Low Rank Approximation in the Presence of Outliers

Aditya Bhaskara
School of Computing, University of Utah, Salt Lake City, UT, USA
http://www.cs.utah.edu/~bhaskara/
bhaskara@cs.utah.edu

Srivatsan Kumar
School of Computing, University of Utah, Salt Lake City, UT, USA
seezha@gmail.com

Abstract

We consider the problem of principal component analysis (PCA) in the presence of outliers. Given a matrix A ($d \times n$) and parameters k, m, the goal is to remove a set of at most m columns of A (outliers), so as to minimize the rank-k approximation error of the remaining matrix (inliers). While much of the work on this problem has focused on recovery of the rank-k subspace under assumptions on the inliers and outliers, we focus on the approximation problem. Our main result shows that sampling-based methods developed in the outlier-free case give non-trivial guarantees even in the presence of outliers. Using this insight, we develop a simple algorithm that has bi-criteria guarantees. Further, unlike similar formulations for clustering, we show that bi-criteria guarantees are unavoidable for the problem, under appropriate complexity assumptions.

2012 ACM Subject Classification Theory of computation → Approximation algorithms analysis

Keywords and phrases Low rank approximation, PCA, Robustness to outliers

Digital Object Identifier 10.4230/LIPIcs.APPROX-RANDOM.2018.4

Funding The first author is partially supported by a Google Faculty Award.

1 Introduction

Low rank approximation is one of the most fundamental and well-studied questions in matrix analysis. It is widely used for dimension reduction, sketching, denoising, and more broadly to find "structure" in large matrices. Usually referred to as principal component analysis (PCA) or singular value decomposition (SVD), low rank approximations are a staple tool in the analysis of large matrix data.

We consider the problem of low rank approximation in the presence of *outlier* columns. Studying the effect of adversarial outliers on known algorithms as well as the problem complexity has become a significant theme in recent research. It is motivated by the practical importance of dealing with noise in data (both intrinsic as well as adversarial). While principal components are often robust to small corruptions of the matrix (which is why they are useful in denoising, e.g. [28]), this typically requires the noise to have a small spectral norm (see [31]). This is an unrealistic assumption in many settings, especially when entire columns can be (arbitrarily and possibly adversarially) noisy.

Dealing with noise has also been well studied in the statistics community (the books of [22, 23] present many of the classic results in the area). Robust estimators for computing parameters of distributions such as the mean, variance, etc. have been extensively studied. More recently, questions of this nature have received a lot of attention in theoretical computer

science, motivated by connections to learning distributions. The works of [12, 26], as well as subsequent works (see [30] and references therein) have obtained novel guarantees on recovery under noise, using semidefinite programming and other techniques.

Meanwhile, clustering problems have long been studied in the presence of outliers, from the point of view of approximation algorithms. Starting with the work of [6], and subsequent improvements (see [8, 18, 25]), we now have a good understanding of the approximability of clustering with outliers. Many of these works use linear programming relaxations that help identify the outliers. One of our motivations is to obtain a similar understanding for PCA.

PCA with outliers. We now define the formulation we study in this paper. It is motivated by the work on clustering algorithms discussed above. Informally, given a matrix A ($d \times n$), we wish to find the best rank k approximation to A, after throwing away m columns of our choice. This can be formally stated as the following optimization problem:

$$\text{minimize } \|A - L - N\|_F^2, \quad \text{subject to } \mathrm{rk}(L) \leq k, \text{ and } N \text{ having at most } m \text{ non-zero columns.}$$

This formulation is has also been studied in the signal processing literature [34, 5], where it is referred to as Robust PCA. (A different notion was given that name in [4].) The formulation also makes sense with different matrix norms, though in this paper, we focus mostly on the Frobenius norm. Our guarantees also work for entry-wise ℓ_p norm error, as we will see.

1.1 Background and related work

The statistics literature on robust estimation of parameters is extensive, and we refer to the book of [23] for a review. We now discuss some of the works most relevant to our results.

As mentioned above, the PCA with outliers problem has been well studied in the signal processing community. The work of [5] shows that by formulating the problem as an optimization problem with appropriate regularization terms, we can efficiently recover the optimal rank k subspace, under certain conditions on the input. Informally, these conditions say that the inliers must be "well spread", which in turn implie the uniqueness of the target subspace (this is similar in spirit to works on matrix completion). The worst-case problem was studied in [32], under a "coverage" variant of the objective (in which the goal is to find a rank k subspace that covers as large a fraction of the inlier mass as possible). The results here do not imply a multiplicative approximation to our formulation, and are thus incomparable.

We also note that while the problem definition naturally suggests $m \ll n$, the problem makes sense when we have a large fraction (approaching 1) of outliers. This version was studied in the work of [20]. They formulate the problem as follows: given a set of points in $\mathbb{R}^n$ with the condition that an α fraction lie in a d-dimensional subspace, the goal is to find the subspace. Informally, under the condition that the "outliers" (the points not in the subspace) are in general position, they develop a simple random sampling algorithm, for the case $\alpha > d/n$. They also show hardness results for this problem. Indeed, we note that the hardness results we present in Section 5 are consequences of their approach.[1]

The main issue with $m \approx n$ is that in the absence of any strong assumptions (along the lines of the above works), the outliers could themselves have an approximate rank-k structure, thus making the problem ill-posed. For problems such as mean estimation, [7] recently showed that in such cases, something very elegant is possible: we can come up with

[1] We thank Amit Deshpande and Ravishankar Krishnaswamy for suggesting this.

a small set of *candidate* solutions, one of which will be a good solution to the inliers. This framework was originally introduced in [1], and together with some very interesting new ideas, it has been used to obtain results for classic problems such as learning mixtures of Gaussians under separation [13, 24, 21].

Finally, as we mentioned above, outlier-robust approximation algorithms are quite well studied for different variants of clustering (see [6, 8, 18, 25] and references therein). For many variants, while the initial algorithms were bi-criteria approximations similar to ours, it turns out that one can actually obtain constant factor approximation algorithms without violating either the bound on the number of centers, or the number of outliers. This is in contrast to what turns out to be possible for PCA, as we will see.

Robust algorithmic methods. The high level goal in the works above (as well as ours) is the development of algorithmic techniques that work in the presence of outliers. To this end, linear and semidefinite relaxations have been powerful in identifying the inliers/outliers, and have emerged as a powerful technique. This is seen in the works of [12, 7], the works on clustering mentioned above, and also the extensive literature on semi-random models (see [14, 27] and references therein). Recently, [30] studied the abstract question of when an estimator can be computed robustly, and defined a condition they call *resilience*. This is a property of the inliers that allows estimation (in principle) in the presence of a small fraction of arbitrary outliers. In this context, our work shows that random sampling based algorithms could be powerful in finding structure in the presence of noise, albeit with weaker (bi-criteria) guarantees. This is the idea behind heuristics such as RANSAC, which we will now discuss.

