Robust PCA via Outlier Pursuit

Huan Xu
Electrical and Computer Engineering
University of Texas at Austin
huan.xu@mail.utexas.edu

Constantine Caramanis
Electrical and Computer Engineering
University of Texas at Austin
cmcaram@ece.utexas.edu

Sujay Sanghavi
Electrical and Computer Engineering
University of Texas at Austin
sanghavi@mail.utexas.edu

Abstract

Singular Value Decomposition (and Principal Component Analysis) is one of the most widely used techniques for dimensionality reduction: successful and efficiently computable, it is nevertheless plagued by a well-known, well-documented sensitivity to outliers. Recent work has considered the setting where each point has a few arbitrarily corrupted components. Yet, in applications of SVD or PCA such as robust collaborative filtering or bioinformatics, malicious agents, defective genes, or simply corrupted or contaminated experiments may effectively yield entire points that are completely corrupted.

We present an efficient convex optimization-based algorithm we call Outlier Pursuit, that under some mild assumptions on the uncorrupted points (satisfied, e.g., by the standard generative assumption in PCA problems) recovers the *exact* optimal low-dimensional subspace, and identifies the corrupted points. Such identification of corrupted points that do not conform to the low-dimensional approximation, is of paramount interest in bioinformatics and financial applications, and beyond. Our techniques involve matrix decomposition using nuclear norm minimization, however, our results, setup, and approach, necessarily differ considerably from the existing line of work in matrix completion and matrix decomposition, since we develop an approach to recover the correct *column space* of the uncorrupted matrix, rather than the exact matrix itself.

1 Introduction

This paper is about the following problem: suppose we are given a large *data matrix* M, and we know it can be decomposed as

$$M = L_0 + C_0,$$

where L_0 is a low-rank matrix, and C_0 is non-zero in only a fraction of the columns. Aside from these broad restrictions, both components are arbitrary. In particular we do not know the rank (or the row/column space) of L_0, or the number and positions of the non-zero columns of C_0. Can we recover the column-space of the low-rank matrix L_0, and the identities of the non-zero columns of C_0, *exactly* and efficiently?

We are primarily motivated by Principal Component Analysis (PCA), arguably the most widely used technique for dimensionality reduction in statistical data analysis. The canonical PCA problem [1], seeks to find the best (in the least-square-error sense) low-dimensional subspace approximation to high-dimensional points. Using the Singular Value Decomposition (SVD), PCA finds the lower-dimensional approximating subspace by forming a low-rank approximation to the data matrix, formed by considering each point as a column; the output of PCA is the (low-dimensional) column space of this low-rank approximation.

It is well known (e.g., [2–4]) that standard PCA is extremely fragile to the presence of *outliers*: even a single corrupted point can arbitrarily alter the quality of the approximation. Such non-probabilistic or persistent data corruption may stem from sensor failures, malicious tampering, or the simple fact that some of the available data may not conform to the presumed low-dimensional source / model. In terms of the data matrix, this means that most of the column vectors will lie in a low-dimensional space – and hence the corresponding matrix L_0 will be low-rank – while the remaining columns will be outliers – corresponding to the column-sparse matrix C. The natural question in this setting is to ask if we can still (exactly or near-exactly) recover the column space of the uncorrupted points, and the identities of the outliers. This is precisely our problem.

Recent years have seen a lot of work on both robust PCA [3, 5–12], and on the use of convex optimization for recovering low-dimensional structure [4, 13–15]. Our work lies at the intersection of these two fields, but has several significant differences from work in either space. We compare and relate our work to existing literature, and expand on the differences, in Section 3.3.

2 Problem Setup

The precise PCA with outlier problem that we consider is as follows: we are given n points in p-dimensional space. A fraction $1-\gamma$ of the points lie on a r-dimensional *true* subspace of the ambient $\mathbb{R}^p$, while the remaining γn points are *arbitrarily* located – we call these outliers/corrupted points. We do not have any prior information about the true subspace or its dimension r. Given the set of points, we would like to learn *(a)* the true subspace and *(b)* the identities of the outliers.

As is common practice, we collate the points into a $p \times n$ *data matrix* M, each of whose columns is one of the points, and each of whose rows is one of the p coordinates. It is then clear that the data matrix can be decomposed as

$$M = L_0 + C_0.$$

Here L_0 is the matrix corresponding to the non-outliers; thus $rank(L_0) = r$. Consider its Singular Value Decomposition (SVD)

$$L_0 = U_0 \Sigma_0 V_0^\top. \tag{1}$$

Thus it is clear that the columns of U_0 form an orthonormal basis for the r-dimensional true subspace. Also note that at most $(1-\gamma)n$ of the columns of L_0 are non-zero (the rest correspond to the outliers). C_0 is the matrix corresponding to the non-outliers; we will denote the set of non-zero columns of C_0 by $\mathcal{I}_0$, with $|\mathcal{I}_0| = \gamma n$. These non-zero columns are completely arbitrary.

