}\rangle - \|\boldsymbol{P}\|_{2\to 2}^{*} \le \|\boldsymbol{Q}\|_{2\to 2}\|\boldsymbol{P} + \boldsymbol{\Delta}\|_{2\to 2}^{*} - \|\boldsymbol{P}\|_{2\to 2}^{*},$$

where the inequality follows by the definition of dual norms. Since $\|\boldsymbol{Q}\| \le 1$ by assumption, the subgradient inequality (18) must hold.

To establish the converse, we must show that the subgradient inequality (18) implies (17a). To this end, assume (18) holds for all perturbations $\boldsymbol{\Delta}$. Taking the specific choice $\boldsymbol{\Delta} = \boldsymbol{P}$ gives $\langle \boldsymbol{Q}, \boldsymbol{P}\rangle \le \|\boldsymbol{P}\|_{2\to 2}^{*}$, while $\boldsymbol{\Delta} = -\boldsymbol{P}$ gives the reverse inequality $\langle \boldsymbol{Q}, \boldsymbol{P}\rangle \ge \|\boldsymbol{P}\|_{2\to 2}^{*}$. Therefore the subgradient inequality (18) implies $\langle \boldsymbol{Q}, \boldsymbol{P}\rangle = \|\boldsymbol{P}\|_{2\to 2}^{*}$, which is the first relation of (17a).

To show the second relation, we take a nonzero perturbation $\boldsymbol{\Delta}$ that satisfies $\langle \boldsymbol{Q}, \boldsymbol{\Delta}\rangle = \|\boldsymbol{Q}\|_{2\to 2}\|\boldsymbol{\Delta}\|_{2\to 2}^{*}$; such a matrix $\boldsymbol{\Delta}$ must always exist in finite dimensions since suprema are attained in the trace definition of norms. Then the subgradient inequality (18) implies

$$\|\boldsymbol{Q}\|_{2\to 2}\|\boldsymbol{\Delta}\|_{2\to 2}^{*} \le \|\boldsymbol{P} + \boldsymbol{\Delta}\|_{2\to 2}^{*} - \|\boldsymbol{P}\|_{2\to 2}^{*} \le \|\boldsymbol{\Delta}\|_{2\to 2}^{*}$$

where the second relation follows by the triangle inequality. Since $\boldsymbol{\Delta} \ne \boldsymbol{0}$, we have shown that the subgradient inequality implies $\|\boldsymbol{Q}\|_{2\to 2} \le 1$. Hence $\boldsymbol{Q} \in \partial\|\boldsymbol{P}\|_{2\to 2}^{*}$ is equivalent to (17a), as claimed. □

Before continuing, we introduce another fact concerning the subgradient $\partial\|\boldsymbol{P}\|_{2\to 2}^{*}$. Let $\boldsymbol{P} = \boldsymbol{U\Sigma V}^{*}$ be the compact SVD of $\boldsymbol{P}$. It follows from [54] that $\boldsymbol{Q} \in \partial\|\boldsymbol{P}\|_{2\to 2}^{*}$ implies $\boldsymbol{Q} = \boldsymbol{UV}^{*} + \boldsymbol{W}$, where, in particular, $\boldsymbol{UV}^{*}\boldsymbol{W} = \boldsymbol{0}$. With this background, we now establish 3.1.

Proof of Theorem 3.1. From the characterization of the subgradient of unitarily invariant norms discussed above [54], we know that $\boldsymbol{Q} = \boldsymbol{U}\boldsymbol{V}^* + \boldsymbol{W}$ with $\boldsymbol{U}\boldsymbol{V}^*\boldsymbol{W}^* = \boldsymbol{0}$. Thus,

$$\boldsymbol{Q}\boldsymbol{Q}^* = \boldsymbol{U}\boldsymbol{U}^* + \boldsymbol{W}\boldsymbol{W}^* \succcurlyeq \boldsymbol{U}\boldsymbol{U}^* = \boldsymbol{H}_\star,$$

where the last equality can be easily checked using the definition of $\boldsymbol{H}_\star$ and the SVD of $\boldsymbol{P}_\star$. Since the diagonal entries of a positive-semidefinite matrix are nonnegative, this relation implies $[\boldsymbol{H}_\star]_{ii} \leq [\boldsymbol{Q}\boldsymbol{Q}^*]_{ii}$. Recall that the $\ell_2 \to \ell_\infty$ operator norm is the maximum ℓ_2 row norm of the matrix. Thus relation (17b) of Lemma 3.2 implies that $[\boldsymbol{Q}\boldsymbol{Q}^*]_{ii} \leq \gamma^2$, which completes the proof. □

3.1.3. The choice of γ

In this section, we study how the value of the regularization parameter γ affects the properties of the decomposition. We begin by showing that, when $\gamma \geq 1$, the degenerate solution $(\boldsymbol{P}_\star, \boldsymbol{C}_\star) = (\boldsymbol{X}, \boldsymbol{0})$ minimizes (16). This claim follows by explicit construction of a dual certificate.

Let $\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^*$ be the compact SVD of $\boldsymbol{X}$, and define $\boldsymbol{Q} = \boldsymbol{U}\boldsymbol{V}^*$. Clearly $\langle \boldsymbol{Q}, \boldsymbol{X} \rangle = \|\boldsymbol{X}\|_{2\to 2}^*$, so $\boldsymbol{Q}$ satisfies (17a) with $\boldsymbol{P}_\star = \boldsymbol{X}$. By construction, the maximum singular value of $\boldsymbol{Q}$ is bounded above by one. Equivalently, $\boldsymbol{Q}\boldsymbol{Q}^* \preccurlyeq \mathbf{I}$. This inequality implies $[\boldsymbol{Q}\boldsymbol{Q}^*]_{ii} \leq 1$. Since the diagonal entries of $\boldsymbol{Q}\boldsymbol{Q}^*$ are the squared Euclidean row norms of $\boldsymbol{Q}$, we have shown that $\|\boldsymbol{Q}\|_{2\to\infty} \leq 1 \leq \gamma$. This bound demonstrates that $\boldsymbol{Q}$ satisfies (17b) with $\boldsymbol{C}_\star = \boldsymbol{0}$, which certifies optimality of this degenerate solution by Lemma 3.2.

We now show that the regularization parameter γ gives an upper bound on the rank of the optimal $\boldsymbol{P}_\star$. It is easy to show using the SVD of $\boldsymbol{P}_\star$ that the trace of the hat matrix $\boldsymbol{H}$ defined above is equal the rank of $\boldsymbol{P}_\star$. Since $[\boldsymbol{H}]_{ii} \leq \gamma^2$ by Theorem 3.1, we must have

$$\operatorname{rank}(\boldsymbol{P}_\star) = \operatorname{trace}(\boldsymbol{H}) \leq n\gamma^2. \tag{19}$$

The rank is a positive integer, so $\gamma < 1/\sqrt{n}$ implies that the optimal $\boldsymbol{P}_\star$ is trivial. Moreover, in order to get T meaningful components in Step 2 of Algorithm 2, we require $\operatorname{rank}(\boldsymbol{P}_\star) \geq T$. Thus, we can limit ourselves to situations where $\gamma \in [\sqrt{T/n}, 1]$.

Inequality (19) has implications for the numerical solution of (16). As we discuss in Section 3.2, the bulk of the computation comes from computing an SVD at each iteration. When the solution of the optimization problem has low rank, the iterates also tend to have low rank. This allows us to save significant computational effort by computing partial singular value decompositions at each step. A judicious choice of γ can increase the performance of our algorithm immensely. We find that taking $n\gamma^2 \approx T^2$ is a useful heuristic for achieving a rank-T optimal solution, so long as $n \gg T^2$.

On the other hand, typical statistical data does not show true low-rank behavior even when there are no outliers. Therefore, forcing the optimal decomposition to be low rank typically results in a dense corruption $\boldsymbol{C}_\star$. This effect may be mitigated somewhat by another formulation we discuss briefly in Section 3.3. In practice we find that setting γ somewhat less than $\sqrt{p/n}$, say $\gamma = 0.8\sqrt{p/n}$, provides good principal components, but it does poorly in the context of outlier identification. We discuss specific parameter choices for our experiments in Section 5.

3.2. Computing the low-leverage decomposition

Although general-purpose semidefinite programming software such as CVX [26, 27] can solve small instances of (16) efficiently, the interior-point methods they utilize may be unable to complete even a single iteration of a large-scale problem. This observation indicates that we need to use different methods for large-scale problems.