Heuristics. There have been several heuristics developed specifically for the robust PCA problem, as well as more generally for estimation on noisy data. One example is the class of "Lloyd-style" heuristics, where the idea is to fit a rank-k subspace to the full data set, remove a small number of points that are "far away" from the space, and recurse (see [18, 33]). Another heuristic the famous RANSAC (random sampling and consensus) paradigm [15]. The idea here is that if we randomly sample a subset of the points and fit a k-subspace, then different samples yield spaces that align "along the inliers" but have differing components along the outliers. Assuming the outliers do not have "structure", we can expect that an averaging (consensus) step helps zero in on the inliers. One issue with the above is that the outliers could have an approximate rank k structure. Intuitively, this is one of the reasons for which we only obtain bi-criteria guarantees.

1.2 Our results

In the remainder of the paper, we will write $A = B + N$, where B consists of the inliers (and zeros in the outlier columns), and N contains the outliers (and zeros in the inlier columns). Thus, in this notation, the problem can be reformulated as that of finding such a decomposition $B + N$ where N has at most m non-zero columns, with the goal of minimizing $\|B - B_k\|_F^2$, where B_k is the best rank-k approximation of B. Also, we will throughout think of ϵ, δ as parameters in the range $(0, 1]$.

We present two (incomparable) algorithmic results. The first is a simple algorithm based on iteratively computing a subspace that captures more and more "mass" of the matrix, while throwing away a small number of outliers. The second algorithm, which is our main contribution, is inspired by *adaptive sampling*, an idea that has been successful in obtaining

coresets and bi-criteria approximations for problems such as clustering and PCA [11, 9]. To describe the first result, we define the following "rank-k condition number":

$$\Lambda_k := \frac{\|A\|_F^2}{\|B - B_k\|_F^2}.$$

▶ **Theorem 1.** *There is an efficient algorithm that takes as input a matrix A as above, parameters k, m, ϵ, and outputs a decomposition $A = B' + N'$ along with a subspace V, such that the following properties hold: (a) N' has at most $m \log(\Lambda_k/\epsilon)$ columns, (b) the space V satisfies the property*

$$\|\Pi_V^{\perp} B'\|_F^2 \leq (1+\epsilon)\|B - B_k\|_F^2,$$

and (c) the dimension of V is at most $k \log(\Lambda_k/\epsilon)$.

The algorithm above violates the bounds on the number of outliers and the dimension of the space by a factor $\log(\Lambda_k/\epsilon)$. Thus it is interesting in the regimes where m is small compared to the number of columns of A, and the quantity Λ_k is "not too large". For instance, in some practical settings, one might be interested in capturing say 99% of the mass of a matrix using a small subspace, while excluding a small number of outliers. In such a case, the Λ_k is a constant, and all the additional factors are small.

Our second (and main) result has a much better dependence on the parameters. It has as an additional input a parameter δ, which controls the number of outliers the algorithm outputs.

▶ **Theorem 2.** *There is an efficient algorithm that takes as input a matrix A as above, parameters k, m, ϵ, δ, and outputs a decomposition $A = B' + N'$ along with a subspace V, such that the following properties hold: (a) N' has at most $(1+\delta)m$ columns, (b) the space V satisfies the property $\|\Pi_V^{\perp} B'\|_F^2 \leq (1+\epsilon)\|B - B_k\|_F^2$, and (c) V is the span of a subset of the columns of A, of size at most*

$$O\left(\frac{k}{\epsilon^6}\left(\log(n/m) + \frac{2}{\delta}\right)\right).$$

Thus, the theorem in fact gives a "column based" approximation to the space V. Also, if we think of ϵ as a constant, the algorithm outputs $O(k \log(n/m) + k/\delta)$ columns. Note that in many cases of interest, we may have m being a small constant factor of n. In such cases, the algorithm roughly outputs only a $O(k/\delta)$ dimensional space, while obtaining a $(1+\epsilon)$ approximation to the objective and violating the number of constraints by a factor $(1+\delta)$.

Dependence on ϵ. The drawback in Theorem 2 is the dependence on ϵ. Indeed, the first algorithm, while lossy in many other aspects, has a really good dependence on ϵ (which is what makes the algorithms incomparable). However, we note that even in the noise-free case, obtaining a column-based $(1+\epsilon)$ approximation to the best rank-k approximation requires k/ϵ columns (see [19]). Thus we cannot hope to get rid of $1/\epsilon$ entirely.

Technique and extensions. It turns out that the key to Theorem 2 is a modification of a remarkable lemma from [9], on finding low-rank approximations under entry-wise ℓ_p error by using iterative *uniform* sampling. While the modification (see Lemma 6 and the notes following it) is needed for our main result, it turns out that if we use the original lemma of [9] in our framework, we obtain the following as a corollary. Note that the approximation ratio is now $O(k)$ as opposed to $(1+\epsilon)$ for the case of Frobenius norm. (A dependence on k turns out to be unavoidable for column-based approximations for ℓ_p error even without noise [9].)

▶ **Theorem 3.** *There is an efficient algorithm that takes as input a matrix A as above, parameters k, m, δ, and outputs a decomposition $A = B' + N'$ along with a subspace V, such that the following properties hold: (a) N' has at most $(1+\delta)m$ columns, (b) the space V is spanned by $O(k(\log(n/m) + 1/\delta))$ columns of A, and (c) the error satisfies*

$$err(p, B', V) \leq 100(k+1) \cdot \min_{X \in \mathbb{R}^{d \times k}, Y \in \mathbb{R}^{k \times n}} \|B - XY\|_p^p,$$

where $err(p, B', V)$ denotes the minimum ℓ_p^p reconstruction error of the columns of B' using V.

Limits of approximation. It is natural to ask if there needs to be a trade-off between the dimension of the output space and the slack parameter δ. Furthermore, we can even ask if we can avoid having a slack altogether.

By a reduction along the lines of the result of [20], the following result is quite easy to show.

▶ **Theorem 4** (Informal version of Theorem 12)**.** *Under the small set expansion conjecture with suitable parameters, for any constant $C > 0$, there is no polynomial time algorithm that can obtain a multiplicative factor approximation to the objective, while returning a Ck dimensional subspace, and excluding at most $(1+\delta)m$ outliers, for small enough constant δ.*

This rules out the possibility of finding a Ck-dimensional subspace for arbitrarily small δ. A very similar reduction, but from the smallest r-edge subgraph problem (see Section 5.2) implies that if we wish to have $\delta = 0$, then the dimension of the subspace output must be at least $k \cdot n^{\Omega(1)}$. See Corollary 13 for the formal statement. We remark once more that these hardness results indicate that "pure" approximations (as can be obtained for clustering) are impossible for PCA.