With this notation, out intent is to *exactly* recover the column space of L_0, and the set of outliers $\mathcal{I}_0$. Clearly, this is not always going to be possible (regardless of the algorithm used) and thus we need to impose a few additional assumptions. We develop these in Section 2.1 below.

We are also interested in the noisy case, where

$$M = L_0 + C_0 + N,$$

and N corresponds to any additional noise. In this case we are interested in approximate identification of both the true subspace and the outliers.

2.1 Incoherence: When does exact recovery make sense?

In general, our objective of splitting a low-rank matrix from a column-sparse one is not always a well defined one. As an extreme example, consider the case where the data matrix M is non-zero in only one column. Such a matrix is both low-rank and column-sparse, thus the problem is unidentifiable. To make the problem meaningful, we need to impose that the low-rank matrix L_0 cannot itself be column-sparse as well. This is done via the following *incoherence condition*.

Definition: A matrix $L \in \mathbb{R}^{p\times n}$ with SVD as in (1), and $(1-\gamma)n$ of whose columns are non-zero, is said to be *column-incoherent* with parameter μ if

$$\max_i \|V^\top \mathbf{e}_i\|^2 \leq \frac{\mu r}{(1-\gamma)n}$$

where $\{\mathbf{e}_i\}$ are the coordinate unit vectors.

Thus if V has a column aligned with a coordinate axis, then $\mu = (1-\gamma)n/r$. Similarly, if V is perfectly incoherent (e.g. if $r = 1$ and every non-zero entry of V has magnitude $1/\sqrt{(1-\gamma)n}$) then $\mu = 1$.

In the standard PCA setup, if the points are generated by some low-dimensional isometric Gaussian distribution, then with high probability, one will have $\mu = O(\max(1, \log(n)/r))$ [16]. Alternatively, if the points are generated by a uniform distribution over a *bounded* set, then $\mu = \Theta(1)$.

A small incoherence parameter μ essentially enforces that the matrix L_0 will have column support that is spread out. Note that this is quite natural from the application perspective. Indeed, if the left hand side is as big as 1, it essentially means that one of the directions of the column space which we wish to recover, is defined by only a single observation. Given the regime of a constant fraction of *arbitrarily chosen* and *arbitrarily corrupted* points, such a setting is not meaningful. Indeed, having a small incoherence μ is an assumption made in all methods based on nuclear norm minimization up to date [4, 15–17].

We would like to identify the outliers, which can be arbitrary. However, clearly an "outlier" point that lies in the true subspace is a meaningless concept. Thus, in matrix terms, we require that every column of C_0 does not lie in the column space of L_0.

The parameters μ and γ are not required for the execution of the algorithm, and *do not need to be known a priori*. They only arise in the analysis of our algorithm's performance.

Other Notation and Preliminaries: Capital letters such as A are used to represent matrices, and accordingly, A_i denotes the i^{th} column vector. Letters U, V, $\mathcal{I}$ and their variants (complements, subscripts, etc.) are reserved for column space, row space and column support respectively. There are four associated projection operators we use throughout. The projection onto the column space, U, is denoted by $\mathcal{P}_U$ and given by $\mathcal{P}_U(A) = UU^\top A$, and similarly for the row-space $\mathcal{P}_V(A) = AVV^\top$. The matrix $\mathcal{P}_\mathcal{I}(A)$ is obtained from A by setting column A_i to zero for all $i \notin \mathcal{I}$. Finally, $\mathcal{P}_T$ is the projection to the space spanned by U and V, and given by $\mathcal{P}_T(\cdot) = \mathcal{P}_U(\cdot) + \mathcal{P}_V(\cdot) - \mathcal{P}_U\mathcal{P}_V(\cdot)$. Note that $\mathcal{P}_T$ depends on U and V, and we suppress this notation wherever it is clear which U and V we are using. The complementary operators, $\mathcal{P}_{U^\perp}, \mathcal{P}_{V^\perp}, \mathcal{P}_{T^\perp}$ and $\mathcal{P}_{\mathcal{I}^c}$ are defined as usual. The same notation is also used to represent a subspace of matrices: e.g., we write $A \in \mathcal{P}_U$ for any matrix A that satisfies $\mathcal{P}_U(A) = A$. Five matrix norms are used: $\|A\|_*$ is the nuclear norm, $\|A\|$ is the spectral norm, $\|A\|_{1,2}$ is the sum of ℓ_2 norm of the columns A_i, $\|A\|_{\infty,2}$ is the largest ℓ_2 norm of the columns, and $\|A\|_F$ is the Frobenius norm. The only vector norm used is $\|\cdot\|_2$, the ℓ_2 norm. Depending on the context, I is either the unit matrix, or the identity operator; $\mathbf{e}_i$ is the i^{th} base vector. The SVD of L_0 is $U_0\Sigma_0V_0^\top$. The rank of L_0 is denoted as r, and we have $\gamma \triangleq |\mathcal{I}_0|/n$, i.e., the fraction of outliers.

