Streaming, Memory-limited PCA

Ioannis Mitliagkas[t], Constantine Caramanis[t], Prateek Jain[m] *

[t] The University of Texas at Austin

[m] Microsoft Research India, Bangalore

April 12, 2013

Abstract

In this paper, we consider a streaming one-pass-over-the-data model for Principal Component Analysis (PCA). The input, in this case, is a stream of p-dimensional vectors, and the output is a collection of k, p-dimensional principal components that span the best approximating subspace. Consequently, the minimum memory requirement for such problems is $O(kp)$. Yet the standard PCA algorithm requires us to form the empirical covariance matrix, typically a dense $p \times p$ matrix, hence requiring $O(p^2)$ memory. Although there exist several incremental algorithms that require $O(kp)$ memory, to the best of our understanding, these methods currently do not have known finite-sample performance bounds. That is, in the high-dimensional setting where the number of samples and dimensionality scale together, there is no known provably correct algorithm. This paper considers this simple but important problem. We give what is to the best of our knowledge, the first streaming algorithm requiring only $O(kp)$ memory, that makes a single pass over the data, and whose performance matches the standard batch algorithm up to logarithmic factors.

1 Introduction

Principal Component Analysis is a fundamental tool for dimensionality reduction, clustering, classification, and indeed many learning tasks. Various recent works (e.g., see Vershynin (2010a)) have explored the power of PCA in the high dimensional batch setting with sub-Gaussian noise, demonstrating that SVD of the empirical covariance matrix recovers the principal components (equivalently, covariance estimation) when the number of samples scales as $n = O(p \log p)$, where p is the dimension of the ambient space. In the noisy setting (which is of most interest) the empirical covariance matrix is a dense $p \times p$ matrix, and hence requires $O(p^2)$ memory to store.

The quadratic (in p) memory requirement is prohibitive in several important settings where the data points could be high resolution photographs, sound recordings, biometrics, or video. Massive storage systems, in particular the cloud, offer the promise of essentially limitless storage and distributed computing power. Still, for many real systems, the quadratic storage requirement is simply not an option. Interestingly, many devices have processors and available RAM that permit computations manipulating vectors of length p, but do not have hard drive space even approaching p^2. A typical desktop may have 10-20 Gb of RAM, but typically will not have more than a few Tb of total storage. A smart-phone may have a few Mb of RAM, but has Gb not Tb of storage. That

*Email: {ioannis,constantine}@utexas.edu;pjain9@gmail.com

is, at multiple computing scales, manipulating vectors of length $O(p)$ is possible when storage of $O(p^2)$ is not.

This regime is of particular interest in the *streaming data* setting. Here, data are generated sequentially, and hence nowhere stored. If an algorithm does not explicitly call for storing a given data point, it can never be revisited; hence, only one pass over the data is possible. In this case, the input data each take $O(p)$ to store, and the desired output requires $O(kp)$. Do we fundamentally need $O(p^2)$ storage? This paper proves that the answer is no; we provide an algorithm that requires no more than $O(kp)$ storage, yet exhibits the same sample-complexity performance as the full-blown Batch-SVD algorithm. For this problem, we consider the basic generative model for PCA; that is, we assume that points $\mathbf{x}_i \in \mathbb{R}^p$ are generated according to $\mathbf{x}_i = A\mathbf{z}_i + \mathbf{w}_i$, where A is a $p \times k$ matrix defining the principal components and $\mathbf{z}_i$ is a multivariate normal random variable, and $\mathbf{w}_i$ is a Gaussian noise vector with σ^2 variance along each direction. Note that this setting provides for samples $\mathbf{x}_i$ that have poor SNR as $\|A\mathbf{z}_i\|_2/\|\mathbf{w}_i\|_2 \approx 1/\sqrt{p}$.

For this problem, we provide a stochastic algorithm that divides the data into blocks and updates its estimate only once in every block. We also provide *finite sample* analysis of our algorithm that shows, in particular, that in the case when $k \ll p$, our algorithm has the same sample complexity as the batch algorithm, up to an additional log factor in p. On the other hand, our algorithm never requires storage of more than $k+1$ p-length vectors, where k is the number of principal components requested. Finally, the amount of computation required per data point is also $O(p)$.

Our contribution: To the best of our knowledge, our finite sample analysis is the first for an algorithm for PCA in the memory-limited setting. Furthermore, our analysis highlights several key aspects of the problem. One being, dependence on the initial guess for the principal components. In the batch setting, algorithms either do not require initialization (SVD), or the initial guess simply needs to have some non-zero component along the right direction. There, the extent of overlap between the true components and initial components only affects the computational time but not the sample complexity. On the other hand, in the streaming setting, initialization becomes more critical as a very poor initial point leads to a penalty in sample complexity. Our analysis also demonstrates the interplay between the ambient dimension and noise variance and their relative effect to the scaling off sample complexity.

Finally, to corroborate our theoretical analysis, we provide extensive empirical evaluation of our methods on synthetic datasets sampled from the assumed generative model. As expected our method performs competitively with the popular stochastic approximation method Oja & Karhunen (1985), which despite widespread empirical success has still eluded rigorous finite sample analysis.

2 Related Work

Computing principal components quickly and iteratively with controlled computation and controlled memory costs has long been recognized as an important problem. Memory- and computation-efficient algorithms that operate on streaming data are plentiful in the literature. Yet, while there are algorithms that empirically seem to do well, there is no algorithm that provably recovers the principal components in the same noise and sample-complexity regime that the batch PCA algorithm is able to do. Indeed, there has been renewed interest recently in this problem, and the fact that this is an important unresolved issue has been pointed out in numerous places, e.g., Arora et al. (2012).

There has been more work than possible to list. We survey some of the most relevant work here. In particular, we discuss the relationship to EM-based methods, online-PCA, incremental models for SVD and PCA and stochastic-approximation-based methods.

In a Bayesian mindset, some researchers have come up with expectation maximization approaches Roweis (1998); Tipping & Bishop (1999), that can arguably be used in an incremental fashion. Unfortunately, in this body of literature one can only find consistency results at best, and the finite sample behavior is not known.

Online-PCA with the objective of *regret minimization* has been considered in several papers, most recently in Warmuth & Kuzmin (2008), where the multiplicative weights approach is adapted for this problem (now experts correspond to subspaces). The goal there is to control the regret, improving on the natural follow-the-leader algorithm that performs batch-PCA at each step. However, the algorithm can require $O(p^2)$ memory, in order to store the multiplicative weights and even the memory-light variant described in Arora et al. (2012) comes with no guarantee for less than that.

Algorithms focused on sequential SVD (e.g., Brand (2002, 2006), Comon & Golub (1990),Li (2004)) seek to have the best subspace estimate at every time (i.e., each time a new data sample arrives) but without performing full-blown SVD at each step. While these algorithms indeed reduce both the computational and memory burden of batch-PCA, there are no rigorous guarantees on the quality of the principal components or on the *statistical performance* of these methods. More recent approaches Balzano et al. (2010); He et al. (2011) obviate the need for re-orthogonalization by taking into consideration the geodesics of the Grassmanian manifold, but also fail to come up with sample complexity bounds, or even guarantees of universal consistency.

A very interesting class of algorithms, start by left-multiplying the data matrix with a subsampling or sketching matrix. Then those algorithms perform SVD on the resulting, smaller, matrix giving some guarantees on the quality of the results; see Clarkson & Woodruff (2009) for an overview of the state of the art. The one disadvantage is that in those results, the memory requirement also scales with respect to the desired precision, ϵ, unlike the kind of results we are providing. Also, sketching results seem to inherently require a Frobenius norm of the tail in the error bounds. It is not clear how these can be easily adapted for bounds on the top principal components.