To solve (16), we recommend an alternating direction augmented Lagrangian algorithm analogous to the one used in [8]; see also [36]. The generic form of the method is known as the Augmented Lagrangian Method of Multipliers (ALMM) that, according to [19], first appears in [23, 24]. The augmented Lagrangian for (16) with dual variable $\boldsymbol{Q}$ is given by

$$\mathcal{L}_\mu(\boldsymbol{P}, \boldsymbol{C}, \boldsymbol{Q}) = \|\boldsymbol{P}\|^*_{2\to 2} + \gamma \|\boldsymbol{C}\|^*_{2\to\infty} + \langle \boldsymbol{X} - \boldsymbol{P} - \boldsymbol{C}, \boldsymbol{Q}\rangle + \frac{\mu}{2}\|\boldsymbol{X} - \boldsymbol{P} - \boldsymbol{C}\|^2_{\mathrm{F}}.$$

Given the initial starting point $\boldsymbol{P}^0 = \mathbf{0}$, we alternately solve

$$\begin{aligned} \boldsymbol{C}^{k+1} &= \arg\min_{\boldsymbol{C}} \mathcal{L}_\mu(\boldsymbol{P}^k, \boldsymbol{C}, \boldsymbol{Q}^k), \text{ and} \\ \boldsymbol{P}^{k+1} &= \arg\min_{\boldsymbol{P}} \mathcal{L}_\mu(\boldsymbol{P}, \boldsymbol{C}^{k+1}, \boldsymbol{Q}^k) \end{aligned}$$

and then update the multiplier by the feasibility gap $\boldsymbol{Q}^{k+1} = \boldsymbol{Q}^k + \mu(\boldsymbol{X} - \boldsymbol{P}^{k+1} - \boldsymbol{C}^{k+1})$. The minimizations above have an explicit form in terms of shrinkage operations [10].

$$\boldsymbol{C}^{k+1} = \mathrm{RowShrink}\left(\boldsymbol{X} - \boldsymbol{P}^k + \frac{1}{\mu}\boldsymbol{Q}^k, \frac{\gamma}{\mu}\right) \tag{20a}$$

$$\boldsymbol{P}^{k+1} = \mathrm{SpecShrink}\left(\boldsymbol{X} - \boldsymbol{C}^{k+1} + \frac{1}{\mu}\boldsymbol{Q}^k, \frac{1}{\mu}\right). \tag{20b}$$

Here, the operator $\mathrm{RowShrink}(\boldsymbol{A}, \nu)$ soft-thresholds each row $\boldsymbol{a}_i$ of $\boldsymbol{A}$.

$$\mathrm{RowShrink}(\cdot, \nu) : \boldsymbol{A} \longmapsto \mathrm{diag}([1 - \nu/\|\boldsymbol{a}_i\|_2]_+) \cdot \boldsymbol{A},$$

where $[x]_+ = \max\{x, 0\}$. Similarly $\mathrm{SpecShrink}(\boldsymbol{A}, \nu)$ soft-thresholds the singular values of $\boldsymbol{A}$.

$$\mathrm{SpecShrink}(\cdot, \nu) : \boldsymbol{U\Sigma V}^* \longmapsto \boldsymbol{U}\left[\boldsymbol{\Sigma} - \nu\mathbf{I}\right]_+ \boldsymbol{V}^*,$$

where the operator $[\cdot]_+$ is applied elementwise. Inspired by the parameter choices of [36], we initialize the algorithm with $\boldsymbol{P}^0 = \boldsymbol{0}$ and set the parameter $\mu = \sqrt{np}/\|\boldsymbol{X}\|_{2\to\infty}^*$. We stop the algorithm when the iterates are nearly feasible, that is, $\left\|\boldsymbol{X} - \boldsymbol{P}^k - \boldsymbol{C}^k\right\| < 10^{-7}\|\boldsymbol{X}\|_{\mathrm{F}}$.

The major cost when running this algorithm involves computing the spectral shrinkage operator in (20b). When the iterates $\boldsymbol{P}^k$ have low rank, we can save significant computational effort by performing only partial singular value decompositions as suggested by [36]. We can leverage our analysis in Section 3.1.3 to ensure that the optimal $\boldsymbol{P}_\star$ is low rank. Since the algorithmic iterates tend to have low rank when the optimal point has low rank, we may improve the performance of our algorithm by choosing γ to limit the rank of the optimal solution. In practice, we have found that one should set the quantity $n\gamma^2$ somewhat larger than the desired rank of the solution, e.g., $n\gamma^2 \approx T^2$ when we desire a rank-T iterates.

3.3. Extensions for a noisy model

We note that there is an obvious extension of the LLD when one wants to account for an additional of noise in the model. Suppose that in addition to gross corruptions of certain observations, we would also like to account for small corruptions or noise that may be spread throughout the data.

Instead of enforcing the equality $\boldsymbol{X} = \boldsymbol{P} + \boldsymbol{C}$, we allow for some additional slack of the form $\|\boldsymbol{X} - \boldsymbol{P} - \boldsymbol{C}\|_{\mathrm{F}} \le \eta$, where η is an estimate for the noise level. That is, we solve the problem

$$\begin{aligned} \text{minimize} \quad & \|\boldsymbol{P}\|_{2\to2}^* + \gamma\|\boldsymbol{C}\|_{2\to\infty}^* \\ \text{subject to} \quad & \|\boldsymbol{X} - \boldsymbol{P} - \boldsymbol{C}\|_{\mathrm{F}} \le \eta \end{aligned} \tag{21}$$

When $\eta = 0$, this is equivalent to our proposal (16) for the gross corruption model. Loss functions other than the Frobenius norm are also possible, but the Frobenius norm is invariant to a change of the observation basis, a feature of (16) that we would like to preserve.

This formulation is also studied in the independent work [56, 57]. It is shown there that under some restrictive technical conditions, the decomposition in (21) above results in an optimal $\boldsymbol{P}_\star$ that is close to a matrix with the same row space as the true observations, and an optimal $\boldsymbol{C}_\star$ that is close to a matrix that correctly identifies the column support of the corruption.

4. Previous work

This section describes previous work on robust formulations for principal component analysis. Convex approaches to robust PCA are unusual, and, as a consequence, many other attempts at robust PCA lack rigorous algorithms. Many proposals couple a mathematical formulation with a heuristic algorithm; others introduce an algorithm without any mathematical foundation.

In Sections 4.1 and 4.2, we describe the methods in the literature that are most closely related to our proposals. We then detail an approach for robust PCA recommended by Maronna [39]. We conclude this section with a short overview of other robust PCA proposals that have appeared in the literature.

4.1. Antecedents for MDR: Projection pursuit PCA

Our MDR proposal is a particular instance of an approach that has come to be known as *projection-pursuit PCA* (PP-PCA). The theoretical properties of PP-PCA are well understood; see for instance [14] and [13].

All of the algorithms we have found in the literature for computing PP-PCA are meant to operate with an arbitrary scale. In view of the fact that the PP-PCA problem is NP-hard, it is unsurprising that the literature contains no PP-PCA algorithms with proofs of correctness and tractability. Indeed, we have been unable to find other work that recognizes that the PP-PCA problem is intractable in a rigorous sense.

The original study of Li and Chen [35] uses a Monte Carlo approach that was found to be computationally expensive. In theory, even simple Monte Carlo methods (e.g., randomly sampling the unit sphere) can produce arbitrarily good solutions to problem (5) with an arbitrary (continuous) scale. Given the computational hardness of the problem, however, it is unlikely that naïve Monte Carlo approaches can provide guarantees of computational efficiency. In contrast, one may view MDR as a sophisticated Monte Carlo approach for a specific instance of PP-PCA.

Some recent algorithms for PP-PCA rely on heuristics. A popular and fast algorithm for generic projection-pursuit PCA is the finite direction method of Croux and Ruiz-Gazen [13]. This technique replaces the search over the entire unit sphere $\{\boldsymbol{v} \mid \|\boldsymbol{v}\|_2 = 1\}$ with a finite search over the directions that appear among the observations: $\boldsymbol{v} \in \{\boldsymbol{x}_1/\|\boldsymbol{x}_i\|_2, \dots, \boldsymbol{x}_n/\|\boldsymbol{x}_n\|_2\}$. The hope is that directions of large scale are likely to be well approximated by directions appearing in the data. This heuristic may perform poorly when n and p are large because it takes an extremely large number of points to cover a high-dimensional sphere.[4]

Another work highlights recent interest in solving (2). Kwak [34] rediscovers PP-PCA with the MD scale and provides a simple algorithm that is shown to produce a local maximum for the MD scale. No global performance guarantees are provided.

4.2. A convex approach

Recently, a method of Chandrasekaran et al. [9] has been adapted for robust PCA in [8]. This approach attempts to decompose the data matrix into a sum

[4] It is not difficult to construct a situation where this finite direction method performs quite poorly—for instance, if the observations are orthonormal, the finite direction method applied to the MD scale will find a direction with scale no larger than $p^{-1/2}\|\boldsymbol{X}\|_{2\to1}$.

of a low-rank matrix and a sparse matrix via the semidefinite program

$$\begin{array}{ll} \text{minimize} & \|\boldsymbol{L}\|_{2\to 2}^{*} + \lambda \|\boldsymbol{S}\|_{1\to\infty}^{*} \\ \text{subject to} & \boldsymbol{L} + \boldsymbol{S} = \boldsymbol{X}. \end{array} \tag{22}$$

The nuclear norm $\|\cdot\|_{2\to 2}^{*}$ promotes low rank and the matrix ℓ_1 norm $\|\cdot\|_{1\to\infty}^{*}$ promotes sparsity. We refer to this method as N+L1. The works [8, 9] provide conditions under which N+L1 succeeds in *exactly* recovering a low-rank and sparse component.