3 Main Results and Consequences

While we do not recover the matrix L_0, we show that the goal of PCA can be attained: even under our strong corruption model, with a constant fraction of points corrupted, we show that we can – under very weak assumptions – *exactly* recover both the column space of L_0 (i.e the low-dimensional space the uncorrupted points lie on) and the column support of C_0 (i.e. the identities of the outliers), from M. If there is additional noise corrupting the data matrix, i.e. if we have $M = L_0 + C_0 + N$, a natural variant of our approach finds a good approximation.

3.1 Algorithm

Given data matrix M, our algorithm, called *Outlier Pursuit*, generates *(a)* a matrix $\hat{U}$, with orthonormal rows, that spans the low-dimensional true subspace we want to recover, and *(b)* a set of column indices $\hat{\mathcal{I}}$ corresponding to the outlier points. To ensure success, one choice of the tuning parameter is $\lambda = \frac{3}{7\sqrt{\gamma n}}$, as Theorem 1 below suggests.

While in the noiseless case there are simple algorithms with similar performance, the benefit of the algorithm, and of the analysis, is extension to more realistic and interesting situations where in

Algorithm 1 Outlier Pursuit

```
Find $(\hat{L}, \hat{C})$, the optimum of the following convex optimization program.
    $\begin{array}{ll} \text{Minimize:} & \|L\|_* + \lambda\|C\|_{1,2} \\ \text{Subject to:} & M = L + C \end{array}$    (2)
Compute SVD $\hat{L} = U_1 \Sigma_1 V_1^\top$ and output $\hat{U} = U_1$.
Output the set of non-zero columns of $\hat{C}$, i.e. $\hat{I} = \{j : \hat{c}_{ij} \neq 0 \text{ for some } i\}$.
```

addition to gross corruption of some samples, there is additional noise. Adapting the Outlier Pursuit algorithm, we have the following variant for the noisy case.

$$\textbf{Noisy Outlier Pursuit:} \qquad \begin{array}{ll} \text{Minimize:} & \|L\|_* + \lambda\|C\|_{1,2} \\ \text{Subject to:} & \|M - (L + C)\|_F \leq \varepsilon \end{array} \tag{3}$$

Outlier Pursuit (and its noisy variant) is a convex surrogate for the following natural (but combinatorial and intractable) first approach to the recovery problem:

$$\begin{array}{ll} \text{Minimize:} & \text{rank}(L) + \lambda\|C\|_{0,c} \\ \text{Subject to:} & M = L + C \end{array} \tag{4}$$

where $\|\cdot\|_{0,c}$ stands for the number of non-zero columns of a matrix.

3.2 Performance

We show that under rather weak assumptions, Outlier Pursuit exactly recovers the column space of the low-rank matrix L_0, and the identities of the non-zero columns of outlier matrix C_0. The formal statement appears below.

Theorem 1 (Noiseless Case). *Suppose we observe $M = L_0 + C_0$, where L_0 has rank r and incoherence parameter μ. Suppose further that C_0 is supported on at most γn columns. Any output to Outlier Pursuit recovers the column space exactly, and identifies exactly the indices of columns corresponding to outliers not lying in the recovered column space, as long as the fraction of corrupted points, γ, satisfies*

$$\frac{\gamma}{1-\gamma} \leq \frac{c_1}{\mu r}, \tag{5}$$

where $c_1 = \frac{9}{121}$. This can be achieved by setting the parameter λ in outlier pursuit to be $\frac{3}{7\sqrt{\gamma n}}$ – indeed it holds for any λ in a specific range which we provide below.

For the case where in addition to the corrupted points, we have noisy observations, $\tilde{M} = M + W$, we have the following result.

Theorem 2 (Noisy Case). *Suppose we observe $\tilde{M} = M + N = L_0 + C_0 + N$, where*

$$\frac{\gamma}{1-\gamma} \leq \frac{c_2}{\mu r} \tag{6}$$

with $c_2 = \frac{9}{1024}$, and $\|N\|_F \leq \varepsilon$. Let the output of Noisy Outlier Pursuit be L', C'. Then there exists $\tilde{L}, \tilde{C}$ such that $M = \tilde{L} + \tilde{C}$, $\tilde{L}$ has the correct column space, and $\tilde{C}$ the correct column support, and

$$\|L' - \tilde{L}\|_F \leq 10\sqrt{n}\varepsilon; \quad \|C' - \tilde{C}\|_F \leq 9\sqrt{n}\varepsilon; .$$

The conditions in this theorem are essentially tight in the following scaling sense (i.e., up to universal constants). If there is no additional structure imposed, beyond what we have stated above, then up to scaling, in the noiseless case, Outlier Pursuit can recover from as many outliers (i.e., the same fraction) as any possible algorithm with arbitrary complexity. In particular, it is easy to see that if the rank of the matrix L_0 is r, and the fraction of outliers satisfies $\gamma \geq 1/(r+1)$, then the problem is not identifiable, i.e., no algorithm can separate authentic and corrupted points.[1]

[1] Note that this is no longer true in the presence of stronger assumptions, e.g., isometric distribution, on the authentic points [12].