Open problems, directions. While bi-criteria guarantees are unavoidable in the worst case, it is interesting to see if simple iterative algorithms like the ones we proposed can be shown to *recover* the k-PCA subspace of the inliers under appropriate assumptions. A starting point would be assumptions similar to the ones of Donoho et al. Next, the dependence on δ, ϵ in our sampling based algorithm are possibly sub-optimal. It would be interesting to see if more sophisticated algorithms can give better guarantees. More broadly, it would be interesting to show more guarantees for heuristics such as RANSAC and algorithms inspired by them for other problems involving outliers.

2 Notation and preliminaries

We start with some basic matrix notation we use. Let A be a $d \times n$ matrix. Throughout the paper, we write A_k to refer to the best rank-k approximation of A (thus it is also a $d \times n$ matrix). For a subset T of the column indices ($T \subseteq [n]$), we denote by $A_{(T)}$ the $d \times |T|$ sub-matrix of A formed by the columns indexed by T.

Next, given an integer k, we denote by $err_k(A)$ the error in the best rank-k approximation of A. Specifically, $err_k(A) = \|A - A_k\|_F^2$. Also, for a set of vectors W, their linear span is denoted span(W). Finally, the projection matrix orthogonal to the space orthogonal to span(W) will be denoted by $\Pi_W^{\perp}$. So also, for a *subspace* W, we abuse notation slightly and denote by $\Pi_W^{\perp}$ the projection matrix to the space orthogonal to W.

Algorithm 1 Iterative SVD.

Input: Matrix $A \in \mathbb{R}^{d \times n}$, guess ξ for the optimum error, parameter m (bound on # outliers), and accuracy parameter ϵ.

Output: A subspace V, and a set S of inliers.

Initialize $V_0 = \emptyset$, $S_0 = \text{cols}(A)$, and $j = 0$.
while total squared projection of S_j onto the space orthogonal to V_j is $\geq (1+\epsilon)\xi$ **do**
  Let $u_1, u_2, \ldots, u_N$ be the projection of the columns in S_j orthogonal to V_j.
  Define $\mu_j := \sum_i \|u_i\|^2$.
  Let T_j be the m largest (by length) vectors among $\{u_i\}_{i=1}^N$.
  if $(\sum_{u \in T_j} \|u\|^2 \geq \frac{1}{2}(\mu_j - \xi)$) **then**
    $S_{j+1} := S_j \setminus T_j$, and $V_{j+1} = V_j$.
  else
    Let V_{j+1} be the rank-$((j+1)k)$ SVD for S_j.
    Set $S_{j+1} = S_j$.
  end if
  $j \leftarrow j + 1$
end while
return V_j, S_j

Guessing the optimum. In all our algorithms, we assume that we have a guess for the optimum error (error in the low-rank approximation of B), up to a multiplicative factor of $(1+\epsilon)$. A fairly straightforward argument shows that we can always come up with a polynomial number (in the input complexity) of guesses, one of which is accurate. This is shown in Appendix A.

3 Iterative SVD

We now present our first algorithm. It involves repeatedly computing the best low rank approximation, while potentially throwing away some points as outliers. This will establish Theorem 1. Let us start with an informal description of the algorithm.

Algorithm outline. At each step j, we have a subset S_j of the initial column vectors, and a subspace V_j. Let $N = |S_j|$ and let $u_1, u_2, \ldots, u_N$ denote the projections of the columns in S_j, orthogonal to the space V_j. The aim is to either (a) find a new subspace V_{j+1} that captures a significantly larger fraction of the total mass than S_j, or (b) remove a set of at most m columns from S_j and mark them as outliers. In either case, we show that the total "uncaptured" mass remaining drops by a constant factor. This allows us to bound the number of iterations, thus giving the guarantees of Theorem 1.

The algorithm is formally stated below (Algorithm 1). As discussed in Section A, it assumes that we have a guess ξ for the optimum error.

▶ **Lemma 5.** *In every iteration of the algorithm, the total mass of the inliers reduces significantly. More precisely, we have* $\mu_{j+1} \leq (\mu_j + \xi)/2$.

Proof. We consider both the cases in the algorithm.

Case 1. $\sum_{u \in T_j} \|u\|^2 \geq \frac{1}{2}(\mu_j - \xi)$.

In this case, we remove T_j from the set of inliers, and thus

$$\mu_{j+1} = \mu_j - \sum_{u \in T_j} \|u\|^2 \leq \frac{\mu_j + \xi}{2}.$$

Case 2. $\sum_{u \in T_j} \|u\|^2 < \frac{1}{2}(\mu_j - \xi)$.

Let us denote the set of inlier columns by $\mathcal{I}$. Let us argue about the error in the best rank-$(j+1)k$ approximation of S_j. To do so, we consider the space $V' = V_j + V^*$, where V^* is the rank-k SVD space of $\mathcal{I}$ (this is the optimal subspace that we are after). Now, let us consider the projection of the columns of S_j orthogonal to V'. For the columns $S_j \cap \mathcal{I}$, the total error has to be $\leq \xi$, because by assumption, projecting $\mathcal{I}$ orthogonal to V^* has this error. Next, the projection of $S_j \setminus \mathcal{I}$ orthogonal to V' must be smaller than (or equal to) the projections orthogonal to V_j. As $|S_j \setminus \mathcal{I}| \leq m$ (there are at most m outliers), their projections orthogonal to V_j can be bounded by $\sum_{u \in T_j} \|u\|^2$ (as T_j contained the m largest vectors by length). This allows us to bound the total error by

$$\xi + \frac{\mu_j - \xi}{2} = \frac{\mu_j + \xi}{2}.$$

The best $(j+1)k$ dimensional subspace will result in an error only better than the above, and thus the lemma follows.[2] ◀

The lemma above immediately implies that (assuming that the guess of ξ is an upper bound on the optimum), the algorithm runs for at most $\log(\|A\|_F^2/\epsilon\xi)$ iterations. This is because the gap between μ_j and ξ drops by a factor at least 2 in each iteration. Thus, by searching over all the possible ξ and by the comment below, Theorem 1 follows.

Validating the guess of ξ. We note that the above algorithm works whenever ξ is an upper bound on the optimum. If it is lower than the optimum, then in one of the iterations, we may not see a drop in μ_{j+1} as guaranteed by Lemma 5. Thus, we can test for this as the algorithm proceeds and output FAIL if we do not see a drop.