Another line of work, that is also closely related to the sequential SVD approach, is stochastic algorithms that are derived from a stochastic approximation perspective. Such methods go under a variety of names, including *Incremental PCA* (though the term *Incremental* has been used in the online setting as well Herbster & Warmuth (2001)). There have been several algorithms proposed in the spirit of stochastic approximation, that have $O(p)$ storage requirements, e.g., Weng et al. (2003); Arora et al. (2012). The basic algorithms are some version of the following: upon receiving data point $\mathbf{x}_t$ at time t, update the estimate of the top k principal components via:

$$U^{(t+1)} = \text{Proj}(U^{(t)} + \eta_t \mathbf{x}_t \mathbf{x}_t^\top U^{(t)}), \tag{1}$$

where $\text{Proj}(\cdot)$ denotes the "projection" that takes the SVD of the argument, and sets the top k singular values to 1 and the rest to zero (see Arora et al. (2012) for further discussion).

From a time/space complexity standpoint, stochastic approximation-style algorithms are appealing, as they require $O(p)$ complexity per iteration – just enough for vector inner products and additions. Several studies have shown that these algorithms perform well empirically and our simulations, in Section 5, support this.

However, to the best of our knowledge (and efforts) there does not exist any rigorous finite sample guarantee for these algorithms. Most of the studies that analyze these algorithms have asymptotic, consistency type of results and do not offer finite sample bounds. For instance, analysis in Weng et al. (2003) depends on the standard Robbins-Monro type analysis Robbins & Monro (1951) which does not scale to high dimensional regime, where the ambient dimension (and thus

noise magnitude) and the number of samples scale together. A related work by Zhao et al. (2006) (also Zha & Simon (1999)) considers problems where dimensionality is significantly higher than the number of samples available, but requires $O(n^2)$ storage and hence is not feasible in a more realistic streaming setting.

A key difficulty in the analysis of stochastic methods for PCA has been the fact that the variance at each step of the algorithm may be large and hence the standard concentration inequalities will give vacuous bounds. In this work, we instead ensure that the variance at each step of the algorithm is controlled and hence finite sample analysis can be performed. Our algorithm is very simple and easy to implement, but still allows for sharp bounds and matches the performance of the batch methods (up to a logarithmic factor in the number of samples required).

3 Problem Formulation and Notation

We consider a streaming model, where at each time step t, we receive a point $\mathbf{x}_t \in \mathbb{R}^p$. Furthermore, any vector that is not explicitly stored can never be revisited. Now, our goal is to compute the top k principal components of the data: the k-dimensional subspace that offers the best squared-error estimate for the points. We assume a probabilistic generative model, from which the data is sampled at each step t. Specifically, we assume,

$$\mathbf{x}_t = A\mathbf{z}_t + \mathbf{w}_t, \tag{2}$$

where $A \in \mathbb{R}^{p\times k}$ is a fixed matrix, $\mathbf{z}_t \in \mathbb{R}^{k\times 1}$ is a multivariate normal random variable, i.e.,

$$\mathbf{z}_t \sim \mathcal{N}(0_{k\times 1}, I_{k\times k}),$$

and vector $\mathbf{w}_t \in \mathbb{R}^{p\times 1}$ is the "noise" vector and is also sampled from a multivariate normal distribution, i.e.,

$$\mathbf{w}_t \sim \mathcal{N}(0_{p\times 1}, \sigma^2 I_{p\times p}).$$

Furthermore, we assume that all $2n$ random vectors $(\mathbf{z}_t, \mathbf{w}_t, \forall 1 \le t \le n)$ are mutually independent.

In this regime, it is well-known that batch-PCA is asymptotically consistent (hence recovering A up to unitary transformations) with number of samples scaling as $n = O(p)$ Vershynin (2010b). It is interesting to note that in this high-dimensional regime, the signal-to-noise ratio quickly approaches zero, as the signal, or "elongation" of the major axis, $\|Az\|_2$, is $O(1)$, while the noise magnitude, $\|\mathbf{w}\|_2$, scales as $O(\sqrt{p})$. The central goal of this paper is to provide finite sample guarantees for a streaming algorithm that requires memory no more than $O(kp)$ and matches the consistency results of batch PCA in the sampling regime $n = O(p)$ (possibly with additional log factors, or factors depending on σ and k).

We denote matrices by capital letters (e.g. A) and vectors by lower-case bold-face letters ($\mathbf{x}$). $\|\mathbf{x}\|_q$ denotes the ℓ_q norm of $\mathbf{x}$; $\|\mathbf{x}\|$ denotes the ℓ_2 norm of $\mathbf{x}$. $\|A\|$ or $\|A\|_2$ denotes the spectral norm of A while $\|A\|_F$ denotes the Frobenius norm of A. Without loss of generality (WLOG), we assume that: $\|A\|_2 = 1$, where $\|A\|_2 = \max_{\|\mathbf{x}\|_2=1} \|A\mathbf{x}\|_2$ denotes the spectral norm of A. Finally, we write $\langle \mathbf{a}, \mathbf{b}\rangle = \mathbf{a}^\top \mathbf{b}$ for the inner product between $\mathbf{a}$, $\mathbf{b}$. In proofs the constant C is used loosely and its value may vary from line to line.

4 Algorithm and Main Results

In this section, we present our proposed algorithm and its finite sample analysis. It is a block-wise stochastic variant of the classical power-method. Stochastic versions of the power method are

Algorithm 1 Block-Stochastic Power Method

input $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, Block size: B

$\mathbf{q}_0 \sim \mathcal{N}(0, I_{p\times p})$ (Initialization)
$\mathbf{q}_0 = \mathbf{q}_0/\|\mathbf{q}_0\|_2$
for $\tau = 0, \ldots, n/B - 1$ **do**
  $\mathbf{s}_{\tau+1} \leftarrow 0$
  for $t = B\tau + 1, \ldots, B(\tau + 1)$ **do**
    $\mathbf{s}_{\tau+1} \leftarrow \mathbf{s}_{\tau+1} + \frac{1}{B}\langle \mathbf{q}_\tau, \mathbf{x}_t\rangle x_t$
  end for
  $\mathbf{q}_{\tau+1} \leftarrow \mathbf{s}_{\tau+1}/\|\mathbf{s}_{\tau+1}\|_2$
end for

output

already popular in the literature and are known to have good empirical performance; see Arora et al. (2012) for a nice review of such methods. However, as discussed above, a main impediment to the analysis of such stochastic algorithms (as in (1)) is the potentially large variance of each step, due primarily to the high-dimensional regime we consider, and the vanishing SNR.

This motivated us to consider a modified stochastic power method algorithm, that has a variance reduction step built in. At a high-level, our method updates only once in a "block" and within one block we average out noise to reduce the variance.

Below, we first illustrate the main ideas of our method as well as our sample complexity proof for the simpler rank-1 case. We then describe our general rank-k algorithm and provide its analysis in Section 4.2. We note that, while our algorithm describes $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ as "input," we mean this in the streaming sense: the data are no-where stored, and can never be revisited unless the algorithm explicitly stores them.

4.1 Rank-One Case

We first consider the rank-1 case for which each sample $\mathbf{x}_t$ is generated using: $\mathbf{x}_t = \mathbf{u}\mathbf{z}_t + \mathbf{w}_t$ where $\mathbf{u} \in \mathbb{R}^p$ is the principal component that we wish to recover. Our algorithm is a block-wise method where all the n samples are divided in n/B blocks (for simplicity we assume that n/B is an integer). In the $(\tau + 1)$-st block, we compute

$$\mathbf{s}_{\tau+1} = \left(\frac{1}{B}\sum_{t=B\tau+1}^{B(\tau+1)} \mathbf{x}_t\mathbf{x}_t^\top\right)\mathbf{q}_\tau. \tag{3}$$

Then, the iterate $\mathbf{q}_\tau$ is updated using $\mathbf{q}_{\tau+1} = \mathbf{s}_{\tau+1}/\|\mathbf{s}_{\tau+1}\|_2$. Note that, $\mathbf{s}_{\tau+1}$ can be easily computed in an online manner where $O(p)$ operations are required per step. Furthermore, storage requirements are also linear in p.