3.3 Related Work

Robust PCA has a long history (e.g., [3, 5–11]). Each of these algorithms either performs standard PCA on a robust estimate of the covariance matrix, or finds directions that maximize a robust estimate of the variance of the projected data. These algorithms seek to *approximately* recover the column space, and moreover, no existing approach attempts to identify the set of outliers. This outlier identification, while outside the scope of traditional PCA algorithms, is important in a variety of applications such as finance, bio-informatics, and more.

Many existing robust PCA algorithms suffer two pitfalls: performance degradation with dimension increase, and computational intractability. To wit, [18] shows several robust PCA algorithms including M-estimator [19], Convex Peeling [20], Ellipsoidal Peeling [21], Classical Outlier Rejection [22], Iterative Deletion [23] and Iterative Trimming [24] have breakdown points proportional to the inverse of dimensionality, and hence are useless in the high dimensional regime we consider.

Algorithms with non-diminishing breakdown point, such as Projection-Pursuit [25] are non-convex or even combinatorial, and hence computationally intractable (NP-hard) as the size of the problem scales. In contrast to these, the performance of Outlier Pursuit does not depend on p, and can be solved in polynomial time.

Algorithms based on nuclear norm minimization to recover low rank matrices are now standard, since the seminal paper [14]. Recent work [4, 15] has taken the nuclear norm minimization approach to the decomposition of a low-rank matrix and an overall sparse matrix. At a high level, these papers are close in spirit to ours. However, there are critical differences in the problem setup, the results, and in key analysis techniques. First, these algorithms fail in our setting as they cannot handle outliers – entire columns where every entry is corrupted. Second, from a technical and proof perspective, all the above works investigate *exact* signal recovery – the intended outcome is known ahead of time, and one just needs to investigate the conditions needed for success. In our setting however, the convex optimization cannot recover L_0 itself exactly. This requires an auxiliary "oracle problem" as well as different analysis techniques on which we elaborate below.

4 Proof Outline and Comments

In this section we provide an outline of the proof of Theorem 1. The full proofs of all theorems appear in a full version available online [26]. The proof follows three main steps

1. Identify the first-order necessary and sufficient conditions, for any pair (L', C') to be the optimum of the convex program (2).
2. Consider a candidate pair $(\hat{L}, \hat{C})$ that is the optimum of an alternate optimization problem, often called the "oracle problem". The oracle problem ensures that the pair $(\hat{L}, \hat{C})$ has the desired column space and column support, respectively.
3. Show that this $(\hat{L}, \hat{C})$ is the optimum of Outlier Pursuit.

We remark that the aim of the matrix recovery papers [4, 15, 16] was exact recovery of the *entire* matrix, and thus the optimality conditions required are clear. Since our setup precludes exact recovery of L_0 and C_0, [2] our optimality conditions must imply the optimality for Outlier Pursuit of an as-of-yet-undetermined pair $(\hat{L}, \hat{C})$, the solution to the oracle problem. We now elaborate.

Optimality Conditions: We now specify the conditions a candidate optimum needs to satisfy; these arise from the standard subgradient conditions for the norms involved. Suppose the pair (L', C') is a feasible point of (2), i.e. we have that $L' + C' = M$. Let the SVD of L' be given by $L' = U'\Sigma'V'^{\top}$. For any matrix X, define $\mathcal{P}_{T'}(X) := U'U'^{\top}X + XV'V'^{\top} - U'U'^{\top}XV'V'^{\top}$, the projection of X onto matrices that share the same column space or row space with L'.

Let $\mathcal{I}'$ be the set of non-zero columns of C', and let H' be the column-normalized version of C'. That is, column $H'_i = \frac{C'_i}{\|C'_i\|_2}$ for all $i \in \mathcal{I}'$, and $H'_i = 0$ for all $i \notin \mathcal{I}'$. Finally, for any matrix X let $\mathcal{P}_{\mathcal{I}'}(X)$ denote the matrix with all columns in $\mathcal{I}'^c$ set to 0, and the columns in $\mathcal{I}'$ left as-is.

[2] The optimum $\hat{L}$ of (2) will be non-zero in every column of C_0 that is not *orthogonal* to L_0's column space.

Proposition 1. *With notation as above, L', C' is an optimum of the Outlier Pursuit progam (2) if there exists a Q such that*

$$\begin{array}{ll} \mathcal{P}_{T'}(Q) = U'V' & \|Q - \mathcal{P}_{T'}(Q)\| \leq 1 \\ \mathcal{P}_{\mathcal{I}'}(Q) = \lambda H' & \|Q - \mathcal{P}_{\mathcal{I}'}(Q)\|_{\infty,2} \leq \lambda. \end{array} \tag{7}$$

Further, if both inequalities above are strict, dubbed Q strictly satisfies (7)*, then (L', C') is the unique optimum.*

Note that here $\|\cdot\|$ is the spectral norm (i.e. largest singular value) and $\|\cdot\|_{\infty,2}$ is the magnitude – i.e. ℓ_2 norm – of the column with the largest magnitude.