This convex approach is principled in the sense that the mathematical formulation is also algorithmically tractable. On the other hand, it lacks an invariance to a change in the observation basis possessed by all other methods we discuss, including standard PCA. That is, applying a rotation $\boldsymbol{U}^{*}\boldsymbol{U} = \mathbf{I}$ to the data $\widehat{\boldsymbol{X}} = \boldsymbol{X}\boldsymbol{U}$ does not result in a similar rotation of the decomposition due to the fact that the norm $\|\cdot\|_{1\to\infty}^{*}$ is not invariant under this transformation.

One may argue that this invariance is inconsequential: in real data, the particular choice of coordinates has a meaning and outliers may occur coordinatewise. This is the case in some specific examples, such as image data that contain specularities [8]. Nevertheless, PCA is intended to locate a coordinate basis that explains data more effectively than the standard basis [30], and basis invariance is a feature of all of the other methods for robust PCA that are discussed in this work. The question of whether N+L1 is appropriate for a given set of data will likely depend on the types of corruptions present in the data.

4.3. Spherical PCA

Another approach, known as spherical principal components (sphPCA) [38], rescales the observations to unit (Euclidean) norm and applies standard PCA to this modified data. To implement the sphPCA method, we first compute a normalized matrix $\widehat{\boldsymbol{X}}$. Each row of $\widehat{\boldsymbol{X}}$ is the normalized version of the corresponding row of the centered data matrix $\boldsymbol{X}$, that is $\widehat{\boldsymbol{x}}_i = \boldsymbol{x}_i / \|\boldsymbol{x}_i\|_2$. Using the row-normalization operator from (12), we can express the normalized matrix as $\widehat{\boldsymbol{X}} = \mathcal{N}(\boldsymbol{X})$.

The robust components are then defined as the standard principal components of the rescaled matrix $\widehat{\boldsymbol{X}}$. Since all of the observations from the normalized matrix $\widehat{\boldsymbol{X}}$ have unit Euclidean norm, there are no large magnitude observations that exert an undue influence on the principal components.

A study by Maronna [39] shows that sphPCA enjoys good practical performance. The ease of implementation and relatively good behavior of sphPCA leads Maronna et al. [40] to suggest it as the default choice for robust principal component analysis. As a result, we use sphPCA as a baseline comparison for the performance of our robust methods in Section 5.

4.4. Other proposals

Some of the earliest methods for robust PCA compute approximations of correlation or covariance matrices using robust methods. Gnanadesikan and Kettenring propose direct robust estimation of the covariance matrices through robust estimation of the individual entries [25]. This may lead to counterintuitive results such as nonpositive covariance matrices. An alternative approach explicitly enforces positive matrices as minimizers of a functional such as an M-estimator [17]; see also the more recent study [12].

A representative example of robust PCA from the machine learning community is the work of De La Torre and Black [16]. They define the robust components as the minimum of a highly nonconvex energy function and attempt to minimize this energy function using an iteratively reweighted least-squares algorithm coupled with an annealing step. No theoretical guarantees of correctness for the algorithm are provided.

Another line of work[5] in robust PCA involves iteratively identifying and removing outliers in the data [4, 15, 18]. A recent approach along these lines appears in the paper [55] of Xu et al.; their algorithm randomly removes observations that appear to have high influence in the current estimate of the principal components. The principal component estimate is then computed from the trimmed data. There, the authors establish strong theoretical properties of their algorithm, including a high breakdown point in the high-dimensional scaling regime where $n \to \infty$ and $n/p \to c > 0$.

5. Experiments

This section provides some experiments comparing our proposals against classical PCA, sphPCA, and N+L1. Our choice of sphPCA was motivated by its simple formulation and the study [39] which shows that sphPCA provides good performance in comparison to other robust PCA methods, including several types of PP-PCA algorithms. We also compare our methods to N+L1 as the formulation of this method is closely related to the formulation of LLD.

We begin in Section 5.1 with an experiment involving synthetic data where we compare the behavior of the methods when observations are drawn from a mixed model with known covariance structures. In Section 5.2, we look at the projection of two data sets on the top robust component. Section 5.3 repeats a multiple-component experiment of Maronna [40, Fig. 6.1] with additional robust methods.

All of these experiments are conducted using MATLAB. Following the principle of reproducible research [5], we provide code that reproduces the exact experiments in this work [41].

[5] We thank the anonymous reviewers for these references. It was recognized in [15] that identifying a low dimensional subspace to which a large fraction of the observations belong is an NP-hard problem. This fact is an LLD analogue to our hardness result for MDR in Section 2.3, as it shows that determining the robust subspace that LLD seeks is NP-hard even when a small proportion of the observations are corrupted.

Parameter Settings. We have endeavored to use the same parameters in each experiment. A reviewer correctly noted that we could set the LLD parameter γ using cross-validation, but most of our experiments do not have a clear objective for which we seek to optimize the performance of the robust PCA methods. Rather, our experiments are meant to show the qualitative differences in performance of these methods over a variety of scenarios. Moreover, the fact that LLD performs qualitatively well for our fixed choice of $\gamma = 0.8\sqrt{p/n}$ in all of our experiments indicates that this is a robust parameter choice and a good starting point for practical implementations.

For MDR, we use $K = 94$ rounding trials as discussed in Section 2.4. We take the LLD weight parameter $\gamma = 0.8\sqrt{p/n}$, and, unless otherwise noted, set the N+L1 parameter $\lambda = 1/\sqrt{n}$.

Centering. For each experiment involving real data, we center the observations about their Euclidean median. The Euclidean median $\widehat{\boldsymbol{\mu}}$ is a robust estimate of the center of the data, and is defined as

$$\widehat{\boldsymbol{\mu}} = \arg\min_{\boldsymbol{\mu}} \sum_{i=1}^{n} \|\boldsymbol{x}_i - \boldsymbol{\mu}\|_2. \tag{23}$$

Maronna [40, Ch. 9] gives a method to solve this convex problem for $\widehat{\boldsymbol{\mu}}$.

5.1. Synthetic data: Identifying the true subspace(s)

Here, we draw observations from a mixture of two centered Gaussian distributions and apply PCA, LLD, MDR, and sphPCA to the resulting data. We analyze how closely the derived components align with the subspace corresponding to the top eigenvalues of the covariance matrices that generate the data. Unlike our other experiments, we do not use N+L1 in this section, as the observation model underlying N+L1 (i.e., sparse corruptions) is manifestly violated by our synthesized data.

5.1.1. Identifying a subspace with isotropic corruptions

Our first experiment draws $n = 100$ observations, each of dimension $p = 20$, from a two-population Gaussian model: with probability $(1 - \varepsilon)$, the observation is drawn from an anisotropic Gaussian distribution $\textsc{Normal}(\mathbf{0}, \mathbf{\Sigma}_{\text{fast}})$, and with probability ε, the observation is drawn from the isotropic Gaussian distribution $\textsc{Normal}(\mathbf{0}, \boldsymbol{I})$.

The covariance matrix $\mathbf{\Sigma}_{\text{fast}}$ has eigenvalues $\sigma_i(\mathbf{\Sigma}_{\text{fast}}) = 2^{2-2i}$ for $i = 1, 2, \dots, p$. The eigenvectors of $\mathbf{\Sigma}_{\text{fast}}$ are random.[6] Due to the exponential decay of the eigenvalues, we call observations from this population the fast-decay subjects.

[6] By "random eigenvectors", we mean that the matrix corresponding to the eigenbasis of $\mathbf{\Sigma}_{\text{fast}}$ is constructed by orthogonalizing the columns of a $p \times p$ matrix whose entries are iid Gaussian.

To summarize, a $(1-\varepsilon)$-fraction of subjects come from a low-rank model, while the remaining ε-fraction are simply white noise. Both of these distributions are centered by construction, so we apply no further centering to the data.

We apply four principal component methods to this data: PCA, sphPCA, MDR, and LLD. For each method, we determine the top three components, and note the angle between the subspace spanned by these three components and the subspace spanned by the top two eigenvectors of the covariance matrix $\mathbf{\Sigma}_{\text{fast}}$. (See [3, Chapter VII.1] for a discussion of the angle between subspaces, and, in particular, the discussion of the distance between subspaces on page 202.) Note that we oversample the subspace slightly by taking three robust components. This gives each method a bit of extra help, as they only need to capture the two-dimensional subspace inside a three-dimensional subspace.

We repeat this experiment 1000 times for each of 51 values of ε between zero and one. Figure 1 shows the angle to the subspaces as a function of ε. The dark band is the middle 50% of the angles to the subspace, and the line in the center is the median angle. The edges of the light band mark the 5th and 95 percentile of the angles.

The results are striking. As expected, every method performs well when $\varepsilon = 0$, and each method performs equally poorly when $\varepsilon = 1$. Between these two extremes, however, we see clear differences in the behavior of these four methods. As the proportion of isotropic observations increases, the subspace identified by PCA rapidly diverges from the dominant subspace of $\mathbf{\Sigma}_{\text{fast}}$.

This transition is more gradual for MDR, but it begins immediately as ε differs from zero. In contrast, the transition between identifying and not identifying the dominant subspace of $\mathbf{\Sigma}_{\text{fast}}$ occurs much later for sphPCA and LLD. Indeed, LLD consistently identifies the dominant subspace of $\mathbf{\Sigma}_{\text{fast}}$ even as the proportion of isotropically distributed becomes as large as 70%. This clearly shows that LLD is the dominant method for recovering the top eigenspace of $\mathbf{\Sigma}_{\text{fast}}$ in the presence of isotropic corruptions.