4.1.1 Analysis

We now present the sample complexity analysis of our proposed method (Algorithm 1). At a high-level, we show that, using $\Omega(\sigma^4 p \log(p)/\epsilon^2)$ samples, Algorithm 1 obtains a solution $\mathbf{q}_T$ of accuracy ϵ, i.e. $\|\mathbf{q}_T - \mathbf{u}\|_2 \leq \epsilon$.

Theorem 1. *Let $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ where $\mathbf{x}_t \in \mathbb{R}^p, \forall t$ is generated by (2). Set the total number of iterations $T = \Omega(\frac{\log(p/\epsilon)}{\log((\sigma^2+.75)/(\sigma^2+.5))})$ and the block size $B = \Omega(\frac{(1+3(\sigma+\sigma^2)\sqrt{p})^2\log(T)}{\epsilon^2})$. Then, with*

probability 0.99, $\|\mathbf{q}_T - \mathbf{u}\|_2 \leq \epsilon$, *where* $\mathbf{q}_T$ *is the* T*-th iterate of Algorithm 1. That is, Algorithm 1 obtains an* ϵ*-accurate solution with number of samples* (n) *given by:*

$$n = \tilde{\Omega}\left(\frac{(1+3(\sigma+\sigma^2)\sqrt{p})^2\log(p/\epsilon)}{\epsilon^2\log((\sigma^2+.75)/(\sigma^2+.5))}\right).$$

Note that in the total sample complexity, we use the notation $\tilde{\Omega}(\cdot)$ *to suppress the extra* $\log(T)$ *factor for clarity of exposition, as* T *already appears in the expression linearly.*

Proof. Our proof of the theorem has three critical components: (a) we show that for large enough B, the empirical covariance matrix $F_{\tau+1} = \frac{1}{B}\sum_{t=B\tau+1}^{B(\tau+1)} \mathbf{x}_t\mathbf{x}_t^\top$ is close to the true covariance matrix $M = \mathbf{u}\mathbf{u}^\top + \sigma^2 I$, i.e. $\|F_{\tau+1} - M\|_2$ is small. In the process, we obtain "tighter" bounds for $\|\mathbf{u}^\top(F_{\tau+1} - M)\mathbf{u}\|$ for *fixed* $\mathbf{u}$; (b) we show that, with probability 0.99 (or any other constant probability), the initial point $\mathbf{q}_0$ has a component of at least $O(1/\sqrt{p})$ magnitude along the true direction $\mathbf{u}$; (c) we then use these concentration bounds and the bound on the "goodness" of the initial step to show that after τ iterations, the error in estimation is at most $O(\gamma^\tau)$ where $\gamma < 1$ is a constant. Step (a) is proved in Lemmas 2 and 3, while Lemma 4 provides the required result for the initial vector $\mathbf{q}_0$. Using these lemmas, we next complete the proof of the theorem. We note that both (a) and (b) follow from well-known results; we provide them for completeness.

Let $\mathbf{q}_\tau = \sqrt{1-\delta_\tau}\mathbf{u} + \sqrt{\delta_\tau}\mathbf{g}_\tau$, $1 \leq \tau \leq n/B$, where $\mathbf{g}_\tau$ is the component of $\mathbf{q}_\tau$ that is perpendicular to $\mathbf{u}$ and $\sqrt{1-\delta_\tau}$ is the magnitude of the component of $\mathbf{q}_\tau$ along $\mathbf{u}$. Note that $\mathbf{g}_\tau$ may well change at each iteration; the important thing for us is to track the magnitude of the component in the direction of $\mathbf{u}$.

Now, using Lemma 3 and assuming B, T as given in the theorem, with probability at least $1 - C/T$, where $C > 0$ is an arbitrary constant, the following holds,

$$\mathbf{u}^\top\mathbf{s}_{\tau+1} \geq \sqrt{1-\delta_\tau}(1+\sigma^2)\left(1 - \frac{\epsilon}{4(1+\sigma^2)}\right). \tag{4}$$

Next, we consider the component of $\mathbf{s}_{\tau+1}$ that is perpendicular to $\mathbf{u}$:

$$\begin{aligned}\mathbf{g}_{\tau+1}^\top\mathbf{s}_{\tau+1} &= \mathbf{g}_{\tau+1}^\top\left(\frac{1}{B}\sum_{t=B\tau+1}^{B(\tau+1)}\mathbf{x}_t\mathbf{x}_t^\top\right)\mathbf{q}_\tau \\ &= \mathbf{g}_{\tau+1}^\top(M + E_\tau)\mathbf{q}_\tau,\end{aligned}$$

where $M = \mathbf{u}\mathbf{u}^\top + \sigma^2 I$ and E_τ is the error matrix: $E_\tau = M - \frac{1}{B}\sum_{t=B\tau+1}^{B(\tau+1)}\mathbf{x}_t\mathbf{x}_t^\top$.Using Lemma 2, $\|E_\tau\|_2 \leq \epsilon$ (w.p. $\geq 1 - C/T$). Hence, w.p. $\geq 1 - C/T$:

$$\begin{aligned}\mathbf{g}_{\tau+1}^\top\mathbf{s}_{\tau+1} &= \sigma^2\mathbf{g}_{\tau+1}^\top\mathbf{q}_\tau + \|\mathbf{g}_{\tau+1}\|_2\|E_\tau\|_2\|\mathbf{q}_\tau\|_2, \\ &\leq \sigma^2\mathbf{g}_{\tau+1}^\top(\sqrt{1-\delta_\tau}\mathbf{u} + \sqrt{\delta_\tau}\mathbf{g}_\tau) + \epsilon, \\ &\leq \sigma^2\sqrt{\delta_\tau} + \epsilon.\end{aligned} \tag{5}$$

Now, since $\mathbf{q}_{\tau+1} = \mathbf{s}_{\tau+1}/\|\mathbf{s}_{\tau+1}\|_2$,

$$\begin{aligned}\delta_{\tau+1} = (\mathbf{g}_{\tau+1}^\top\mathbf{q}_{\tau+1})^2 &= \frac{(\mathbf{g}_{\tau+1}^\top\mathbf{s}_{\tau+1})^2}{(\mathbf{u}^\top\mathbf{s}_{\tau+1})^2 + (\mathbf{g}_{\tau+1}^\top\mathbf{s}_{\tau+1})^2}, \\ &\overset{(i)}{\leq} \frac{(\mathbf{g}_{\tau+1}^\top\mathbf{s}_{\tau+1})^2}{(1-\delta_\tau)\left(1+\sigma^2-\frac{\epsilon}{4}\right)^2 + (\mathbf{g}_{\tau+1}^\top\mathbf{s}_{\tau+1})^2}, \\ &\overset{(ii)}{\leq} \frac{(\sigma^2\sqrt{\delta_\tau}+\epsilon)^2}{(1-\delta_\tau)\left(1+\sigma^2-\frac{\epsilon}{4}\right)^2 + (\sigma^2\sqrt{\delta_\tau}+\epsilon)^2},\end{aligned} \tag{6}$$

where, (i) follows from (4) and (ii) follows from (5) along with the fact that $\frac{x}{c+x}$ is an increasing function in x for $c, x \geq 0$.