Oracle Problem: We develop our candidate solution $(\hat{L}, \hat{C})$ by considering the alternate optimization problem where we add constraints to (2) based on what we *hope* its optimum should be. In particular, recall the SVD of the true $L_0 = U_0 \Sigma_0 V_0^\top$ and define for any matrix X the projection onto the space of all matrices with column space contained in U_0 as $\mathcal{P}_{U_0}(X) := U_0 U_0^\top X$. Similarly for the column support $\mathcal{I}_0$ of the true C_0, define the projection $\mathcal{P}_{\mathcal{I}_0}(X)$ to be the matrix that results when all the columns in $\mathcal{I}_0^c$ are set to 0.

Note that U_0 and $\mathcal{I}_0$ above correspond to the *truth*. Thus, with this notation, we would like the optimum of (2) to satisfy $\mathcal{P}_{U_0}(\hat{L}) = \hat{L}$, as this is nothing but the fact that $\hat{L}$ has recovered the true subspace. Similarly, having $\hat{C}$ satisfy $\mathcal{P}_{\mathcal{I}_0}(\hat{C}) = \hat{C}$ means that we have succeeded in identifying the outliers. The oracle problem arises by *imposing* these as additional constraints in (2). Formally:

$$\textbf{Oracle Problem:} \qquad \begin{array}{ll} \text{Minimize:} & \|L\|_* + \lambda \|C\|_{1,2} \\ \text{Subject to:} & M = L + C;\ \mathcal{P}_{U_0}(L) = L;\ \mathcal{P}_{\mathcal{I}_0}(C) = C. \end{array} \tag{8}$$

Obtaining Dual Certificates for Outlier Pursuit: We now construct a dual certificate of $(\hat{L}, \hat{C})$ to establish Theorem 1. Let the SVD of $\hat{L}$ be $\hat{U}\hat{\Sigma}\hat{V}^\top$. It is easy to see that there exists an orthonormal matrix $\overline{V} \in \mathbb{R}^{r \times n}$ such that $\hat{U}\hat{V}^\top = U_0 \overline{V}^\top$, where U_0 is the column space of L_0. Moreover, it is easy to show that $\mathcal{P}_{\hat{U}}(\cdot) = \mathcal{P}_{U_0}(\cdot)$, $\mathcal{P}_{\hat{V}}(\cdot) = \mathcal{P}_{\overline{V}}$, and hence the operator $\mathcal{P}_{\hat{T}}$ defined by $\hat{U}$ and $\hat{V}$, obeys $\mathcal{P}_{\hat{T}}(\cdot) = \mathcal{P}_{U_0}(\cdot) + \mathcal{P}_{\overline{V}}(\cdot) - \mathcal{P}_{U_0}\mathcal{P}_{\overline{V}}(\cdot)$. Let $\hat{H}$ be the matrix satisfying that $\mathcal{P}_{\mathcal{I}_0^c}(\hat{H}) = 0$ and $\forall i \in \mathcal{I}_0$, $\hat{H}_i = \hat{C}_i / \|\hat{C}_i\|_2$.

Define matrix $G \in \mathbb{R}^{r \times r}$ as

$$G \triangleq \mathcal{P}_{\mathcal{I}_0}(\overline{V}^\top)(\mathcal{P}_{\mathcal{I}_0}(\overline{V}^\top))^\top = \sum_{i \in \mathcal{I}_0} [(\overline{V}^\top)_i][(\overline{V}^\top)_i]^\top,$$

and constant $c \triangleq \|G\|$. Further define matrices $\Delta_1 \triangleq \lambda \mathcal{P}_{U_0}(\hat{H})$, and

$$\Delta_2 \triangleq \mathcal{P}_{U_0^\perp}\mathcal{P}_{\mathcal{I}_0^c}\mathcal{P}_{\overline{V}}\big[I + \sum_{i=1}^{\infty}(\mathcal{P}_{\overline{V}}\mathcal{P}_{\mathcal{I}_0}\mathcal{P}_{\overline{V}})^i\big]\mathcal{P}_{\overline{V}}(\lambda\hat{H}) = \mathcal{P}_{\mathcal{I}_0^c}\mathcal{P}_{\overline{V}}\big[I + \sum_{i=1}^{\infty}(\mathcal{P}_{\overline{V}}\mathcal{P}_{\mathcal{I}_0}\mathcal{P}_{\overline{V}})^i\big]\mathcal{P}_{\overline{V}}\mathcal{P}_{U_0^\perp}(\lambda\hat{H}).$$

Then we can define the dual certificate for strict optimality of the pair $(\hat{L}, \hat{C})$.