4 Iterative uniform sampling

We next present our main algorithm for the PCA with outliers problem. We will start with a sampling lemma that is at the heart of the algorithm. This lemma applies to the case when there are no outlier columns. As mentioned earlier, the lemma is a variant of a lemma from [9], which applies to ℓ_p norms and has additional factors of k. In Section 4.2, we show how the lemma can be used even in the presence of outliers, thus establishing Theorem 2.

4.1 Sampling without outliers

The first lemma is simply about low rank approximation via columns (without any outliers).

[2] We note that a more "natural" algorithm is to add the top-k SVD space of the u_i vectors to the current space in step 9. This is closer to adaptive sampling paradigm (see [11]), but it runs into the issue that it is not clear the projection of the u_i corresponding to inliers reduces to ξ. This stems from the fact that for two spaces V, W, $\Pi_V u$ could be smaller in length than $\Pi_V(\Pi_W u)$.

▶ **Lemma 6.** *Let $A \in \mathbb{R}^{d\times n}$, and let $\epsilon \in (0,1]$ be any parameter. Let S be a uniformly random subset of $[n]$ of size s, where $s \geq 4k/\epsilon^2$. Then, with probability at least $\epsilon^2/8$, there exist a set of $\epsilon^2 n/8$ columns of A, whose projection orthogonal to the column span of $A_{(S)}$ is upper bounded by $(1+\epsilon)\|A - A_k\|_F^2/n$.*

Note. The lemma is quite surprising: for say $\epsilon = 1$, it says that a *uniformly random* subset of $4k$ columns covers a constant fraction of the columns up to a constant times the k-SVD error. Such a guarantee is not even clear for norm-based sampling (see [16]). So, while uniform sampling does not necessarily give a low *total error* in expectation, it ends up giving small error for a constant fraction of columns.

We use the same rough outline as the proof of [9]. There are, however, two main differences. First, as we need a column-based $(1+\epsilon)$ guarantee on rank-k approximation, we appeal to the results of [19], instead of a weaker $O(k)$ bound used in [9] for the ℓ_p norm. Second (and more significant), their proof uses a simple union bound over failure probabilities. This does not suffice for a $(1+\epsilon)$ approximation, as two of the events are low probability (roughly ϵ). We observe that these events are effectively independent, and so we obtain a constant probability of success. .

Proof. Let us denote $s = 4k/\epsilon^2$. Let T be a random subset of $[n]$ of size $s+1$. We think of sampling S as first sampling T and then removing a random element $u \in T$. For convenience, let us write

$$\theta = \frac{\|A - A_k\|_F^2}{n}.$$

First, let us call a subset T *good* if $A_{(T)}$ has a small error rank-k approximation. Concretely, T is said to be good if $err_k(A_{(T)}) \leq (1+\epsilon)\cdot|T|\theta$. Now, if Π_k is the projection matrix onto the space orthogonal to the best rank k approximation for the full matrix A, we have $\mathbb{E}[\sum_{u\in T}\|\Pi_k u\|^2] = |T|\theta$ (where the expectation is over the choice of T). Thus, by Markov's inequality, we have

$$\Pr\left[\sum_{u\in T}\|\Pi_k u\|^2 > (1+\epsilon)|T|\theta\right] \leq \frac{1}{1+\epsilon} \leq 1 - \frac{\epsilon}{2}. \tag{1}$$

Thus, as the best rank-k approximation for $A_{(T)}$ can only have a smaller error, we have that T is good with probability at least $\epsilon/2$. Next, we show the following claim.

▶ **Claim 1.** *For any T of size $\geq 4k/\epsilon^2$, if we form S by randomly removing a $u \in T$, then with probability at least $\epsilon/2$ (over the choice of u), we have*

$$\|\Pi_S^{\perp} u\|^2 \leq (1+\epsilon)\cdot\frac{err_k(A_{(T)})}{|T|}. \tag{2}$$

To show this claim, we first appeal to the existence of good column-based low-rank approximations. Specifically, Guruswami and Sinop showed the following.

▶ **Theorem 7** (Guruswami, Sinop [19])**.** *For any matrix B and parameter k, there exist a subset W of at most k/ϵ columns of B with the property that*

$$\|\Pi_W^{\perp} B\|_F^2 \leq (1+\epsilon)\cdot err_k(B).$$

We apply the result to the matrix $A_{(T)}$, and let W denote the set of columns guaranteed by the theorem. Now, the probability that a random column u of $A_{(T)}$ belongs to W is at most $|W|/|T|$, which is at most $\epsilon/2$, by our choice of s. Thus, the probability that $S \supset W$ is at least $1 - \epsilon/4$.

Next, we also have that for a random column u of $A_{(T)}$, the expected value

$$\mathbb{E}[\|\Pi_W^{\perp} u\|^2] = \frac{\|\Pi_W^{\perp} A_{(T)}\|_F^2}{|T|} \leq (1+\epsilon)\ \frac{err_k(A_{(T)})}{|T|}.$$

Once again, by Markov's inequality, the probability that $\|\Pi_W^{\perp} u\|^2$ is bounded by $(1+\epsilon)$ times the RHS is at least $\epsilon/2$. Thus by a union bound, with probability at least $\epsilon/4$, we have this condition, as well as the event $S \supset W$. In this case, we clearly have $\|\Pi_S^{\perp} u\| \leq \|\Pi_W^{\perp} u\|$, and this completes the proof of Claim 1.

Now, consider the bipartite graph in which the left side consists of all $(s+1)$-tuples of $[n]$ and the right side consists of all s-tuples of $[n]$. We place an edge between T and S if (a) T is good, and (b) $u = T \setminus S$ satisfies Eq. (2). The claim above, together with Eq. (1) (which lower bounds the probability of T being good), imply that the total number of edges is at least

$$\frac{\epsilon}{2}\binom{n}{s+1}\frac{\epsilon}{4}(s+1) = \frac{\epsilon^2}{8}(n-s)\binom{n}{s}.$$

Note that $n - s$ is the maximum number of edges that could be incident to a vertex on the right. Thus, we conclude that at least an $\epsilon^2/8$ fraction of the vertices on the right have degree at least $\epsilon^2(n-s)/8$. Since an edge implies that (a) T is good and (b) Eq. (2) holds, the conclusion of the lemma follows. ◀

4.2 Incorporating outliers

Next, suppose the set of columns contains (at most) m outliers. We then consider Algorithm 2. The analysis will use the value of the average error per column in the optimum solution, namely $\theta := \|B - B_k\|_F^2/(n-m)$.

Define $n' = n - m$. Let $z_1, z_2, \ldots, z_{n'}$ be the rank k approximation errors of the columns B_i of B. (So $\sum_i z_i = n'\theta$.) We will assume (without loss of generality) that z_i are in increasing order.