5.1.2. Identifying a subspace with anisotropic corruptions

This experiment modifies the experiment in Section 5.1.1 by replacing the isotropic population with an anisotropic Gaussian population whose covariance matrix has slowly decaying eigenvalues.

As before, $n = 100$ and $p = 20$. With probability $(1-\varepsilon)$, we draw our observations from the fast-decay population described in Section 5.1.1. With probability ε, we draw our observations $\textsc{Normal}(\mathbf{0}, \mathbf{\Sigma}_{\text{slow}})$, where the covariance matrix $\mathbf{\Sigma}_{\text{slow}}$ has eigenvalues i^{-2} for $i = 1, 2, \ldots, p$, and the corresponding eigenspaces are random.

We find the top three components of the data using PCA, LLD, sphPCA, and MDR, and then compute the angle between the subspace spanned by these components and the top two-dimensional eigenspace of $\mathbf{\Sigma}_{\text{fast}}$ and $\mathbf{\Sigma}_{\text{slow}}$, oversampling the subspace by one component as before. We repeat the experiment 1000 times for each of 51 values of ε between zero and one. The results are

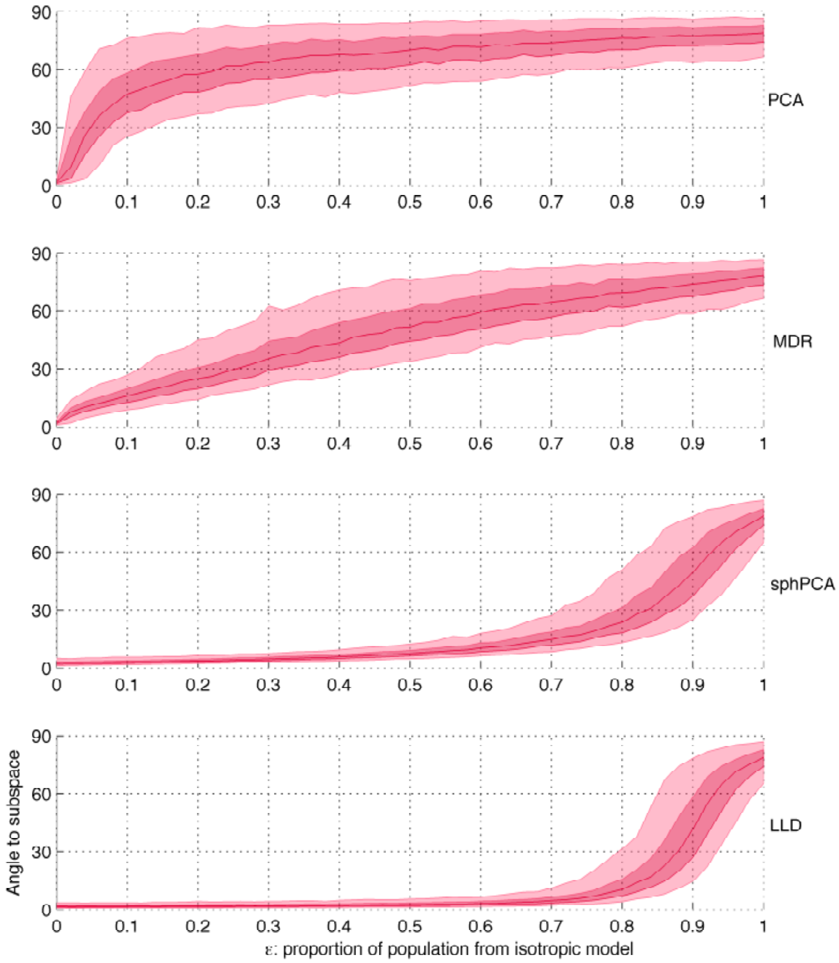

Fig 1. Subspace identification with isotropic corruptions. *The angle between the subspace spanned by the top two components of a rapidly decaying covariance matrix and the top three components calculated from the data, as a function of the proportion of isotropic corruption. The dark band is the middle* 50% *of the angles, and the edge of the light band denotes the* 5*th and* 95*th percentiles of the angles. The dark line in the center is the median.*

shown in Figure 2, where, as in Figure 1, the dark band is the interquartile range of the angles, the light band extends to the 5th and 95th percentiles, and the centerline is the median angle.

The difference lies in the way that the methods transition from identifying the fast-decay subspace to the slow-decay subspace. It appears that PCA and MDR do not strongly prefer the fast decaying subspace over the slow decaying

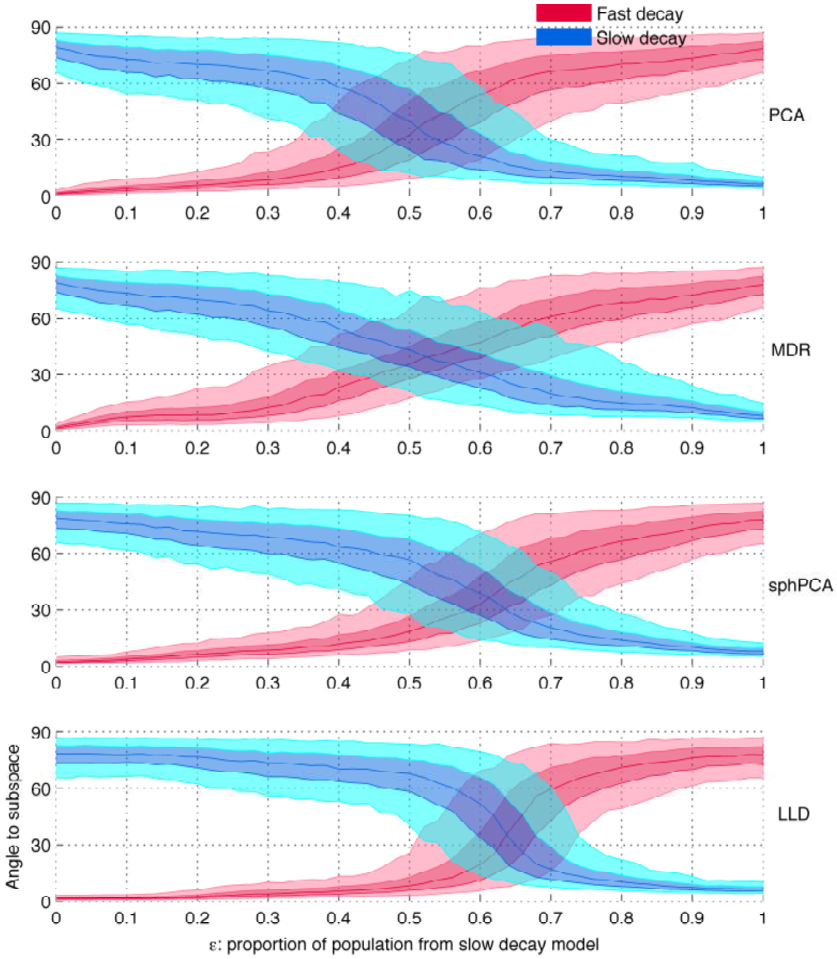

Fig 2. Subspace identification in a mixed population model. *The angle between the top three components and the top two-dimensional eigenspaces of the fast-decay (red) and slow-decay (blue) populations, displayed as a function of the proportion of data drawn from the slow-decay population. The dark bands show the middle* 50% *of the angles, and the light bands demarcate the* 5*th and* 95*th percentiles of the angles. The dark line in the center of the band is the median angle for each* ε.

subspace; rather, the transition region is relatively centered about $\varepsilon = 0.5$. In comparison, sphPCA and LLD appear to prefer to find the fast-decay subspace over the slow decay subspace. The transition region is much wider for MDR than for any of the other methods, which hints at a different behavior in the transition region.

In order to investigate this transition between the identified subspaces, Figure 3 shows a scatter plot of the angle to the slow-decay subspace versus the

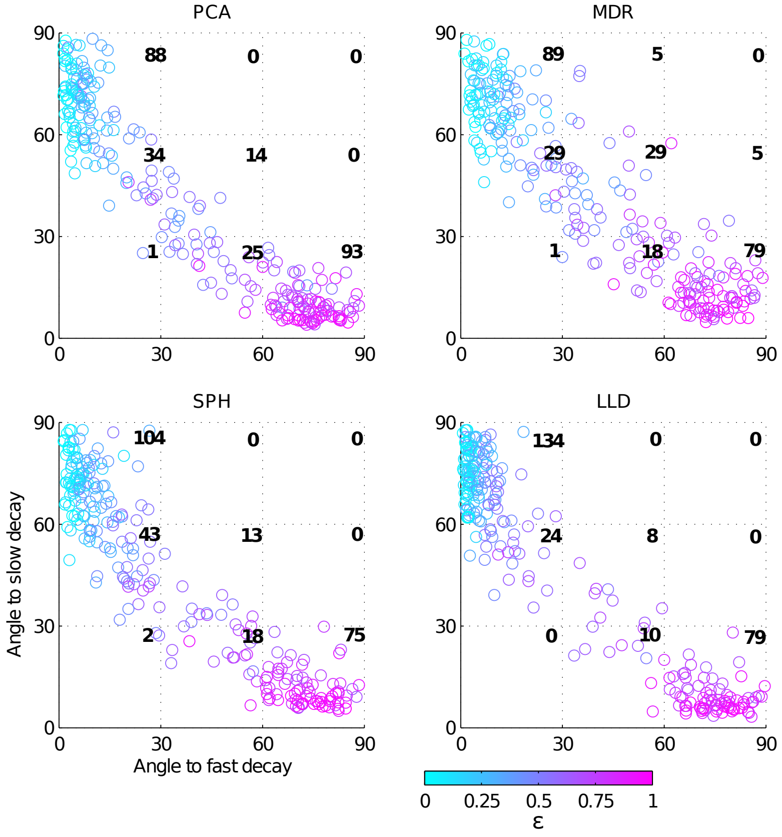

Fig 3. Angle to the fast-decay subspace vs. the slow-decay subspace. *This scatter plot shows the angle of the subspace spanned by the principal components to the subspace spanned by the slow decay subspace versus the fast decay subspace. The color of the scattered points denotes the proportion* ε *of points from the slow decay subspace. Five points are used for each of* 51 *values of* ε *between zero and one, with a total of* 255 *points on each plot. The numbers denote the total number of points in each grid box.*

angle to the fast decay subspace. The color of the scatter points indicates the value of ε. We plot only 5 points for each value of ε for visual clarity.