Proposition 2. *If $c < 1$, $\frac{\gamma}{1-\gamma} < \frac{(1-c)^2}{(3-c)^2 \mu r}$, and $\frac{(1-c)\sqrt{\frac{\mu r}{1-\gamma}}}{\sqrt{n}(1-c-\sqrt{\frac{\gamma}{1-\gamma}\mu r})} < \lambda < \frac{1-c}{(2-c)\sqrt{n\gamma}}$, then $Q \triangleq U_0\overline{V}^\top + \lambda\hat{H} - \Delta_1 - \Delta_2$ strictly satisfies Condition (7), i.e., it is the dual certificate.*

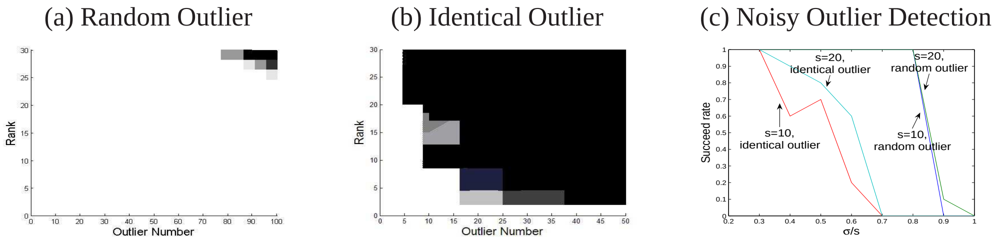
Consider the (much) simpler case where the corrupted columns are assumed to be orthogonal to the column space of L_0 which we seek to recover. Indeed, in that setting, where $V_0 = \hat{V} = \overline{V}$, we automatically satisfy the condition $\mathcal{P}_{\mathcal{I}_0} \bigcap \mathcal{P}_{V_0} = \{0\}$. In the general case, we require the condition $c < 1$ to recover the same property. Moreover, considering that the columns of H are either zero, or defined as normalizations of the columns of matrix C (i.e., normalizations of outliers), that $\mathcal{P}_{U_0}(H) = \mathcal{P}_{V_0}(H) = \mathcal{P}_{T_0}(H) = 0$, is immediate, as is the condition that $\mathcal{P}_{\mathcal{I}_0}(U_0 V_0^\top) = 0$.

For the general, non-orthogonal case, however, we require the matrices Δ_1 and Δ_2 to obtain these equalities, and the rest of the dual certificate properties. In the full version [26] we show in detail how these ideas and the oracle problem, are used to construct the dual certificate Q. Extending these ideas, we then quickly obtain the proof for the noisy case.

5 Implementation issue and numerical experiments

Solving nuclear-norm minimizations naively requires use of general purpose SDP solvers, which unfortunately still have questionable scaling capabilities. Instead, we use the *proximal gradient algorithms* [27], a.k.a., Singular Value Thresholding [28] to solve Outlier Pursuit. The algorithm converges with a rate of $O(k^{-2})$ where k is the number of iterations, and in each iteration, it involves a singular value decomposition and thresholding, therefore, requiring significantly less computational time than interior point methods.

Our first experiment investigates the phase-transition property of Outlier Pursuit, using randomly generated synthetic data. Fix $n = p = 400$. For different r and number of outliers γn, we generated matrices $A \in \mathbb{R}^{p\times r}$ and $B \in \mathbb{R}^{(n-\gamma n)\times r}$ where each entry is an independent $\mathcal{N}(0,1)$ random variable, and then set $L^* := A \times B^\top$ (the "clean" part of M). Outliers, $C^* \in \mathbb{R}^{\gamma n \times p}$ are generated either *neutrally*, where each entry of C^* is iid $\mathcal{N}(0,1)$, or *adversarial*, where every column is an identical copy of a random Gaussian vector. Outlier Pursuit succeeds if $\hat{C} \in \mathcal{P}_{\mathcal{I}}$, and $\hat{L} \in \mathcal{P}_U$. Note that if a lot of outliers span a same direction, it would be difficult to identify whether they are all outliers, or just a new direction of the true space. Indeed, such a setup is order-wise worst, as we proved in the full version [26] a matching lower bound is achieved when all outliers are identical.


(a) Random Outlier

(b) Identical Outlier

(c) Noisy Outlier Detection

Figure 1: Complete Observation: Results averaged over 10 trials.