Proof outline. Consider the first iteration of the algorithm. We claim that $\|\Pi_{R^*} A^*\|_F^2 \leq (1+\epsilon)(z_1 + z_2 + \cdots + z_{\epsilon n'})/(\epsilon n')$ with probability at least $1 - \frac{1}{n^4}$. This is because with high probability, one of the candidates for R would have chosen $4k/\epsilon^2$ columns *among* $B_1, \ldots, B_{\epsilon n'}$, and then we can apply Lemma 6. Thus, the error for the first $n'\epsilon^3$ columns covered is at most the average error for the first $\epsilon n'$ columns. Subsequently, we will be able to bound the error in each step in terms of the smallest ϵ fraction of the z_i's that remain.

▶ **Lemma 8.** *Consider the procedure* SELECT$(A', m, k, \epsilon, \delta)$*, and suppose that* $N := \#cols(A')$ *satisfies* $N \geq \max\{n_0, (1+\delta)m\}$*, where* n_0 *is defined in step 3 of the algorithm. Let* $B' = A' \cap B$*, i.e., the set of inliers in* A'*, and let* $y_1, y_2, \ldots, y_{|B'|}$ *denote the values* z_i *for* $i \in B'$*. Assume w.l.o.g. that* y_i *are in increasing order. Then with probability* $\geq 1 - \frac{1}{n^4}$*, the sets* R^*, A^* *chosen by the procedure satisfy*

$$\text{for all } u \in A^*,\ \|\Pi_{R^*}^{\perp} u\|_F^2 \leq \frac{\sum_{i=1}^{\epsilon|B'|} y_i}{\epsilon|B'|}. \tag{3}$$

Algorithm 2 Iterative sampling with outliers.

Input: Matrix $A \in \mathbb{R}^{d\times n}$, parameters m, k, δ, ϵ, guess θ for optimum error per column.
Output: A set of outliers $\mathcal{O}$ and a set of columns V of A.

1: **procedure** SELECT($A, m, k, \epsilon, \delta$)
2: Initialize $\mathcal{O} = V = \emptyset$. Define $N = \#\text{cols}(A)$.
3: Define $n_0 = \frac{\alpha}{\alpha-1} \cdot \frac{8k}{\epsilon^3}$, where $\alpha = N/m$.
4: **if** $N < n_0$ **then return** add all the columns of A to V and return $(\mathcal{O}, V)$
5: **else if** $N \leq (1+\delta)m$ **then**
6: add all the columns of A to $\mathcal{O}$ and return $(\mathcal{O}, V)$
7: **else**
8: **for** $16 \log n/\epsilon^2$ iterations **do**
9: Let $R \leftarrow$ a uniformly random sample of n_0 columns of A
10: Let $\hat{A}$ be the set of $\epsilon^3(N-m)$ columns of A that have the least projection to $\Pi_R^{\perp}$
11: Let $X = \|\Pi_R^{\perp}\hat{A}\|_F^2$
12: If X is smaller than the least such quantity so far, set $A^* = \hat{A}$, $R^* = R$
13: **end for**
14: Mark the columns A^* as *covered*
15: Let $(\mathcal{O}', V')$ be the output of the recursive call SELECT($A \setminus A^*, m, k, \epsilon, \delta, \theta$)
16: **return** $(\mathcal{O}', R^* \cup V')$
17: **end if**
18: **end procedure**

Proof. For each iteration of the loop (8-13) of the algorithm, we show that the probability of the chosen $R, \hat{A}$ satisfying the bound in Eq. (3) is at least $\epsilon^2/2$. The conclusion of the lemma then follows immediately.

First, note that for $\alpha = N/m$, we have $|B'|/|A'| \geq (\alpha-1)/\alpha$. Let us also denote by Q the set of $\epsilon|B'|$ columns of B' that have the smallest z_i values. Now, the expected size of $R \cap Q$ is

$$\frac{|R|}{|A'|} \cdot \epsilon|B'| \geq \frac{8k}{\epsilon^2}.$$

Thus the probability that this is $\geq 4k/\epsilon^2$ is at least $1/2$. (Formally, this follows from Hoeffding's inequality, which also applies to sums of random variables without replacement.)

Conditioned on $|R \cap Q| \geq 4k/\epsilon^2$, we can apply Lemma 6 to conclude that with probability $\geq \epsilon^2$, at least an ϵ^2 fraction of the columns in Q, the projection orthogonal to span(R) is bounded by $\sum_{i\in Q} z_i/|Q|$. Note that by definition, this is precisely the RHS of Eq (3). Thus the probability that the chosen $R, \hat{A}$ satisfy (3) is at least $\epsilon^2/2$, and this completes the proof of the lemma. ◀

▶ **Lemma 9.** *When the algorithm terminates, the total error incurred is bounded by* $\frac{1+\epsilon}{1-\epsilon}n'\theta$, *with high probability. Further, the depth of the recursion is upper bounded by* $\frac{\log(n/(\delta m))}{\epsilon^3}$. *The total number of columns chosen is at most*

$$\frac{16k}{\epsilon^6}\left(\log(n/m) + \frac{2}{\delta}\right).$$

Proof. Using Lemma 8, we can analyze the overall error as follows. Since the success probability in each iteration is $1 - \frac{1}{n^4}$ we assume henceforth that the conclusion of Lemma 8

holds for all the recursive calls. Let us divide the indices $[n']$ (from left to right) into groups of size $\epsilon n', \epsilon(1-\epsilon)n', \epsilon(1-\epsilon)^2 n', \ldots$. Let us call the groups $G_1, G_2, \ldots$, and let E_i denote $\sum_{j \in G_i} z_j$. Now, by the lemma, until the algorithm marks $\epsilon n'$ columns as covered, we have that the average error in the marked columns is at most $(1+\epsilon)E_2/|G_2|$, w.h.p. (Note that the set of columns marked by the algorithm can be quite different from G_1; but this will only help the argument, which only requires that an ϵ fraction of the uncovered columns has average error $\leq E_2/|G_2|$.) Likewise, until the next $\epsilon(1-\epsilon)n'$ columns are marked covered, the average error is $\leq (1+\epsilon)E_3/|G_3|$. This continues until the number of unmarked columns falls below k/ϵ^3, at which point the error will be zero, as all the columns will be picked.

Thus, we cover the first $|G_1|$ columns with average error $\leq (1+\epsilon)E_2/|G_2|$, the next $|G_2|$ columns with average error $\leq (1+\epsilon)E_3/|G_3|$, and so on. And when $|G_i|$ gets to k/ϵ^3 (or δm), the error is zero and the procedure stops. Since $|G_i|/|G_{i+1}| = 1/(1-\epsilon)$, we can bound the total error by

$$(1+\epsilon)\frac{E_2 + E_3 + \ldots}{1-\epsilon} \leq \frac{(1+\epsilon)n'\theta}{1-\epsilon}.$$

This completes the proof of the error bound.