We are most interested in the extent to which the scatter plot shows bimodality: do the components split the difference between the subspaces, or do they choose one subspace over the other? It is clear that LLD is the most strongly bimodal, with most points clustered tightly in the upper-left and lower right corners. MDR shows the most spread, while sphPCA and PCA exhibit intermediate behavior.

Why might MDR behave this way? Suppose that our observations are orthonormal. The variance is then constant in every direction, but the deviation is larger by a factor of $\sqrt{p}$ in the *average* direction than along any of the directions expressed in the data. This thought experiment points to a fundamentally different qualitative behavior between directions of maximum deviance and maximum variance, and may explain in part the behavior of MDR in Figures 1 and 2.

Despite this thought experiment, if the goal of the analysis is the identification of a subspace that corresponds to the fast-decay group when $\varepsilon < 0.5$ (or the slow-decay group for $\varepsilon > 0.75$), then LLD is by far the most effective method. If we seek a method that finds a balance between competing subspaces, perhaps MDR is a better choice.

5.2. Projection onto the top component

In this section, we study the robust component methods applied to two data sets. The first set is a selection of environmental factors that may affect the concentration of nitrogen dioxide around Oslo, Norway. The second example is constructed from Fisher's classic iris data. In each case, we examine the spread of the data in the direction of the top robust component.

5.2.1. Experimental setup

We extract the dominant component from each data set using our methods (MDR and LLD), other robust methods (sphPCA and N+L1), and standard PCA. We project the data onto the top component for each method and compare the performance of the methods by the *interquartile range* (IQR), that is, the distance between the 25th and 75th percentile of the projected data.

With the iris data in Section 5.2.3, we find that $\lambda = 1/\sqrt{n}$ gives a trivial result: no outliers were identified by N+L1. Instead, we use the more favorable choice $\lambda = 0.3/\sqrt{n}$.

5.2.2. Norwegian nitrogen dioxide data

Our data for this experiment consists of 500 observations of eight environmental factors around Oslo, Norway, available on the Statlib archive [1]. The variables include the log-concentration of nitrogen dioxide (NO_2) particles, the number of cars per hour, the wind speed, and several additional factors useful for predicting the concentration of NO_2 particles. The projection onto the top component may be interpreted as an air quality index.

We calculate the top component of the data using each method. In Figure 4 we plot the projection of the data onto the direction of these components using a standard box-and-whisker plot. The whiskers extend either 1.5 times the IQR beyond the edge of the box or to the extreme data point. We consider points that lie beyond the whiskers outliers. We give the percentage of outliers and several order statistics of the data in Table 2.

Every robust method results in a larger IQR than PCA. The MDR component finds the largest IQR, and the LLD method finds the smallest IQR among the

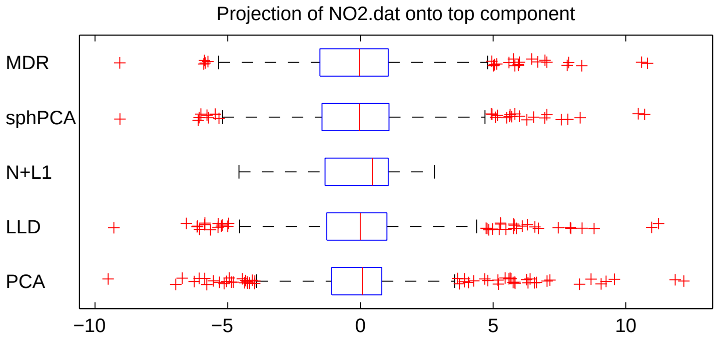

Fig 4. Projection of the Oslo NO_2 data onto the top principal component. *The box surrounds the middle 50% of the data. The vertical line in the box is the median of the data. Each whisker extends either* 1.5 *times the length of the IQR or to the extreme value of the data, and the red crosses beyond the whiskers are the outlying points. The plots are ordered by decreasing IQR.*

Table 2

Statistics for the projected NO_2 data. *The second column gives the interquartile range of the points; the next four columns give some order statistics for projected points. The last column lists the percentage of points lying outside the whiskers in Figure 4.*

Method	**IQR**	**min**	**25th**	**75th**	**max**	**% outliers**
MDR	2.57	−9.07	−1.53	1.05	10.82	5.00%
sphPCA	2.53	−9.06	−1.45	1.08	10.71	5.60%
N+L1	2.38	−4.58	−1.34	1.05	2.79	0.00%
LLD	2.27	−9.29	−1.27	1.00	11.24	7.40%
PCA	1.89	−9.51	−1.08	0.81	12.18	11.00%

robust methods. Except for N+L1, every method identifies a direction with a relatively large number of outliers, which indicates that the data has heavy tails in the directions identified by LLD, MDR, sphPCA, and PCA.

The N+L1 method is unique because it does not identify a direction of large spread *outside* of the middle 50% of the data. While N+L1 does find a direction with large IQR as compared to standard PCA, this direction does not capture much spread in the tail of the data. Given the theoretical guarantees in [8], this fact indicates that the assumptions underlying N+L1, such as corruptions of individual elements of the measurements, are violated in this data.

We note that the approximation ratio for the top MDR component is nearly optimal at 0.978, which shows that we have essentially found a direction of maximum mean deviation.

5.2.3. Iris data

We use Fisher's iris data [21, 22] in this experiment. The data contains 60 observations from three different species of iris: *Iris setosa*, *Iris virginica*, and

Table 3

Order statistics for the projection of the setosa data onto the top principal component. *The second column gives the IQR in Figure 5. The next four columns give some order statistics for the* setosa *points. The last column lists the percentage of* setosa *points further than 1.5 times the IQR left of the 25th percentile or right of the 75th percentile.*

Method	**IQR**	**min**	**25th**	**75th**	**max**	**% outliers**
LLD	0.70	−1.21	−0.41	0.29	1.14	0.00%
Setosa-only PCA	0.70	−1.22	−0.41	0.29	1.14	0.00%
sphPCA	0.69	−1.19	−0.41	0.28	1.13	0.00%
N+L1	0.66	−1.16	−0.40	0.26	1.07	0.00%
MDR	0.37	−0.79	−0.24	0.13	0.53	0.00%
PCA	0.19	−0.60	−0.15	0.04	0.37	6.00%

Iris versicolor. Each observation consists of four measurements, namely sepal length, sepal width, petal length, and petal width.

Fifty of the observations come from the *setosa* flowers. We corrupt these observations with five measurements of *Iris virginica* and five measurements of *Iris versicolor*. We hope that the robust principal components identify a direction of large spread in the bulk of the data, drawn from the *setosa* population. As a baseline comparison, we also calculate the dominant principal component of the *setosa* population without the outlying flowers.

As in Section 5.2.2, we project the data onto the direction of the dominant component. These points are plotted in Figure 5; we distinguish the bulk *setosa* points from the *versicolor* and *virginica* observations, and compute an approximate density of the *setosa* observations by convolving the projected data with a unit volume Gaussian kernel of width $\sigma = 0.2$. Table 3 gives some order statistics of the projections.

The dominant component of LLD, sphPCA, and N+L1 each achieves an IQR at least three times that of PCA. These components do not clearly distinguish among the three populations, indicating that these methods are insensitive to the effect of the outliers. LLD and sphPCA appear the most effective in this situation: the projection onto the LLD and sphPCA components are nearly indistinguishable from the projection onto the standard principal component calculated using only the *setosa* data (for this reason, we do not plot this projection).

Although MDR results in the most modest IQR in the *setosa* among the robust methods, the IQR associated with the MDR component is 1.95 times the IQR of the *setosa* family along the dominant PCA component. Unlike the other robust methods, the MDR component discriminates among the three distinct populations. While it is clear that MDR *does not* reject the influence of the outliers, MDR balances the influence of outliers and the bulk of the data better than PCA.