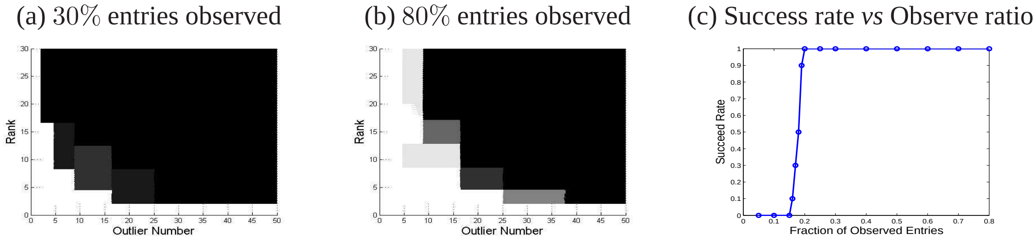
Figure 1 shows the phase transition property. We represent success in gray scale, with white denoting success, and black failure. When outliers are random (easier case) Outlier Pursuit succeeds even when $r = 20$ with 100 outliers. In the adversarial case, we observe a phase transition: Outlier Pursuit succeeds when $r \times \gamma$ is small, and fails otherwise, consistent with our theory's predictions. We then fix $r = \gamma n = 5$ and examine the outlier identification ability of Outlier Pursuit with noisy observations. We scale each outlier so that the ℓ_2 distance of the outlier to the span of true samples equals a pre-determined value s. Each true sample is thus corrupted with a Gaussian random vector with an ℓ_2 magnitude σ. We perform (noiseless) Outlier Pursuit on this noisy observation matrix, and claim that the algorithm successfully identifies outliers if for the resulting $\hat{C}$ matrix, $\|\hat{C}_j\|_2 < \|\hat{C}_i\|_2$ for all $j \notin \mathcal{I}$ and $i \in \mathcal{I}$, i.e., there exists a threshold value to separate out outliers. Figure 1 (c) shows the result: when $\sigma/s \leq 0.3$ for the identical outlier case, and $\sigma/s \leq 0.7$ for the random outlier case, Outlier Pursuit correctly identifies the outliers.

We further study the case of decomposing M under incomplete observation, which is motivated by *robust collaborative filtering*: we generate M as before, but only observe each entry with a given probability (independently). Letting Ω be the set of observed entries, we solve

$$\text{Minimize:} \quad \|L\|_* + \lambda\|C\|_{1,2}; \qquad \text{Subject to:} \quad \mathcal{P}_\Omega(L+C) = \mathcal{P}_\Omega(M). \tag{9}$$

The same success condition is used. Figure 2 shows a very promising result: the successful decomposition rate under incomplete observation is close to the complete observation case even when only 30% of entries are observed. Given this empirical result, a natural direction of future research is to understand theoretical guarantee of (9) in the incomplete observation case.

Next we report some experiment results on the USPS digit data-set. The goal of this experiment is to show that Outlier Pursuit can be used to identify anomalies within the dataset. We use the data from [29], and construct the observation matrix M as containing the first 220 samples of digit "1" and the last 11 samples of "7". The learning objective is to correctly identify all the "7's". Note that throughout the experiment, label information is unavailable to the algorithm, i.e., there is no training stage. Since the columns of digit "1" are not exactly low rank, an exact decomposition

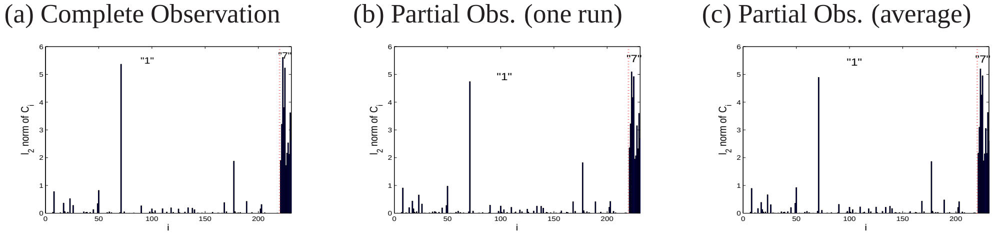

(a) 30% entries observed

(b) 80% entries observed

(c) Success rate *vs* Observe ratio

Figure 2: Partial Observation.

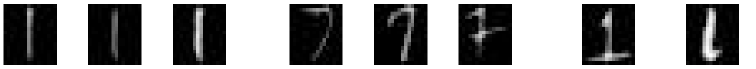
is not possible. Hence, we use the ℓ_2 norm of each column in the resulting C matrix to identify the outliers: a larger ℓ_2 norm means that the sample is more likely to be an outlier — essentially, we apply thresholding after C is obtained. Figure 3(a) shows the ℓ_2 norm of each column of the resulting C matrix. We see that all "7's" are indeed identified. However, two "1" samples (columns 71 and 137) are also identified as outliers, due to the fact that these two samples are written in a way that is different from the rest "1's" as showed in Figure 4. Under the same setup, we also simulate the case where only 80% of entries are observed. As Figure 3 (b) and (c) show, similar results as that of the complete observation case are obtained, i.e., all true "7's" and also "1's" No 71, No 177 are identified.


(a) Complete Observation

(b) Partial Obs. (one run)

(c) Partial Obs. (average)

Figure 3: Outlyingness: ℓ_2 norm of C_i.


"1"

"7"

No 71

No 177

Figure 4: Typical "1", "7" and abnormal "1".

6 Conclusion and Future Direction

This paper considers robust PCA from a matrix decomposition approach, and develops the algorithm Outlier Pursuit. Under some mild conditions, we show that Outlier Pursuit can exactly recover the column support, and exactly identify outliers. This result is new, differing both from results in Robust PCA, and also from results using nuclear-norm approaches for matrix completion and matrix reconstruction. One central innovation we introduce is the use of an oracle problem. Whenever the recovery concept (in this case, column space) does not uniquely correspond to a single matrix (we believe many, if not most cases of interest, will fall under this description), the use of such a tool will be quite useful. Immediate goals for future work include considering specific applications, in particular, robust collaborative filtering (here, the goal is to decompose a partially observed column-corrupted matrix) and also obtaining tight bounds for outlier identification in the noisy case.