For the depth of the recursion, note that we always mark precisely $\epsilon^3(N-m)$ columns as marked. The procedure terminates when $N-m$ is $\max\{n_0, (1+\delta)m\}$, and this gives the desired bound.

To bound the number of columns chosen, we need to analyze the sum of n_0 over the recursive calls. We note that as long as $\alpha \geq 2$, we have $n_0 \leq 16k/\epsilon^3$. By the argument above, the number of recursive steps needed to reach $\alpha = 2$ is at most $(1/\epsilon^3) \cdot \log(n/m)$. This bounds the number of columns chosen until this step by $(16k/\epsilon^6)\log(n/m)$. Next, as $(\alpha - 1)$ drops from 2^{-j} to 2^{-j-1}, we end up with $n_0 \leq 2^{j+1} \cdot 8k/\epsilon^3$ columns in each call to Select. The number of recursive calls necessary for this drop in α is at most $1/\epsilon^3$. Thus, summing over j from 0 through $\log(1/\delta)$, we have that the number of columns chosen is at most

$$\sum_{j=0}^{\log(1/\delta)} 2^{j+1}\frac{8k}{\epsilon^6} \leq \frac{32k}{\epsilon^6\delta}.$$

This completes the proof of the lemma. ◀

Theorem 2 now follows easily.

Proof of Theorem 2. The desired bound on the number of columns, as well as the bound on the approximation ratio and the number of outliers all follow from Lemma 9. ◀

4.3 Entry-wise ℓ_p error

We notice that the algorithm above can easily be adapted to the case in which we care about the entry wise ℓ_p error. Again, we have a sampling lemma in the setting without outliers. As mentioned earlier, the following lemma was shown in [9].

▶ **Lemma 10** (Lemma 6 of [9]). *Let $A \in \mathbb{R}^{d\times n}$, and let $A_{k,p}$ be a minimizer of $\|A - X\|_p$ over rank-k matrices X. Let S be a uniformly random subset of $[n]$ of size $2k$. Then, w.p. at least $1/10$, there exists a set of $n/10$ columns of A whose optimum ℓ_p^p reconstruction error using the columns of $A_{(S)}$ is at most $100(k+1)\|A - A_{k,p}\|_p^p/n$.*

We can now apply the reasoning from earlier, along with a modified algorithm, in which we treat a column as "covered" if the ℓ_p^p reconstruction error using the columns chosen is at most $100(k+1)/n$ times the guess for the optimum. One component we need here is to be able to compute the optimum ℓ_p^p reconstruction error. This can be done via a convex program for any $p \geq 1$. We refer to [9] for the details. By following the proof earlier, we obtain Theorem 3. We omit the details.

The number of columns is now $O\left(k(\log(n/m) + 1/\delta)\right)$.

5 Limits of approximation

We make some simple observations about the computational complexity of the problem of PCA with outliers. First, we consider the consequences of a reduction due to Hardt and Moitra [20] to our setting. This yields a strong impossibility result assuming the small set expansion (SSE) conjecture (see [29]). Then, we show that if we do not allow a slack in the number of outliers, then we cannot even hope to find a "reasonably small" dimensional subspace with an error within any multiplicative factor.

Notes on our reductions. Before proceeding, we note the following

1. In the reductions, the number of outliers is very close to the total number of columns. While this is intuitively not the regime of "practical interest", it is easy to see that by padding $O(n)$ copies of a vector orthogonal to all the ones produced in the reduction, we (a) obtain the setting in which the number of outliers is a small constant fraction of the total number of columns, and (b) have the same lower bounds (because a change of ± 1 to the target dimension does not matter in our proofs). Intuitively, these are cases in which one of the components is easy to find, and it becomes trickier to find the others. These are precisely the type of pathologies avoided by the "well spread" assumptions in [5, 34].
2. Next, we note that both the reductions are in the case when the optimum error is zero. Thus, our lower bounds imply that we cannot obtain *any* multiplicative approximations under the corresponding assumptions.

5.1 Reduction from SSE

Hardt and Moitra [20] give a reduction from small set expansion (SSE) to *robust subspace recovery*, a problem closely related to PCA with outliers. Let us start by recalling the SSE(δ, ϵ) problem. We are given a Δ-regular graph $G = (V, E)$ on n vertices, and the goal is to distinguish between the following two cases

YES: there exists a set S of size δn with $\Phi(S) \leq 1/2$.[3]
NO: every set of size $\leq \delta n$ has expansion $\geq 1 - \epsilon$.

We note that typically the SSE problem is stated with the YES case having $\Phi(S) \leq \epsilon$. The version above is clearly at least as hard. The "SSE conjecture" [29] states that for any $\epsilon > 0$, there is a small enough $\delta > 0$ such that SSE(δ, ϵ) is hard (for polynomial time algorithms).

Now, the reduction of [20] constructs a collection of vectors in $\mathbb{R}^{|V|}$, one for each edge. The edge $\{i, j\}$ corresponds to the vector $\mathbf{e}_i + \mathbf{e}_j$, where $\mathbf{e}_i$ denotes the unit vector of the standard basis. For an edge f, we denote this by $v(f)$. The main lemma behind their reduction is the following:

[3] As is standard, $\Phi(S)$ denotes the conductance of S, namely the quantity $\frac{E(S,\overline{S})}{\Delta \cdot \min\{|S|, n-|S|\}}$.

▶ **Lemma 11** (Hardt, Moitra [20]). *Let E' be a subset of edges, and let n' be the total number of vertices that these edges are incident to. Then we have*

$$n'/2 \leq \dim(span(\{v(f) : f \in E'\})) \leq n'.$$

The proof is simple, and proceeds as follows. First, the upper bound is trivial, as all the vectors are spanned by the vectors in the standard basis corresponding to the vertices incident to E'. The lower bound follows from arguing that for any spanning tree T', the vectors $v(f)$ corresponding to the edges of T' are linearly independent. This can be shown by way of contradiction. Suppose there exist α_f such that $\sum_{f \in T'} \alpha_f v(f) = 0$. The edges corresponding to the non-zero coefficients form a forest. Now consider any *leaf* j, i.e., a vertex that is incident to precisely one edge in the forest (which has to exist). Then $\langle \mathbf{e}_j, \sum_f \alpha_f v(f) \rangle \neq 0$, which is a contradiction.

The lemma implies the following about PCA with outliers.

▶ **Theorem 12.** *Suppose δ, ϵ are constants such that SSE(δ, ϵ) is hard. Then, even in the zero error case of PCA with outliers, there is no efficient algorithm that can find a subspace of dimension $k/6\sqrt{\epsilon}$ containing all but $(1 + \delta/4)m$ of the points. (Recall m is the prescribed number of outliers.)*

The theorem says that even if we allow a $(1 + \delta)$ fraction more outliers, we cannot have any constant factor approximation, assuming SSE. This, in a sense, is why the dimension of the space returned in Theorem 2 has to have a dependence on δ.