In this experiment the optimality ratio for MDR is 0.9975, certifying that the MDR component is essentially the direction of maximum mean deviation in the data. Of course, the mean deviation is not insensitive to outliers, merely

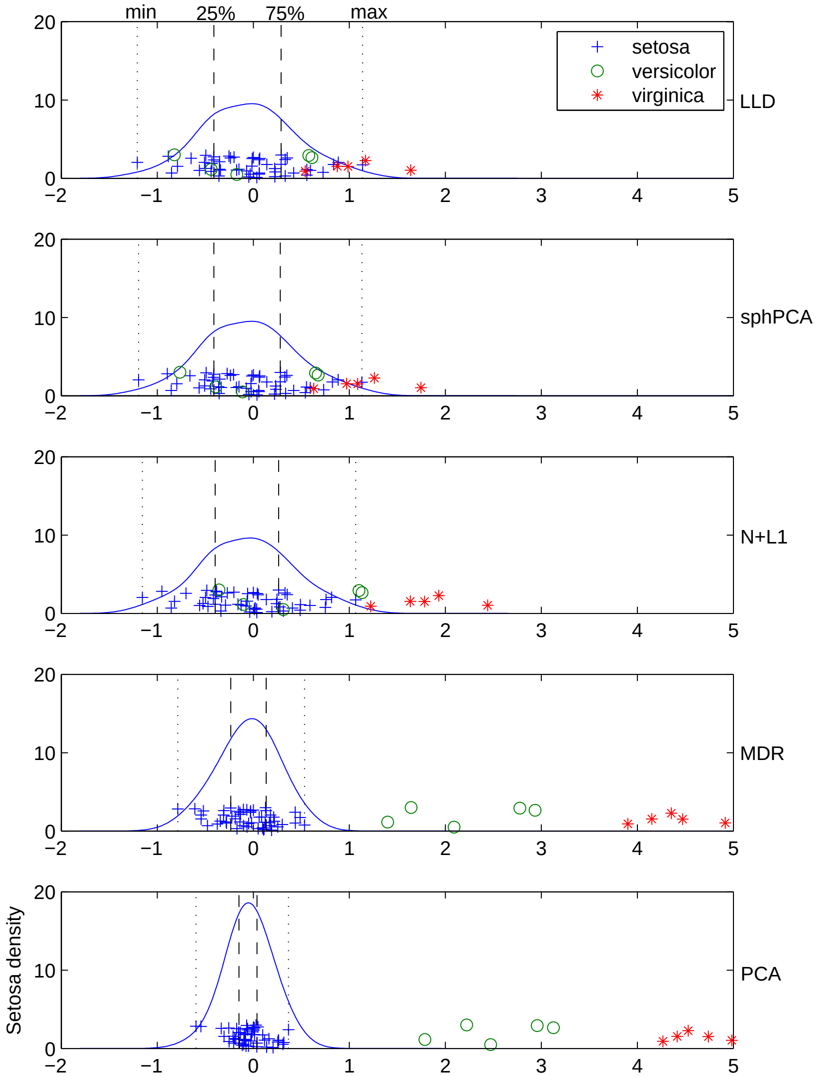

Fig 5. The projections of the iris data onto the top components. *The points are randomly jittered above the zero line for readability. The blue curve represents the approximate local point density of* setosa. *Note that the* LLD *projection is visually indistinguishable from the classical principal component computed using only the setosa population. We sort the plots by decreasing IQR.*

less sensitive than the variance. The rather large distance of the *versicolor* and *virginica* populations to zero means that mean deviation is certainly influenced by these points—yet the IQR of the *setosa* bulk in the direction of the MDR component demonstrates that the balance the influence of the outlying points is very much damped in comparison to classical PCA.

5.3. Regression surface for bus data

In this experiment, we construct a regression surface using multiple principal components. A point is well-described by a surface if its Euclidean distance from the surface is small. The dominant T classical principal components span a T-dimensional regression surface such that the sum of the squared distances of the observations to the plane is minimized. We would hope that robust components describe the bulk of the points better than standard components when outliers contaminate the data. We illustrate this behavior with an experiment of Maronna et al. [40, p. 214], which we augment with additional robust methods.

5.3.1. Experimental setup

Our data consists of $p = 18$ geometric features collected from $n = 218$ bus silhouettes [50] that we arrange into an $n \times p$ matrix $\boldsymbol{X}$. Following Maronna et al., we remove the ninth variable from the data and divide the columns of $\boldsymbol{X}$ by their median absolute deviation (MADN), a robust measure of scale defined as

$$\mathrm{MADN}(\boldsymbol{x}) = \mathrm{median}(|\boldsymbol{x} - \mathrm{median}(\boldsymbol{x})|).$$

We then center the observations by their Euclidean median and compute the top three components using PCA, MDR, LLD, and sphPCA.

For each method, we determine the Euclidean distance from each observation to the orthogonal regression plane spanned by the dominant three components. Figure 6 is a QQ-plot of the ordered distances to the robust hyperplanes against the ordered distances to the PCA hyperplane.

Since the PCA regression surface minimizes the sum of squared distances to the observations, not all of the observations can lie below the one-to-one line. However, a large number of points below the one-to-one line indicates that a robust regression surface explains the bulk of the data better than the classical surface.

5.3.2. Discussion

Figure 6 focuses on the third and fourth quantiles of the data; the first and second quantiles roughly follow the pattern apparent in the third quantile. For clarity, we omit the three most outlying points that would appear in the upper right corner of the figure. Each robust method results in a regression surfaces that explains the data better than PCA for more than 75% of the points. In the third quantile, both N+L1 and sphPCA lose their explanatory advantage over PCA. It is not until the after 95% of the data that MDR and LLD provide worse explanations than PCA.

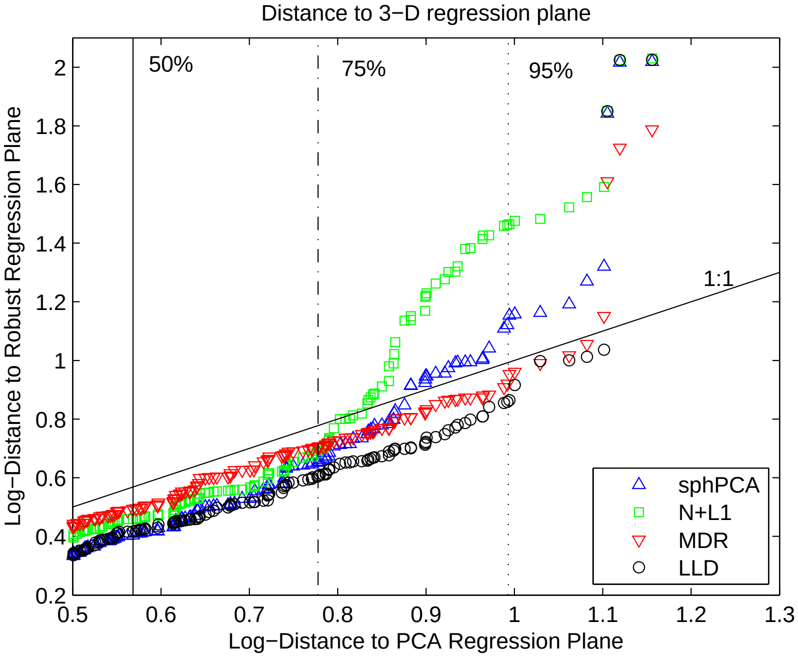

FIG 6. The distance of points to robust regression surfaces as a function of the distance of points to the standard PCA regression surface. *The regression surface is determined by the top three components from each method. Points to the left of the median follow the same generic pattern as points in the third quartile and are therefore omitted. Three extreme points to the right are also omitted.*

MDR explains the bulk of the data less effectively than the other robust methods, yet the final outlying observations are explained better by MDR than the other methods. This indicates that MDR is more sensitive to outlying points than the other robust methods, but is less sensitive to outliers than standard PCA. The optimality ratios for the first three MDR components are, respectively, 0.99999, 0.99992, and 0.97253, implying that MDR essentially succeeds in PP-PCA with the MD scale for this data.

As a reviewer remarked, this experiment does not address the assumptions underlying N+L1. If entries of observations are corrupted throughout the bus data, one would hope that a robust method would determine components that are not influenced by these corruptions. In this case, a large number of observations would lie far from the regression surface if the method is successful. If we view the goal as correcting for entire outlying observations, then considering the distance of an observation to the regression surface is appropriate. This experiment leads us to conclude that MDR, LLD, and sphPCA are better at controlling for entire outlying observations than N+L1, but it does not give us information about how well the methods handle errors within the observations.

6. Conclusions

Our experiments, taken together, suggest that LLD is the most stable and reliable method for robust PCA. While MDR certainly shows robust behavior compared to standard PCA, its performance is markedly worse than LLD under many metrics. Spherical PCA, the method recommended by [39], performs well in most of our experiments. However, LLD performs nearly uniformly better than sphPCA, the exception being the results of Section 5.2.2.

Our theoretical results give us confidence that MDR is very nearly achieving the direction of maximum mean deviation in our experiments, so we can conclude with some confidence that the projection-pursuit approach to robust PCA with the MD-scale is not as robust as the convex approach taken in LLD when a subset of the observations lie near a low-dimensional surface. The main disadvantage of LLD is the computational cost; indeed, both MDR and sphPCA (though not N+L1) tend to find a few principal components significantly more quickly than LLD.