Acknowledgements H. Xu would like to acknowledge support from DTRA grant HDTRA1-08-0029. C. Caramanis would like to acknowledge support from NSF grants EFRI-0735905, CNS-0721532, CNS- 0831580, and DTRA grant HDTRA1-08-0029. S. Sanghavi would like to acknowledge support from the NSF CAREER program, Grant 0954059.

References

[1] I. T. Jolliffe. *Principal Component Analysis*. Springer Series in Statistics, Berlin: Springer, 1986.

[2] P. J. Huber. *Robust Statistics*. John Wiley & Sons, New York, 1981.

[3] L. Xu and A. L. Yuille. Robust principal component analysis by self-organizing rules based on statistical physics approach. *IEEE Tran. on Neural Networks*, 6(1):131–143, 1995.

[4] E. Candès, X. Li, Y. Ma, and J. Wright. Robust pricinpal component analysis? ArXiv:0912.3599, 2009.

[5] S. J. Devlin, R. Gnanadesikan, and J. R. Kettenring. Robust estimation of dispersion matrices and principal components. *Journal of the American Statistical Association*, 76(374):354–362, 1981.

[6] T. N. Yang and S. D. Wang. Robust algorithms for principal component analysis. *Pattern Recognition Letters*, 20(9):927–933, 1999.

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

[8] F. De la Torre and M. J. Black. Robust principal component analysis for computer vision. In *ICCV'01*, pages 362–369, 2001.

[9] F. De la Torre and M. J. Black. A framework for robust subspace learning. *International Journal of Computer Vision*, 54(1/2/3):117–142, 2003.

[10] C. Croux, P. Filzmoser, and M. Oliveira. Algorithms for Projection$-$Pursuit robust principal component analysis. *Chemometrics and Intelligent Laboratory Systems*, 87(2):218–225, 2007.

[11] S. C. Brubaker. Robust PCA and clustering on noisy mixtures. In *SODA'09*, pages 1078–1087, 2009.

[12] H. Xu, C. Caramanis, and S. Mannor. Principal component analysis with contaminated data: The high dimensional case. In *COLT'10*, pages 490–502, 2010.

[13] E. J. Candès, J. Romberg, and T. Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Tran. on Information Theory*, 52(2):489–509, 2006.

[14] B. Recht, M. Fazel, and P. Parrilo. Guaranteed minimum rank solutions to linear matrix equations via nuclear norm minimization. To appear in *SIAM Review*, 2010.

[15] V. Chandrasekaran, S. Sanghavi, P. Parrilo, and A. Willsky. Rank-sparsity incoherence for matrix decomposition. ArXiv:0906.2220, 2009.

[16] E. Candès and B. Recht. Exact matrix completion via convex optimization. *Foundations of Computational Mathematics*, 9:717–772, 2009.

[17] E. Candès and T. Tao. The power of convex relaxation: Near-optimal matrix completion. *IEEE Tran. on Information Theory*, 56(2053-2080), 2010.

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

[19] R. Maronna. Robust M-estimators of multivariate location and scatter. *The Annals of Statistics*, 4:51–67, 1976.

[20] V. Barnett. The ordering of multivariate data. *Journal of Royal Statistics Society, A*, 138:318–344, 1976.

[21] D. Titterington. Estimation of correlation coefficients by ellipsoidal trimming. *Applied Statistics*, 27:227–234, 1978.

[22] V. Barnett and T. Lewis. *Outliers in Statistical Data*. Wiley, New York, 1978.

[23] A. Dempster and M. Gasko-Green. New tools for residual analysis. *The Annals of Statistics*, 9(5):945–959, 1981.

[24] S. J. Devlin, R. Gnanadesikan, and J. R. Kettenring. Robust estimation and outlier detection with correlation coefficients. *Biometrika*, 62:531–545, 1975.

[25] G. Li and Z. Chen. Projection-pursuit approach to robust dispersion matrices and principal components: Primary theory and monte carlo. *Journal of the American Statistical Association*, 80(391):759–766, 1985.

[26] H. Xu, C. Caramanis, and S. Sanghavi. Robust PCA via outlier pursuit. http://arxiv.org/abs/1010.4237, 2010.

[27] Y. Nesterov. A method of solving a convex programming problem with convergence rate $o(1/k^2)$. *Soviet Mathematics Doklady*, 27(372-376), 1983.

[28] J-F. Cai, E. Candès, and Z. Shen. A singular value thresholding algorithm for matrix completion. *SIAM Journal on Optimization*, 20:1956–1982, 2008.

[29] C. Rasmussen and C. Williams. *Gaussian Processes for Machine Learning*. the MIT Press, 2006.