Proof. We consider the reduction as above, and set the upper bound on the number of outliers to

$$m := \frac{n\Delta}{2} - \frac{\delta n \Delta}{2} = \frac{n\Delta}{2}\left(1 - \frac{\delta}{2}\right).$$

In the YES case of SSE, as discussed above, the space spanned by the standard basis vectors corresponding to the non-expanding set contains all the edges incident to that set, and hence we obtain the desired bound on the number of outliers.

In the NO case, by the choice of parameters, violating the number of outliers by a factor $(1 + \delta/4)$ still means that we must have at least $\Delta \cdot n\delta/8$ *inlier* columns. Suppose there is a space of dimension $\leq c\delta n$ that contains this many columns. Then, by Lemma 11, there must be a set of $2c\delta n$ that has at least $\Delta \cdot n\delta/8$ edges. Now a uniformly random subset of δn of these vertices has (in expectation) at least $\Delta \cdot n\delta/32c^2$ edges. Now, if every set of size δn has expansion at least $1 - \epsilon$ (since we are in the NO case), then every such set contains at most $\epsilon\Delta \cdot n\delta$ edges. Thus, we have that

$$\frac{1}{32c^2} \leq \epsilon, \quad \text{which implies } c \geq \frac{1}{6\sqrt{\epsilon}}.$$

This completes the proof of the theorem. ◀

5.2 Reduction from smallest r-edge subgraph

If we do not allow *any* slack in the number of outliers, we show that the situation is rather hopeless. The smallest r-edge subgraph (SrES) [10] problem is the following: given a graph $G = (V, E)$ on n vertices, and a parameter $r \leq |E|$, the goal is to find an induced subgraph H with the smallest number of *vertices*, that has at least r edges. The problem is closely related to (and in some sense is a *dual* of) the densest k-subgraph problem (DkS) [2]. [10] obtained

the best known approximation algorithms for the SrES problem, with an approximation factor $O(n^{2-\sqrt{3}+\epsilon})$, for any $\epsilon > 0$. The complexity of the problem is also closely related to that of DkS, and indeed, the following is believed to be true (see [10, 3] and references therein):

▶ **Conjecture 1.** *The smallest r-edge subgraph problem is NP-hard to approximate to a factor n^c, for some absolute constant $c > 0$. Specifically, there exist parameters r, d such that it is NP-hard to distinguish between*

YES: there exists an induced subgraph on d vertices with r edges.
NO: the smallest induced subgraph containing r edges has at least dn^c vertices.

Now, the following is a simple corollary.

▶ **Corollary 13.** *Consider the PCA with outliers problem in which the algorithm is constrained to ignore at most m outliers (without any slack). Then, assuming Conjecture 1, there is a constant $c > 0$ such that it is NP-hard to find a kn^c dimensional subspace that results in a multiplicative approximation to the objective.*

Proof. We can use the same argument as before, and give a reduction from a gap version of SrES. If we set the number of outliers to be $|E| - r$, then in the YES case, there is a subspace of dimension d containing r edges, while in the NO case, any such subspace must have a dimension at least $dn^c/2$ (using Lemma 11). This completes the proof. ◀

References

1 Maria-Florina Balcan, Avrim Blum, and Santosh Vempala. A discriminative framework for clustering via similarity functions. In *Proceedings of the Fortieth Annual ACM Symposium on Theory of Computing*, STOC '08, pages 671–680, New York, NY, USA, 2008. ACM. doi:10.1145/1374376.1374474.

2 Aditya Bhaskara, Moses Charikar, Eden Chlamtac, Uriel Feige, and Aravindan Vijayaraghavan. Detecting high log-densities: an $n^{1/4}$ approximation for densest k-subgraph. In *Proceedings of the 42nd ACM Symposium on Theory of Computing, STOC 2010, Cambridge, Massachusetts, USA*, pages 201–210, 2010. doi:10.1145/1806689.1806718.

3 Aditya Bhaskara, Moses Charikar, Venkatesan Guruswami, Aravindan Vijayaraghavan, and Yuan Zhou. Polynomial integrality gaps for strong sdp relaxations of densest k-subgraph. In *ACM-SIAM Symposium on Discrete Algorithms, SODA 2012, Kyoto, Japan*, 2012. doi:10.1145/1806689.1806718.

4 S. Charles Brubaker. Robust PCA and clustering in noisy mixtures. In *Proceedings of the Twentieth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2009, New York, NY, USA, January 4-6, 2009*, pages 1078–1087, 2009.

5 Emmanuel J. Candès, Xiaodong Li, Yi Ma, and John Wright. Robust principal component analysis? *J. ACM*, 58(3):11:1–11:37, jun 2011. doi:10.1145/1970392.1970395.

6 Moses Charikar, Samir Khuller, David M. Mount, and Giri Narasimhan. Algorithms for facility location problems with outliers. In *Proceedings of the Twelfth Annual Symposium on Discrete Algorithms, January 7-9, 2001, Washington, DC, USA.*, pages 642–651, 2001. URL: http://dl.acm.org/citation.cfm?id=365411.365555.

7 Moses Charikar, Jacob Steinhardt, and Gregory Valiant. Learning from untrusted data. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2017, pages 47–60, New York, NY, USA, 2017. ACM. doi:10.1145/3055399.3055491.

8 Ke Chen. A constant factor approximation algorithm for k-median clustering with outliers. In *Proceedings of the Nineteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '08, pages 826–835, Philadelphia, PA, USA, 2008. Society for Industrial and Applied Mathematics. URL: `http://dl.acm.org/citation.cfm?id=1347082.1347173`.

9 Flavio Chierichetti, Sreenivas Gollapudi, Ravi Kumar, Silvio Lattanzi, Rina Panigrahy, and David P. Woodruff. Algorithms for lp low-rank approximation. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, pages 806–814, 2017. URL: `http://proceedings.mlr.press/v70/chierichetti17a.html`.

10 Eden Chlamtac, Michael Dinitz, and Robert Krauthgamer. Everywhere-sparse spanners via dense subgraphs. In *53rd Annual IEEE Symposium on Foundations of Computer Science, FOCS 2012, New Brunswick, NJ, USA, October 20-23, 2012*, pages 758–767, 2012. `doi:10.1109/FOCS.2012.61`.

11 Amit Deshpande and Santosh Vempala. Adaptive sampling and fast low-rank matrix approximation. In Josep Díaz, Klaus Jansen, José D. P. Rolim, and Uri Zwick, editors, *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, pages 292–303, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.