If the goal is to damp, but not ignore, the effect of outliers, then MDR is a good choice: in every experiment, MDR exhibits behavior that is qualitatively more robust than PCA, but still tends to find directions in which the data has large tails. Moreover, our theory gives us confidence that the algorithm essentially achieves the direction of maximum deviation, an easily interpretable formulation for robust PCA.

Appendix A: Proof of Theorem 2.2

This appendix contains the proof of Theorem 2.2 that we repeat below as Theorem A.4. We begin with some supporting results. The following result of Alon and Naor [2, Sec. 4.2] allows us to bound the expectation of $\|\boldsymbol{X}\boldsymbol{v}_\star\|_1$ below. The essence of this result goes back to a 1953 paper of Grothendieck [28]; see also the little Grothendieck theorem in [46, Sec. 5b].

Lemma A.1. *Let $\alpha_\star^2$ be the value of the optimization problem (11) in Algorithm 1. Then $\alpha_\star^2 \geq \|\boldsymbol{X}\boldsymbol{X}^*\|_{\infty\to1}$. Moreover, let $\boldsymbol{y}^{(k)}$ be one of the vectors generated in Step 3. Then $\mathbb{E}\left\|\boldsymbol{X}^*\boldsymbol{y}^{(k)}\right\|_2^2 \geq \frac{2}{\pi}\alpha_\star^2$.*

The claim $\alpha_\star^2 \geq \|\boldsymbol{X}\boldsymbol{X}^*\|_{\infty\to1}$ also follows from our discussion of the semidefinite relaxation in Section 2.4.1. We also need the following proposition.

Proposition A.2. *For each matrix $\boldsymbol{X}$, the identity $\|\boldsymbol{X}\boldsymbol{X}^*\|_{\infty\to1} = \|\boldsymbol{X}\|_{2\to1}^2$ holds.*

Proof. We can express

$$\|\boldsymbol{X}\boldsymbol{X}^*\|_{\infty\to1} = \max_{\substack{\|\boldsymbol{w}\|_\infty=1\\ \|\boldsymbol{y}\|_\infty=1}} \langle \boldsymbol{X}^*\boldsymbol{w}, \boldsymbol{X}^*\boldsymbol{y}\rangle\,.$$

By the conditions for equality in the Cauchy–Schwarz inequality, it follows that we can take $\boldsymbol{w} = \boldsymbol{y}$ above. Hence

$$\|\boldsymbol{X}\boldsymbol{X}^*\|_{\infty\to 1} = \max_{\|\boldsymbol{y}\|_\infty = 1} \|\boldsymbol{X}^*\boldsymbol{y}\|_2^2 = \|\boldsymbol{X}^*\|_{\infty\to 2}^2 = \|\boldsymbol{X}\|_{2\to 1}^2,$$

where the last equality is a standard fact concerning adjoint operators (see [33, p. 232]). □

We use the following variant of the Paley–Zygmund integral inequality [45] to bound the probability that $\|\boldsymbol{X}\boldsymbol{v}_\star\|_1$ is less than its expectation.

Lemma A.3. *Let C be a positive constant and suppose Z is a random variable that satisfies $0 \le Z \le C$. Then, for any scalar $\theta \in [0, 1]$, we have*

$$\mathbb{P}\{Z > \theta\,\mathbb{E}[Z]\} \ge \frac{1}{C}(1-\theta)\,\mathbb{E}[Z].$$

Proof. Split $\mathbb{E}[Z]$ into two integrals, the first over the region $Z \le \theta\,\mathbb{E}[Z]$ and the second over the region $Z > \theta\,\mathbb{E}[Z]$. Notice that the former integral is bounded above by $\theta\,\mathbb{E}[Z]$, while the latter integral is bounded above by $C\,\mathbb{P}\{Z > \theta\,\mathbb{E}[Z]\}$. Simple algebraic manipulation then shows the claim. □

We now restate and prove the main result of Section 2.

Theorem A.4. *Suppose that $\boldsymbol{X}$ is an $n \times p$ matrix, and let K be the number of rounding trials. Let $(\boldsymbol{v}_\star, \alpha_\star)$ be the output of Algorithm 1. Then $\alpha_\star \ge \|\boldsymbol{X}\|_{2\to 1}$. Moreover, for $\theta \in [0, 1]$, the inequality*

$$\|\boldsymbol{X}\boldsymbol{v}_\star\|_1 > \theta\sqrt{\frac{2}{\pi}}\alpha_\star$$

holds except with probability $\mathrm{e}^{-2K(1-\theta^2)/\pi}$.

Proof. Let $\boldsymbol{y} \in \{\pm 1\}^n$ be a sign vector and define $\boldsymbol{v} = \boldsymbol{X}^*\boldsymbol{y}/\|\boldsymbol{X}^*\boldsymbol{y}\|_2$. Then

$$\|\boldsymbol{X}\boldsymbol{v}\|_1 = \|\boldsymbol{X}^*\boldsymbol{y}\|_2^{-1} \max_{\boldsymbol{w}\in\{\pm 1\}^n} \langle \boldsymbol{w}, \boldsymbol{X}\boldsymbol{X}^*\boldsymbol{y}\rangle \ge \|\boldsymbol{X}^*\boldsymbol{y}\|_2$$

where the inequality follows by taking the specific choice $\boldsymbol{w} = \boldsymbol{y}$. In particular, this relation implies that the vectors $\boldsymbol{v}^{(k)} = \boldsymbol{X}^*\boldsymbol{y}^{(k)}/\|\boldsymbol{X}^*\boldsymbol{y}^{(k)}\|_2$ generated in Step 3 of Algorithm 1 satisfy

$$\mathbb{E}\|\boldsymbol{X}\boldsymbol{v}^{(k)}\|_1^2 \ge \mathbb{E}\|\boldsymbol{X}^*\boldsymbol{y}^{(k)}\|_2^2 \ge \frac{2}{\pi}\alpha_\star^2, \tag{24}$$

where the last inequality follows from the second claim in Lemma A.1.

Since $\|\boldsymbol{v}^{(k)}\|_2 = 1$, the quantity $\|\boldsymbol{X}\boldsymbol{v}^{(k)}\|_1^2$ is a positive random variable bounded above by $\|\boldsymbol{X}\|_{2\to 1}^2$. Therefore, inequality (24) and the Paley-Zygmund inequality from Lemma A.3 imply that

$$\mathbb{P}\left\{\|\boldsymbol{X}\boldsymbol{v}^{(k)}\|_1^2 > \theta^2 \cdot \frac{2\alpha_\star^2}{\pi}\right\} \ge (1-\theta^2)\frac{2}{\pi}\cdot\left(\frac{\alpha_\star}{\|\boldsymbol{X}\|_{2\to 1}}\right)^2 \ge \frac{2}{\pi}\cdot(1-\theta^2), \tag{25}$$

where we have used the fact that $\alpha_\star \geq \|\boldsymbol{X}\|_{2\to 1}$ by Proposition A.2 and the first claim of Lemma A.1.

In Step 4 of the algorithm we have chosen $\boldsymbol{v}_\star$ to maximize $\|\boldsymbol{X}\boldsymbol{v}_\star\|_1^2$, so the inequality $\|\boldsymbol{X}\boldsymbol{v}_\star\|_1^2 \leq 2\theta^2/\pi$ holds if and only if $\|\boldsymbol{X}\boldsymbol{v}^{(k)}\|_1 \leq 2\theta^2/\pi$ for all k. Therefore, the independence of $\boldsymbol{v}^{(k)}$ for $k = 1, \ldots, K$ implies

$$\mathbb{P}\left\{\|\boldsymbol{X}\boldsymbol{v}_\star\|_1 \leq \theta\sqrt{\frac{2}{\pi}}\|\boldsymbol{X}\|_{2\to 1}\right\} \leq \left(1 - \frac{2}{\pi}\cdot(1-\theta^2)\right)^K < \mathrm{e}^{-2K(1-\theta^2)/\pi},$$

which completes the claim. □

Acknowledgments

The authors would like to thank the anonymous referees for their thoughtful suggestions, as well as Alex Gittens, Richard Chen, and Stephen Becker for valuable discussions regarding this work.

References

[1] Aldrin, M. (2004). NO2.dat. http://lib.stat.cmu.edu/datasets/NO2.dat.

[2] Alon, N. and Naor, A. (2006). Approximating the cut-norm via Grothendieck's inequality. *SIAM J. Comput.* **35** 787. MR2203567

[3] Bhatia, R. (1997). *Matrix analysis* **169**. Springer Verlag. MR1477662

[4] Brubaker, S. C. (2009). Robust PCA and clustering in noisy mixtures. In *Proceedings of the twentieth Annual ACM-SIAM Symposium on Discrete Algorithms* 1078–1087. SIAM.

[5] Buckheit, J. and Donoho, D. (1995). Wavelab and reproducible research. *Wavelets and Statistics* 55–81.

[6] Burer, S. and Monteiro, R. D. C. (2003). A nonlinear programming algorithm for solving semidefinite programs via low-rank factorization. *Math. Program.* **95** 329–357. MR1976484

[7] Burer, S. and Monteiro, R. D. C. (2004). Local minima and convergence in low-rank semidefinite programming. *Math. Program.* **103** 427–444. MR2166543

[8] Candes, E. J., Li, X., Ma, Y. and Wright, J. (2009). Robust principal component analysis? *J. Assoc. Comput. Mach.* **58**. arXiv:0912.3599.