Assuming $\sqrt{\delta_\tau} \geq 2\epsilon$ and using (6) and bounding the failure probability with a union bound, we get (w.p. $\geq 1 - \tau \cdot C/T$)

$$\begin{aligned}
\delta_{\tau+1} &\leq \frac{\delta_\tau(\sigma^2 + 1/2)^2}{(1-\delta_\tau)(\sigma^2+3/4)^2 + \delta_\tau(\sigma^2+1/2)^2}, \\
&\overset{(i)}{\leq} \frac{\gamma^{2\tau}\delta_0}{1-(1-\gamma^{2\tau})\delta_0}, \\
&\overset{(ii)}{\leq} C_1\gamma^{2\tau}p, \qquad (7)
\end{aligned}$$

where $\gamma = \frac{\sigma^2+1/2}{\sigma^2+3/4}$ and $C_1 > 0$ is a global constant. (i) follows from Lemma 5 and (ii) follows from Lemma 4.

Hence, using the above equation after $T = \Omega\left(\log(p/\epsilon)/\log\left(1/\gamma\right)\right)$ updates, with probability at least $1 - C$, $\sqrt{\delta_T} \leq 2\epsilon$. The result now follows by noting that $\|\mathbf{u} - \mathbf{q}_T\|_2 \leq 2\sqrt{\delta_T}$ and by selecting constant C to be small enough, say ≈ 0.001. □

Remark: Note that in Theorem 1, the probability of accurate principal component recovery is a constant and does not decay with p. However, by running $O(\log p)$ instances of Algorithm 1 in parallel, at least one of the instances should correctly recover the true component (within ϵ error) with probability at least $1 - \frac{1}{p^{O(1)}}$. Note that this does not increase the sample complexity, it only increases the computation time and storage requirements by a factor of $O(\log p)$. Alternatively, we can run Algorihm 1 $O(\log p)$ times on fresh data each time, using the next block of data to evaluate the old solutions, always keeping the best one. At a cost of an additional $O(\log p)$ factor in sample complexity, we again guarantee a success probability of at least $1 - \frac{1}{p^{O(1)}}$.

We now present the lemmas used by our proof given above. For the more technical results the proofs are deferred to the appendix.

The following lemma is used to bound the difference between $\frac{1}{B}\sum_{B\tau<t\leq B(\tau+1)} \mathbf{x}_t\mathbf{x}_t^\top$ and M. It provides a concentration on the spectral norm of the covariance estimate error using recent statistical results.

Lemma 2. *Let $\mathcal{X}$, B, T be as defined in Theorem 1. Then, w.p. $1 - C/T$ we have:*

$$\left\| \frac{1}{B}\sum_t \mathbf{x}_t\mathbf{x}_t^\top - \mathbf{u}\mathbf{u}^\top - \sigma^2 I \right\|_2 \leq \epsilon.$$

The full proof is provided in Appendix B.

The following lemma provides a probabilistic lower bound on the magnitude of the new estimate along the real principal component, $\mathbf{u}$.

Lemma 3. *Let $\mathcal{X}$, B, T be as defined in Theorem 1. Then, w.p. $1 - C/T$ we have:*

$$\mathbf{u}^\top\mathbf{s}_{\tau+1} \geq \mathbf{u}^\top\mathbf{q}_\tau(1+\sigma^2)\left(1 - \frac{\epsilon}{4(1+\sigma^2)}\right),$$

where $\mathbf{s}_t = \frac{1}{B}\sum_{B\tau<t\leq B(\tau+1)} \mathbf{x}_t\mathbf{x}_t^\top\mathbf{q}_\tau$.

For the full proof, please refer to Appendix C.

Lemma 4. *Let $\mathbf{q}_0$ be the initial guess for $\mathbf{u}$, given by Steps 1 and 2 of Algorithm 1. Then, w.p. 0.99: $\langle \mathbf{q}_0, \mathbf{u} \rangle \geq \frac{C_0}{\sqrt{p}}$, where $C_0 > 0$ is a universal constant.*

Proof. Using standard tail bounds for Gaussians (see Lemma 13), $\|\mathbf{q}_0\|_2 \leq 2\sqrt{p}$ with probability $1 - \exp(-C_1 p)$, where $C_1 > 0$ is a universal constant. Furthermore, $(\|\mathbf{q}_0\|_2 \mathbf{q}_0)^\top \mathbf{u} \sim N(0, 1)$. Hence, there exists $C_0 > 0$, s.t., with probability 0.99, $(\|\mathbf{q}_0\|_2 \mathbf{q}_0)^T \mathbf{u} \geq C_0$. Hence, $\mathbf{q}_0^\top \mathbf{u} \geq \frac{C_0}{2\sqrt{p}}$. □

Finally, we present the following lemma that shows that the recursion given by (7) decreases δ_τ at a fast rate. Interestingly, the rate of decrease in error δ_τ initially (for small τ) might be sub-linear but for large enough τ the rate turns out to be linear.

Lemma 5. *If for any $\tau \geq 0$ and $0 < \gamma < 1$, we have $\delta_{\tau+1} \leq \frac{\gamma^2 \delta_\tau}{1 - \delta_\tau + \gamma^2 \delta_\tau}$, then,*

$$\delta_{\tau+1} \leq \frac{\gamma^{2t+2} \delta_0}{1 - (1 - \gamma^{2t+2}) \delta_0}.$$

Proof. We prove the lemma using induction. The base case (for $\tau = 0$) follows trivially.

Now, by the inductive hypothesis, $\delta_\tau \leq \frac{\gamma^{2t} \delta_0}{1 - (1 - \gamma^{2t}) \delta_0}$. That is,

$$\frac{1}{\delta_\tau} \geq \frac{1 - (1 - \gamma^{2t}) \delta_0}{\gamma^{2t} \delta_0}.$$

Finally, by assumption,

$$\delta_{\tau+1} \leq \frac{\gamma^2}{\frac{1}{\delta_\tau} - (1 - \gamma^2)} \leq \frac{\gamma^2}{\frac{1 - (1 - \gamma^{2t}) \delta_0}{\gamma^{2t} \delta_0} - (1 - \gamma^2)}.$$

The lemma follows after simplification of the above given expression. □

4.2 General Rank-k Case

In this section, we consider the general rank-k PCA problem where each sample is assumed to be generated using the model of equation (2), where $A \in \mathbb{R}^{p \times k}$ represents the k principal components that need to be recovered. Let $A = U \Lambda V^\top$ be the SVD of A where $U \in \mathbb{R}^{p \times k}$, $\Lambda, V \in \mathbb{R}^{k \times k}$. The matrices U and V are orthogonal, i.e. $U^\top U = I, V^\top V = I$, and Σ is a diagonal matrix with diagonal elements $\lambda_1 \geq \lambda_2 \cdots \geq \lambda_k$. Then, the goal is to recover the space spanned by A, i.e. $span(U)$. Without loss of generality, we can assume that $\|A\|_2 = \lambda_1 = 1$.

Similar to the rank-1 problem, our algorithm for the rank-k problem (see Algorithm 2) can be viewed as a streaming variant of the classical orthogonal iteration used for SVD. Our algorithm proceeds in a blockwise manner; for each block of samples, we compute

$$S_{\tau+1} = \left(\frac{1}{B} \sum_{t=B\tau+1}^{B(\tau+1)} \mathbf{x}_t \mathbf{x}_t^\top \right) Q_\tau, \tag{8}$$

where $Q_\tau \in \mathbb{R}^{p \times k}$ is the τ-th block iterate and is an orthogonal matrix, i.e., $Q_\tau^\top Q_\tau = I_{k \times k}$. Given $S_{\tau+1}$, the next iterate, $Q_{\tau+1}$, is computed by the QR-decomposition of $S_{\tau+1}$. That is,

$$S_{\tau+1} = Q_{\tau+1} R_{\tau+1}, \tag{9}$$

where $R_{\tau+1} \in \mathbb{R}^{k \times k}$ is an upper-triangular matrix. Note that $S_{\tau+1}$ can be computed in a streaming fashion, i.e., none of the points need to be stored. The computational complexity is $O(pk)$ for "within" block updates (step 6 of Algorithm 2) and $O(pk^2)$ for updating Q_τ (step 8 of Algorithm 2). The space required by our method is $O(pk)$.