12 I. Diakonikolas, G. Kamath, D. M. Kane, J. Li, A. Moitra, and A. Stewart. Robust estimators in high dimensions without the computational intractability. In *2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 655–664, Oct 2016. `doi:10.1109/FOCS.2016.85`.

13 Ilias Diakonikolas, Daniel M. Kane, and Alistair Stewart. List-decodable robust mean estimation and learning mixtures of spherical gaussians. *CoRR*, abs/1711.07211, 2017. `arXiv:1711.07211`.

14 Uriel Feige and Joe Kilian. Heuristics for semirandom graph problems. *Journal of Computing and System Sciences*, 63:639–671, 2001.

15 Martin A. Fischler and Robert C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395, 1981. `doi:10.1145/358669.358692`.

16 Alan M. Frieze and Ravi Kannan. Quick approximation to matrices and applications. *Combinatorica*, 19(2):175–220, 1999. `doi:10.1007/s004930050052`.

17 Gene H. Golub and Charles F. Van Loan. *Matrix Computations (3rd Ed.)*. Johns Hopkins University Press, Baltimore, MD, USA, 1996.

18 Shalmoli Gupta, Ravi Kumar, Kefu Lu, Benjamin Moseley, and Sergei Vassilvitskii. Local search methods for k-means with outliers. *Proc. VLDB Endow.*, 10(7):757–768, 2017. `doi:10.14778/3067421.3067425`.

19 V. Guruswami and A. K. Sinop. Optimal column-based low-rank matrix reconstruction. In *SODA*, 2012.

20 Moritz Hardt and Ankur Moitra. Algorithms and hardness for robust subspace recovery. In *COLT 2013 - The 26th Annual Conference on Learning Theory, June 12-14, 2013, Princeton University, NJ, USA*, pages 354–375, 2013. URL: `http://jmlr.org/proceedings/papers/v30/Hardt13.html`.

21 Samuel B. Hopkins and Jerry Li. Mixture models, robustness, and sum of squares proofs. *CoRR*, abs/1711.07454, 2017. `arXiv:1711.07454`.

22 Peter J. Huber. *Robust Statistics*. Wiley, 1981.

23 Peter J. Huber and Elvezio M. Ronchetti. *Robust Statistics, 2nd Edition*. Wiley, 2009.

24 Pravesh K. Kothari and Jacob Steinhardt. Better agnostic clustering via relaxed tensor norms. *CoRR*, abs/1711.07465, 2017. `arXiv:1711.07465`.

25 Ravishankar Krishnaswamy, Shi Li, and Sai Sandeep. Constant approximation for k-median and k-means with outliers via iterative rounding. *CoRR*, abs/1711.01323, 2017.

26 K. A. Lai, A. B. Rao, and S. Vempala. Agnostic estimation of mean and covariance. In *2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 665–674, Oct 2016. doi:10.1109/FOCS.2016.76.

27 Konstantin Makarychev, Yury Makarychev, and Aravindan Vijayaraghavan. Learning communities in the presence of errors. In *Proceedings of the 29th Conference on Learning Theory, COLT 2016, New York, USA, June 23-26, 2016*, pages 1258–1291, 2016.

28 Frank McSherry. Spectral partitioning of random graphs. In *42nd Annual Symposium on Foundations of Computer Science, FOCS 2001, 14-17 October 2001, Las Vegas, Nevada, USA*, pages 529–537, 2001. doi:10.1109/SFCS.2001.959929.

29 Prasad Raghavendra and David Steurer. Graph expansion and the unique games conjecture. In Leonard J. Schulman, editor, *STOC*, pages 755–764. ACM, 2010. doi:10.1145/1806689.1806788.

30 Jacob Steinhardt, Moses Charikar, and Gregory Valiant. Resilience: A criterion for learning in the presence of arbitrary outliers. In *9th Innovations in Theoretical Computer Science Conference, ITCS 2018, January 11-14, 2018, Cambridge, MA, USA*, pages 45:1–45:21, 2018. doi:10.4230/LIPIcs.ITCS.2018.45.

31 G. W. Stewart and Ji guang Sun. *Matrix Perturbation Theory*. Academic Press, 1990.

32 H. Xu, C. Caramanis, and S. Mannor. Outlier-robust pca: The high-dimensional case. *IEEE Transactions on Information Theory*, 59(1):546–572, Jan 2013. doi:10.1109/TIT.2012.2212415.

33 H. Xu, C. Caramanis, and S. Sanghavi. Robust pca via outlier pursuit. *IEEE Transactions on Information Theory*, 58(5):3047–3064, May 2012. doi:10.1109/TIT.2011.2173156.

34 Xinyang Yi, Dohyung Park, Yudong Chen, and Constantine Caramanis. Fast algorithms for robust PCA via gradient descent. *CoRR*, abs/1605.07784, 2016. arXiv:1605.07784.

A Guessing the optimum

In all our algorithms, we assume that we have a guess for the optimum error, up to a multiplicative factor of $(1+\epsilon)$. Note that the error is $\|B - B_k\|_F^2$, as defined in the introduction. We now argue that we can always obtain a polynomial number of guesses, one of which is accurate. This follows immediately if we can show that the error lies in a range $[L, U]$, where U/L is at most $\exp(\text{poly}(n, b))$, where b is the total bit complexity of the input. (This is because we can discretize the range into $\text{poly}(n, b)/\epsilon$ intervals using multiples of $(1+\epsilon)$.)

This is not immediate in our setting because $\|B - B_k\|_F^2$ can actually be zero. But we can show that if the error is *non-zero*, it can be lower bounded by $\exp(-\text{poly}(n, b))$. Such results are quite well-known (see [17]), but we sketch a short proof below.

Exponential range for the optimum. The key claim is that if a $d \times k$ matrix C has linearly independent columns, it has $\sigma_{\min} \geq \exp(-\text{poly}(n, b))$, where b is the total bit complexity of C. This can be proved as follows. Note that $\sigma_{\min}$ is the smallest eigenvalue of $C^T C$. Now the characteristic polynomial of $C^T C$ has coefficients that are all at most $\exp(\text{poly}(n, b))$, as they are appropriate determinants. Suppose the polynomial is $\sum_{i=0}^{k} a_i \lambda^i$. By linear independence, we know that $a_0 \neq 0$. Further, a_0 is the determinant of $C^T C$, and hence is bounded from below by $\exp(-\text{poly}(n, b))$; this is because the sum of a set of numbers of a certain precision is either zero or is at least the "least count" of that precision.

Now, the magnitude of the smallest non-zero real root of any polynomial can be bounded from below by $|a_0|/(|a_0| + |a_1| + \cdots + |a_d|)$. As the a_i are all at most $\exp(\text{poly}(n, b))$, the desired conclusion follows.