[9] Chandrasekaran, V., Sanghavi, S., Parrilo, P. A. and Willsky, A. S. (2011). Rank-sparsity incoherence for matrix decomposition. *SIAM J. Opt.* **21** 572–596.

[10] Combettes, P. L. and Wajs, V. R. (2006). Signal recovery by proximal forward-backward splitting. *Multiscale Modeling and Simulation* **4** 1168–1200. MR2203849

[11] Croux, C., Filzmoser, P. and Oliveira, M. R. (2007). Algorithms for projection-pursuit robust principal component analysis. *Chemom. Intell. Lab. Syst.* **87** 218–225.

[12] Croux, C. and Haesbroeck, H. (2000). Principal component analysis based on robust estimators of the covariance or correlation matrix: influence functions and efficiencies. *Biometrika* **87** 603–618. MR1789812

[13] Croux, C. and Ruiz-Gazen, A. (2005). High breakdown estimators for principal components: the projection-pursuit approach revisited. *J. Multivariate Anal.* **95** 206–226. MR2164129

[14] Cui, H. (2003). Asymptotic distributions of principal components based on robust dispersions. *Biometrika* **90** 953–966. MR2024769

[15] Dasgupta, S. (2003). Subspace detection: a robust statistics formulation. In *Learning Theory and Kernel Machines*, (B. Schölkopf and M. Warmuth, eds.). *Lecture Notes in Computer Science* **2777** 734-734. Springer.

[16] De La Torre, F. and Black, M. J. (2003). A framework for robust subspace learning. *Int. J. Comput. Vision* **54** 117–142.

[17] Devlin, S. J., Gnanadesikan, R. and Kettenring, J. R. (1981). Robust estimation of dispersion matrices and principal components. *J. Am. Stat. Assoc.* **76** 354–362.

[18] Dunagan, J. and Vempala, S. (2001). Optimal outlier removal in high-dimensional. In *Proceedings of the thirty-third annual ACM symposium on Theory of computing. STOC '01* 627–636. ACM, New York, NY, USA. MR2120365

[19] Eckstein, J. and Bertsekas, D. P. (1992). On the Douglas–Rachford splitting method and the proximal point algorithm for maximal monotone operators. *Math. Program.* **55** 293–318. MR1168183

[20] Fazel, M. (2002). Matrix rank minimization with applications Dissertation, Stanford University, Stanford, CA.

[21] Fisher, R. A. (1936). The use of multiple measurements in taxonomic problems. *Ann. Eugenic.* **7** 179–188.

[22] Frank, A. and Asuncion, A. (2010). UCI Machine Learning Repository. `http://archive.ics.uci.edu/ml/datasets/Iris`.

[23] Gabay, D. and Mercier, B. (1976). A dual algorithm for the solution of nonlinear variational problems via finite element approximations. *Comp. Math. Appl.* **2** 17–40.

[24] Glowinski, R. and Marocco, A. (1975). Sur l'approximation, par elements finis d'ordre un, et la resolution, par penalisation-dualité, d'une classe de problemes de Dirichlet non lineares. *Revue Française Automat. Informat. Recherche Opérationelle Ser. Rouge Anal. Numér.* **9** 41–76. MR0388811

[25] Gnanadesikan, R. and Kettenring, J. R. (1972). Robust estimates, residuals, and outlier detection with multiresponse data. *Biometrics* **28** 81–124.

[26] Grant, M. and Boyd, S. (2008). Graph implementations for nonsmooth convex programs. In *Recent Advances in Learning and Control*, (V. Blondel, S. Boyd and H. Kimura, eds.). *Lecture Notes in Control and Information Sciences* 95–110. Springer-Verlag Limited, London. `http://stanford.edu/~boyd/graph_dcp.html`. MR2409077

[27] Grant, M. and Boyd, S. (2010). CVX: Matlab Software for Disciplined Convex Programming, version 1.21. http://cvxr.com/cvx.

[28] Grothendieck, A. (1953). Résumé de la théorie métrique des produits tensoriels topologiques (French). *Bol. Soc. Mat. So Paulo* **8** 1–79. MR0094682

[29] Hager, W. W. and Zhang, H. (2006). Algorithm 851: CG_DESCENT, a conjugate gradient method with guaranteed descent. *ACM Trans. Math. Software* **32** 137. MR2272354

[30] Hotelling, H. (1933). Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology* **24** 417–441.

[31] Huber, P. J. (1981). *Robust statistics*, First ed. Wiley, Hoboken, New Jersey. MR0606374

[32] Huber, P. J. and Ronchetti, E. (2009). *Robust statistics*, Second ed. Wiley, Hoboken, New Jersey. MR2488795

[33] Kreyszig, E. (1989). *Introductory functional analysis with applications. Wiley Classics Library*. Wiley, USA. MR0992618

[34] Kwak, N. (2008). Principal component analysis based on L1-norm maximization. *IEEE Trans. Pattern Anal. Mach. Intell.* 1672–1680.

[35] Li, G. and Chen, Z. (1985). Projection-pursuit approach to robust dispersion matrices and principal components: primary theory and Monte Carlo. *J. Am. Stat. Assoc.* **80** 759–766.

[36] Lin, Z., Chen, M., Wu, L. and Ma, Y. (2009). The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices Technical Report, UIUC UILU-ENG-09-2215. arXiv:1009.5055.

[37] Liu, G., Lin, Z. and Yu, Y. (2010). Robust subspace segmentation by low-rank representation. In *Proceedings of the 26th International Conference on Machine Learning (ICML)*. Citeseer.

[38] Locantore, N., Marron, J. S., Simpson, D. G., Tripoli, N., Zhang, J. T. and Cohen, K. L. (1999). Robust principal component analysis for functional data. *Test* **8** 1–73. MR1707596

[39] Maronna, R. A. (2005). Principal components and orthogonal regression based on robust scales. *Technometrics* **47** 264–273. MR2164700

[40] Maronna, R. A., Martin, D. R. and Yohai, V. J. (2006). *Robust statistics: theory and methods. Wiley Series in Probability and Statistics*. Wiley, Hoboken, NJ. MR2238141

[41] McCoy, M. and Tropp, J. A. (2010). Online Code. http://users.cms.caltech.edu/~mccoy/code/robustPCA_code2.tar.gz.

[42] Montgomery, D. C., Peck, E. A. and Vining, G. G. (2006). *Introduction to Linear Regression Analysis. Wiley Series in Probability and Statistics*. Wiley, Hoboken, NJ. MR1820113

[43] Nemirovski, A. (2007). Sums of random symmetric matrices and quadratic optimization under orthogonality constraints. *Math. Program.* **109** 283–317. MR2295145

[44] Nesterov, Y. E. (1998). Semidefinite relaxation and nonconvex quadratic optimization. *Optim. Methods Softw.* **9** 141–160. MR1618100

[45] Paley, R. E. A. C. and Zygmund, A. (1932). A note on analytic functions in the unit circle. *Math. Proc. Cambridge Philos. Soc.* **28** 266–272.

[46] Pisier, G. (1986). *Factorization of linear operators and geometry of Banach spaces. Regional Conference Series in Mathematics*. American Mathematical Society, Providence, RI. MR0829919

[47] Rao, B. D. and Kreutz-Delgado, K. (1998). Sparse solutions to linear inverse problems with multiple measurement vectors. *Proceedings of the 8th IEEE Digital Signal Processing Workshop*.

[48] Rockafellar, R. T. (1970). *Convex analysis*. *Princeton Mathematical Series*. Princeton University Press. MR0274683

[49] Rohn, J. (2000). Computing the $\|\cdot\|_{\infty\to1}$ norm is NP-hard. *Linear and Multilinear Algebra* **47** 195–204. MR1785027

[50] Siebert, J. P. (1987). Vehicle recognition using rule based methods. Turing Institute Research Memorandum TIRM-87-018.

[51] So, A. M.-C. (2009). Improved approximation bound for quadratic optimization problems with orthogonality constraints. *Symposium on Discrete Algorithms*.

[52] Stoer, J. and Bulirsch, R. (2002). *Introduction to numerical analysis*. Springer, New York, NY. MR1923481

[53] Tukey, J. W. (1960). A survey of sampling from contaminated distributions. In *Contributions to probability and statistics: essays in honor of Harold Hotelling* (I. Olkin, ed.) 448–474. Stanford University Press, Stanford, CA. MR0120720

[54] Watson, G. (1992). Characterization of the subdifferential of some matrix norms. *Linear Algebra Appl.* **170** 33–45. MR1160950

[55] Xu, H., Caramanis, C. and Mannor, S. (2010). Principal component analysis with contaminated data: the high dimensional case. In *COLT 2010* 1–37.

[56] Xu, H., Caramanis, C. and Sanghavi, S. (2010). Robust PCA via outlier pursuit. *IEEE Trans. Inform. Theory, to appear* 1–24. arXiv:1010.4237.

[57] Xu, H., Caramanis, C. and Sanghavi, S. (2010). Robust PCA via outlier pursuit. In *NIPS 23* (J. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R. S. Zemel and A. Culotta, eds.) 2496–2504.