Algorithm 2 Block-Stochastic Orthogonal Iteration

input $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, Block size: B

$H^i \sim \mathcal{N}(0, I_{p\times p}), 1 \le i \le k$ (Initialization)

$H = Q_0 R_0$ (QR-decomposition)

for $\tau = 0, \ldots, n/B - 1$ **do**

  $S_{\tau+1} \leftarrow 0$

  for $t = B\tau + 1, \ldots, B(\tau + 1)$ **do**

    $S_{\tau+1} \leftarrow S_{\tau+1} + \frac{1}{B} x_t x_t^\top Q_\tau$

  end for

  $S_{\tau+1} = Q_{\tau+1} R_{\tau+1}$ (QR-decomposition)

end for

output

4.2.1 Analysis

We now present our analysis for the general rank-k problem and provide sharp sample complexity bounds. Unlike the rank-1 case, where the choice of distance between vectors u and v is clear, there are several natural distance functions in the rank-k case. We use the following largest-principal-angle-based distance function between any two given subspaces:

Definition 6. *Let $U, V \in \mathbb{R}^{p\times k}$ represent the orthogonal basis of subspaces span(U) and span(V), respectively. Then, the distance between span(U) and span(V) is given by*

$$\begin{aligned} dist\left(span(U), span(V)\right) &= dist(U, V) \\ &= \|U_\perp^\top V\|_2 = \|V_\perp^\top U\|_2, \end{aligned}$$

where $U_\perp$ and $V_\perp$ represents an orthogonal basis of the perpendicular subspace to span(U) and span(V), respectively.

We now present our main theorem which shows that using $\Omega(C_{\sigma,\lambda_k} p \log(p/\epsilon)/\epsilon^2)$ samples, Algorithm 2 produces a subspace basis Q_τ, which with high probability satisfies $\text{dist}(Q_\tau, U) \le \epsilon$. $C_{\sigma,\lambda_k} > 0$ is a constant depending only on σ, λ_k.

Theorem 7. *Let $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ where $\mathbf{x}_t \in \mathbb{R}^p$ for every t is generated by (2), and the SVD of $A \in \mathbb{R}^{p\times k}$ is given by $A = U\Lambda V^\top$. Let, wlog, $\lambda_1 = 1 \ge \lambda_2 \ge \cdots \ge \lambda_k > 0$. Let,*

$$T = \Omega\left(\log(p/k\epsilon)/\log\left(\frac{\sigma^2 + 0.75\lambda_k^2}{\sigma^2 + 0.5\lambda_k^2}\right)\right).$$

Also, let the block size be,

$$B = \Omega\left(\frac{\left((1+\sigma)^2\sqrt{k} + \sigma\sqrt{1+\sigma^2}k\sqrt{p}\right)^2 \log(T)}{\lambda_k^4 \epsilon^2}\right).$$

Then, after T block-updates, w.p. 0.99, dist$(U, Q_T) \le \epsilon$. Hence, the sufficient number of samples for ϵ-accurate recovery of all the top-k principal components is:

$$n = \tilde{\Omega}\left(\frac{\left((1+\sigma)^2\sqrt{k} + \sigma\sqrt{1+\sigma^2}k\sqrt{p}\right)^2 \log(p/k\epsilon)}{\lambda_k^4 \epsilon^2 \log\left(\frac{\sigma^2+0.75\lambda_k^2}{\sigma^2+0.5\lambda_k^2}\right)}\right).$$

Again, for the total sample complexity, we use the notation $\tilde{\Omega}(\cdot)$ to suppress the extra $\log(T)$ factor for clarity of exposition, as T already appears in the expression linearly.

Please refer to the Remark after Algorithm 1 for a couple of ways to get the same result with high probability.

As in the proof for the rank-1 problem, we first show that in each block, $F_{\tau+1} = \frac{1}{B}\sum_{t=B\tau+1}^{B(\tau+1)} \mathbf{x}_t\mathbf{x}_t^\top$ is close to the true "covariance" matrix $M = AA^\top + \sigma^2 I$, and that the initial iterate Q_0 has large enough component along the true subspace, $span(U)$. Finally, we combine these two components to provide a recursion that ensures that $\text{dist}(U, Q_\tau)$ decreases at a fast rate.

Compared to the rank-1 case, we require a more careful analysis as we need to bound spectral norms of various quantities in intermediate steps and simple, crude analysis can lead to significantly worse bounds. Interestingly, the analysis is entirely different from the standard analysis of the orthogonal iteration as there, the empirical estimate of the covariance matrix is fixed while in our case it varies with each block.

Proof. By using update for $Q_{\tau+1}$ (see (8), (9)):

$$Q_{\tau+1}R_{\tau+1} = F_{\tau+1}Q_\tau, \tag{10}$$

where $F_{\tau+1} = \frac{1}{B}\sum_{B\tau<t\leq B(\tau+1)} x_t x_t^\top$. That is,

$$U_\perp^\top Q_{\tau+1}R_{\tau+1}\mathbf{v} = U_\perp^\top F_{\tau+1}Q_\tau\mathbf{v}, \quad \forall \mathbf{v}\in\mathbb{R}^k, \tag{11}$$

where $U_\perp$ is an orthogonal basis of the subspace orthogonal to $span(U)$. Now, let $\mathbf{v}_1$ be the singular vector corresponding to the largest singular value, then:

$$\begin{aligned}
\|U_\perp^\top Q_{\tau+1}\|_2^2 &= \frac{\|U_\perp^\top Q_{\tau+1}\mathbf{v}_1\|_2^2}{\|\mathbf{v}_1\|_2^2} = \frac{\|U_\perp^\top Q_{\tau+1}R_{\tau+1}\tilde{\mathbf{v}}_1\|_2^2}{\|R_{\tau+1}\tilde{\mathbf{v}}_1\|_2^2} \\
&\overset{(i)}{=} \frac{\|U_\perp^\top Q_{\tau+1}R_{\tau+1}\tilde{\mathbf{v}}_1\|_2^2}{\|U^\top Q_{\tau+1}R_{\tau+1}\tilde{\mathbf{v}}_1\|_2^2 + \|U_\perp^\top Q_{\tau+1}R_{\tau+1}\tilde{\mathbf{v}}_1\|_2^2} \\
&\overset{(ii)}{=} \frac{\|U_\perp^\top F_{\tau+1}Q_\tau\tilde{\mathbf{v}}_1\|_2^2}{\|U^\top F_{\tau+1}Q_\tau\tilde{\mathbf{v}}_1\|_2^2 + \|U_\perp^\top F_{\tau+1}Q_\tau\tilde{\mathbf{v}}_1\|_2^2}.
\end{aligned} \tag{12}$$

where $\tilde{\mathbf{v}}_1 = \frac{R_{\tau+1}^{-1}\mathbf{v}_1}{\|R_{\tau+1}^{-1}\mathbf{v}_1\|_2}$. (i) follows as $Q_{\tau+1}$ is an orthogonal matrix and $[U\ U_\perp]$ form a complete orthogonal basis; (ii) follows by using (10). The existence of $R_{\tau+1}^{-1}$ follows using Lemma 8 along with the fact that $\sigma_k(R_{\tau+1}) = \|R_{\tau+1}\zeta_0\|_2 \geq \|U^\top Q_{\tau+1}R_{\tau+1}\zeta_0\|_2 = \|U^\top F_{\tau+1}Q_\tau\zeta_0\|_2 > 0$, where ζ_0 is the singular vector of $R_{\tau+1}$ corresponding to its smallest singular value, $\sigma_k(R_{\tau+1})$.

Now, using (12) with Lemmas 8, 9 and using the fact that $x/(x+c)$ is an increasing function of x, for all $x > 0$, we get (w.p. $\geq 1 - 2C/T$):

$$\|U_\perp^\top Q_{\tau+1}\|_2^2 \leq \frac{(\sigma^2\|U_\perp^\top Q_\tau\|_2 + \lambda_k^2\epsilon/2)^2}{(\lambda_k^2 + \sigma^2 - \frac{\lambda_k^2\epsilon}{4})^2(1 - \|U_\perp^\top Q_\tau\|_2^2) + (\sigma^2\|U_\perp^\top Q_\tau\|_2 + 0.5\lambda_k^2\epsilon)^2}.$$

Now, assuming $\epsilon \leq \|U_\perp^\top Q_\tau\|_2^2$, using the above equation and by using union bound, we get (w.p. $\geq 1 - 2\tau C/T$):

$$\|U_\perp^\top Q_{\tau+1}\|_2^2 \leq \frac{\gamma^2\|U_\perp^\top Q_\tau\|_2^2}{1 - \|U_\perp^\top Q_\tau\|_2^2 + \gamma^2\|U_\perp^\top Q_\tau\|_2^2}, \tag{13}$$

where $\gamma = \frac{\sigma^2+\lambda_k^2/2}{\sigma^2+3\lambda_k^2/4} < 1$ for $\lambda_k > 0$. Using Lemma 5 along with the above equation, we get (w.p. $\geq 1 - 2\tau C/T$):

$$\|U_\perp^\top Q_{\tau+1}\|_2^2 \leq \gamma^{2\tau} \frac{\|U_\perp^\top Q_0\|_2^2}{1 - \|U_\perp^\top Q_0\|_2^2}.$$

Now, using Lemma 10 we know that $\|U_\perp^\top Q_0\|_2^2$ is at most $1-\Omega(1/(kp))$. Hence, for $T = \Omega(\log(p/\epsilon)/\log(1/\gamma)$, we get: $\|U_\perp^\top Q_T\|_2^2 \leq \epsilon$. Furthermore, we require B (as mentioned in the Theorem) samples per block. Hence, the total sample complexity bound is given by $\Omega(BT)$, concluding the proof. □

The following lemma provides a probabilistic lower bound on the "energy" of U along the updated $S_{\tau+1} = F_{\tau+1}Q_\tau$.

Lemma 8. *Let $\mathcal{X}$, A, B, and T be as defined in Theorem 7. Also, let σ be the variance of noise, $F_{\tau+1} = \frac{1}{B}\sum_{B\tau<t\leq B(\tau+1)} x_t x_t^\top$ and Q_τ be the τ-th iterate of Algorithm 2. Then, $\forall\ \mathbf{v} \in \mathbb{R}^k$ and $\|\mathbf{v}\|_2 = 1$, w.p. $1 - 5C/T$ we have:*

$$\|U^\top F_{\tau+1} Q_\tau \mathbf{v}\|_2 \geq (\lambda_k^2 + \sigma^2 - \frac{\lambda_k^2\epsilon}{4})\sqrt{1 - \|U_\perp^\top Q_\tau\|_2^2}.$$

Our proof of the above lemma uses concentration techniques similar to the ones used to prove Lemma 3 and critically uses "goodness" of the initial Q_0 given by Lemma 10.

The following lemma complements the above lemma by providing a probabilistic upper bound on the "energy" of $U_\perp$ along the updated $S_{\tau+1} = F_{\tau+1}Q_\tau$.

Lemma 9. *Let $\mathcal{X}$, A, B, $F_{\tau+1}$, Q_τ be as defined in Lemma 8. Then, w.p. $1-4C/T$, $\|U_\perp^\top F_{\tau+1} Q_\tau\|_2 \leq \sigma^2\|U_\perp^\top Q_\tau\|_2 + \lambda_k^2\epsilon/2$.*

See Appendix E for a detailed proof.

The following lemma guarantees that the initial guess Q_0 which is a uniformly random subspace of dimension k in a space of ambient dimension p is guaranteed to have $\Omega(1/\sqrt{kp})$ total energy along an arbitrary, fixed subspace span(U).

Lemma 10. *Let $Q_0 \in \mathbb{R}^{p\times k}$ be sampled uniformly at random from the set of all k-dimensional subspaces (see Initialization Steps of Algorithm 2). Then, w.p. at least 0.99: $\sigma_k(U^\top Q_0) \geq C\sqrt{\frac{1}{kp}}$, where $C > 0$ is a global constant.*

Proof. Using Step 2 of Algorithm 2: $H = Q_0 R_0$. Let $\mathbf{v}_k$ be the singular vector of $U^\top Q_0$ corresponding to the smallest singular value. Then,

$$\begin{aligned}\sigma_k(U^\top Q_0) &= \frac{\|U^\top Q_0 R_0 R_0^{-1}\mathbf{v}_k\|_2}{\|R_0^{-1}\mathbf{v}_k\|_2}\|R_0^{-1}\mathbf{v}_k\|_2 \\ &\geq \sigma_k(U^\top Q_0 R_0)\sigma_k(R_0^{-1}). \end{aligned} \tag{14}$$

Now, $\sigma_k(R_0^{-1}) = \frac{1}{\|R_0\|_2} = \frac{1}{\|Q_0 R_0\|_2} = \frac{1}{\|H\|_2}$. Note that $\|H\|_2$ is the spectral norm of a random matrix with i.i.d. Gaussian entries and hence can be easily bounded using standard results. In particular, using Lemma 13, we get: $\|H\|_2 \leq C_1\sqrt{p}$ w.p. $\geq 1 - e^{-C_2 p}$, where $C_1, C_2 > 0$ are global constants.

By Theorem 1.1 of Rudelson & Vershynin (2009) (see Lemma 14), w.p. ≥ 0.99, $\sigma_k(U^\top Q_0 R_0) = \sigma_k(H) \geq C/\sqrt{k}$. The lemma now follows using the above two bounds with (14). □

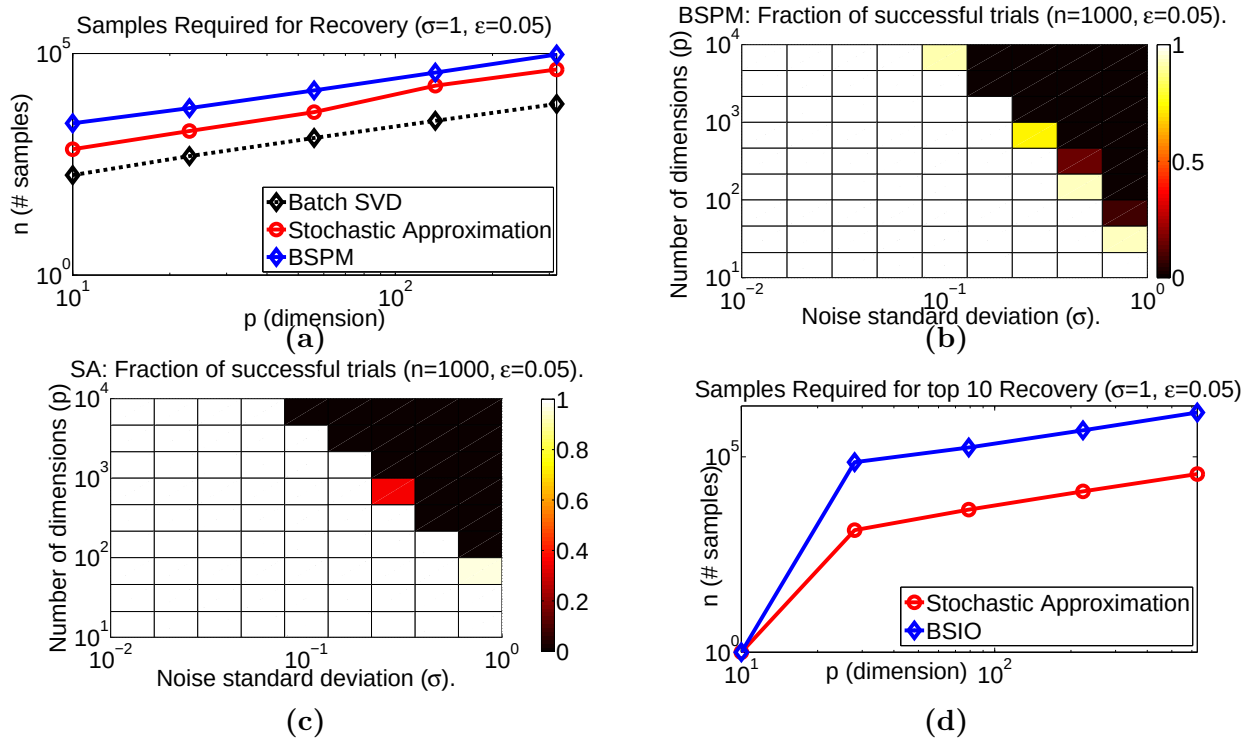

Figure 1: **(a)**: Number of samples required for recovery of the top PC ($k = 1$) with noise standard deviation $\sigma = 1$ and desired accuracy $\epsilon = 0.05$. The figure compares batch SVD, SA and BSPM, **(b)**: Fraction of trials in which BSPM successfully recovers the principal component ($k = 1$) with $\epsilon = 0.05$ and $n = 1000$ samples, **(c)**: Fraction of trials in which SA successfully recovers the top PC ($k = 1$) with $\epsilon = 0.05$ and $n = 1000$, **(d)**: Number of samples required for the recovery of the $k = 10$ top components with $\sigma = 1$, $\epsilon = 0.05$. The figure compares the stochastic approximation and block-stochastic orthogonal iteration algorithms.

5 Empirical Results

It has long been observed empirically that Stochastic Approximation (SA) performs similarly to the Batch-SVD method. Yet no finite-sample analysis of SA is available. In this section, we show that, as predicted by our theoretical results (Sections 4.1 and 4.2), our modified SA-type algorithm, Block-Stochastic Power Method (BSPM), also performs similarly to Batch-SVD (and hence to SA).

We provide phase-transition-type figures to show that for a large fraction of problems, SA and BSPM require a similar number of samples.

In all the experiments, we draw our data from the generative model of (2), and report results averaged over at least 500 independent runs. We provide all algorithms that need random initialization with the exact same random initial estimate. The BSPM algorithm uses the block size prescribed in Theorem 7, with the empirically tuned constant of 0.2. Similarly, for SA we used the popular $\eta_t = C/t$ step size, where $C = 10$ is also empirically tuned.

Rank-1 case: We first study scaling of the number of samples required by different methods as the dimensionality of the data (p) increases. Figure 1 (a) As indicated by our Theorem 1, BSPM exhibits a linear scaling of number of samples with p.

Next, we present phase-transition type of plots to demonstrate recovery properties of both BSPM and SA under a variety of problem settings. Here, we fix number of samples $n = 1000$ while varying the two problem parameters, i.e., p, σ. In all cases we count the fraction of trials in

which each algorithm came up with a solution with accuracy $\epsilon = 0.05$. Figures 1 (b), (c) show the phase-transition behavior exhibited by the two stochastic algorithms; white indicates 100% success rate while black represents 0% rate of success. Note that, here again the behavior of both the methods is similar, with SA being marginally better.

Rank-k case: Finally, to cover the full extent of the cases solved by the analysed algorithms, we also present simulation results for the rank-k case. We used the BSOI algorithm from Section 4.2 and the orthogonalized (via QR decomposition) version of the stochastic approximation algorithm for our simulations. Figure 1 (d) compares the sample complexity of different methods for $k = 10$ in the model of (2). Here again we notice that the curves for BSOI follow the scaling behaviour of the curves for SA and differ just by a small constant.

6 Conclusion

We consider the streaming one-pass-over-the-data model for PCA. Where as the classical PCA algorithm requires forming the covariance matrix, and hence requires storage $O(p^2)$, our algorithm uses only $O(kp)$ storage; this is the best possible memory requirement, since that is needed just to store the k principal components. The computational effort per step is also $O(p)$, like other stochastic analysis approaches. Yet, unlike other covariance-free approaches that have been proposed, we provide finite sample bounds that demonstrate, in particular, that our algorithm exhibits essentially (up to log and constant factors) the same convergence rates as batch PCA. This theoretical agreement is also illustrated in our experiments section.

References

Arora, R., Cotter, A., Livescu, K., and Srebro, N. Stochastic optimization for PCA and PLS. In *50th Allerton Conference on Communication, Control, and Computing*, Monticello, IL, 2012.

Balzano, L., Nowak, R., and Recht, B. Online identification and tracking of subspaces from highly incomplete information. In *Communication, Control, and Computing (Allerton), 2010 48th Annual Allerton Conference on*, pp. 704–711, 2010.

Brand, M. Fast low-rank modifications of the thin singular value decomposition. *Linear algebra and its applications*, 415(1):20–30, 2006.

Brand, Matthew. Incremental singular value decomposition of uncertain data with missing values. *Computer Vision—ECCV 2002*, pp. 707–720, 2002.

Clarkson, Kenneth L. and Woodruff, David P. Numerical linear algebra in the streaming model. In *Proceedings of the 41st annual ACM symposium on Theory of computing*, pp. 205214, 2009.

Comon, P. and Golub, G. H. Tracking a few extreme singular values and vectors in signal processing. *Proceedings of the IEEE*, 78(8):1327–1343, 1990.

He, J., Balzano, L., and Lui, J. Online robust subspace tracking from partial information. *arXiv preprint arXiv:1109.3827*, 2011.

Herbster, Mark and Warmuth, Manfred K. Tracking the best linear predictor. *The Journal of Machine Learning Research*, 1:281–309, 2001.

Li, Y. On incremental and robust subspace learning. *Pattern recognition*, 37(7):1509–1518, 2004.

Oja, E. and Karhunen, J. On stochastic approximation of the eigenvectors and eigenvalues of the expectation of a random matrix. *Journal of mathematical analysis and applications*, 106(1): 69–84, 1985.

Robbins, Herbert and Monro, Sutton. A stochastic approximation method. *The Annals of Mathematical Statistics*, pp. 400–407, 1951.

Roweis, Sam. EM algorithms for PCA and SPCA. *Advances in neural information processing systems*, pp. 626–632, 1998.

Rudelson, Mark and Vershynin, Roman. The Littlewood-Offord Problem and invertibility of random matrices. *ArXiv Mathematics e-prints*, 2007.

Rudelson, Mark and Vershynin, Roman. Smallest singular value of a random rectangular matrix. *Communications on Pure and Applied Mathematics*, 62(12):1707–1739, 2009.

Tipping, Michael E. and Bishop, Christopher M. Probabilistic principal component analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 61(3):611–622, 1999.

Vershynin, R. How close is the sample covariance matrix to the actual covariance matrix? *Journal of Theoretical Probability*, pp. 1–32, 2010a.

Vershynin, Roman. Introduction to the non-asymptotic analysis of random matrices. *arXiv preprint arXiv:1011.3027*, 2010b.

Warmuth, Manfred K. and Kuzmin, Dima. Randomized online PCA algorithms with regret bounds that are logarithmic in the dimension. *Journal of Machine Learning Research*, 9:2287–2320, 2008.

Weng, J., Zhang, Y., and Hwang, W.S. Candid covariance-free incremental principal component analysis. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 25(8):1034–1040, 2003.

Zha, H. and Simon, H. D. On updating problems in latent semantic indexing. *SIAM Journal on Scientific Computing*, 21(2):782–791, 1999.

Zhao, H., Yuen, P. C., and Kwok, J. T. A novel incremental principal component analysis and its application for face recognition. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 36(4):873–886, 2006.

A Preliminaries

Lemma 11 (Lemma 5.4 of Vershynin (2010b)). *Let A be a symmetric $k \times k$ matrix, and let $\mathcal{N}_\epsilon$ be an ϵ-net of S^{k-1} for some $\epsilon \in [0, 1)$. Then,*

$$\|A\|_2 \leq \frac{1}{(1-2\epsilon)} \sup_{\mathbf{x} \in \mathcal{N}_\epsilon} |\langle A\mathbf{x}, \mathbf{x}\rangle|.$$

Lemma 12 (Proposition 2.1 of Vershynin (2010a)). *Consider independent random vectors $\mathbf{x}_1, \ldots, \mathbf{x}_n$ in $\mathbb{R}^p$, $n \geq p$, which have sub-Gaussian distribution with parameter 1. Then for every $\delta > 0$ with probability at least $1 - \delta$ one has,*

$$\|\frac{1}{n}\sum_{i=1}^{n} \mathbf{x}_i\mathbf{x}_i^T - \mathbb{E}[\mathbf{x}_i\mathbf{x}_i^T]\|_2 \leq C\sqrt{\log(2/\delta)}\sqrt{\frac{p}{n}}.$$

Lemma 13 (Corollary 3.5 of Vershynin (2010b)). *Let A be an $N \times n$ matrix whose entries are independent standard normal random variables. Then for every $t \geq 0$, with probability at least $1 - 2\exp(-t^2/2)$ one has,*

$$\sqrt{N} - \sqrt{n} - t \leq \sigma_k(A) \leq \sigma_1(A) \leq \sqrt{N} + \sqrt{n} + t.$$

Lemma 14 (Theorem 1.2 of Rudelson & Vershynin (2007)). *Let $\zeta_1, \ldots, \zeta_n$ be independent centered real random variables with variances at least 1 and subgaussian moments bounded by B. Let A be an $k \times k$ matrix whose rows are independent copies of the random vector $(\zeta_1, \ldots, \zeta_n)$. Then for every $\epsilon \geq 0$ one has*

$$\Pr(\sigma_{min}(A) \leq \epsilon/\sqrt{k}) \leq C\epsilon + c^n,$$

where $C > 0$ and $c \in (0, 1)$ depend only on B. Note that $B = 1$ for the standard Gaussian variables.

Lemma 15. *Let $\mathbf{x}_i \in \mathbb{R}^m, 1 \leq i \leq B$ be i.i.d. standard multivariate normal variables. Also, $\mathbf{y}_i \in \mathbb{R}^n$ are also i.i.d. normal variables and are independent of $\mathbf{x}_i, \forall i$. Then, w.p. $1 - \delta$,*

$$\left\| \frac{1}{B} \sum_i \mathbf{x}_i \mathbf{y}_i^\top \right\|_2 \leq \sqrt{\frac{C \max(m, n) \log(2/\delta)}{B}}.$$

Proof. Let $M = \sum_i \mathbf{x}_i\mathbf{y}_i^T$ and let $m > n$. Then, the goal is to show that, the following holds w.p. $1 - \delta$: $\frac{1}{B}\|M\mathbf{v}\|_2 \leq \sqrt{\frac{Cm\log(2/\delta)}{B}}$ for all $\mathbf{v} \in \mathbb{R}^n$ s.t. $\|\mathbf{v}\|_2 = 1$.

We prove the lemma by first showing that the above mentioned result holds for any *fixed* vector v and then use standard epsilon-net argument to prove it for all $\mathbf{v}$.

Let $\mathcal{N}$ be the 1/4-net of S^{n-1}. Then, using Lemma 5.4 of Vershynin (2010b) (see Lemma 11),

$$\|\frac{1}{Bm}M^TM\|_2 \leq 2\max_{\mathbf{v} \in \mathcal{N}} \frac{1}{Bm}\|M\mathbf{v}\|_2^2. \tag{15}$$

Now, for any fixed $\mathbf{v}$: $M\mathbf{v} = \sum_i \mathbf{x}_i\mathbf{y}_i^T\mathbf{v} = \sum_i \mathbf{x}_i c_i$, where $c_i = \mathbf{y}_i^T\mathbf{v} \sim N(0, 1)$. Hence,

$$\|M\mathbf{v}\|_2^2 = \sum_{\ell=1}^{m}(\sum_{i=1}^{B} x_{i\ell}c_i)^2.$$

Now, $\sum_{i=1}^{B} x_{i\ell}c_i \sim N(0, \|c\|_2^2)$ where $c^T = [c_1\ c_2 \cdots c_B]$. Hence, $\sum_{i=1}^{B} x_{i\ell}c_i = \|c\|_2 h_\ell$ where $h_\ell \sim N(0, 1)$.

Therefore, $\|M\mathbf{v}\|_2^2 = \|c\|_2^2\|h\|_2^2$ where $h^T = [h_1\ h_2 \cdots h_B]$. Now,

$$Pr(\frac{\|c\|_2^2\|h\|_2^2}{Bm} \geq 1+\gamma) \leq Pr(\frac{\|c\|_2^2}{B} \geq \sqrt{1+\gamma}) + Pr(\frac{\|h\|_2^2}{m} \geq \sqrt{1+\gamma})$$
$$\overset{\zeta_1}{\leq} 2\exp(-\frac{B\gamma^2}{32}) + 2\exp(-\frac{m\gamma^2}{32}) \leq 4\exp(-\frac{m\gamma^2}{32}), \quad (16)$$

where $0 < \gamma < 3$ and ζ_1 follows from Lemma 13.

Using (15), (16), the following holds with probability $(1 - 9^{n+1}e^{-\frac{m\gamma^2}{32}})$:

$$\frac{\|M\|_2^2}{Bm} \leq 1 + 2\gamma. \quad (17)$$

The result now follows by setting γ appropriately and assuming $n < Cm$ for small enough C. □

B Proof of Lemma 2

Proof. Note that,

$$\frac{1}{B}\sum_t \mathbf{x}_t\mathbf{x}_t^\top - \mathbf{u}\mathbf{u}^\top - \sigma^2 I = \mathbf{u}\mathbf{u}^\top \frac{1}{B}\sum_t (z_t^2 - 1) +$$
$$\frac{1}{B}\sum_t (\mathbf{w}_t\mathbf{w}_t^\top - \sigma^2 I) + \frac{1}{B}\sum_t z_t\mathbf{w}_t\mathbf{u}^\top + \frac{1}{B}\mathbf{u}\sum_t z_t\mathbf{w}_t^\top. \quad (18)$$

We now individually bound each of the above given terms in the RHS. Using standard tail bounds for covariance estimation (see Lemma 12), we can bound the first two terms (w.p. $1 - 2C/T$):

$$\frac{1}{B}\left|\sum_t (z_t^2 - 1)\right| \leq \sqrt{\frac{C\log(T)}{B}},$$
$$\|\frac{1}{B}\sum_t (\mathbf{w}_t\mathbf{w}_t^\top - \sigma^2 I)\|_2 \leq \sigma^2\sqrt{\frac{C_1 p\log(T)}{B}}. \quad (19)$$

Similarly, using Lemma 15, we can bound the last two terms in (18) (w.p. $1 - 2C/T$):

$$\|\frac{1}{B}\sum_t z_t\mathbf{w}_t\mathbf{u}^\top\|_2 = \|\frac{1}{B}\mathbf{u}\sum_t z_t\mathbf{w}_t^\top\|_2 \leq \sigma\sqrt{\frac{C_1 p\log(T)}{B}}. \quad (20)$$

The lemma now follows by using (18), (19), (20) along with B as given by Theorem 1. □

C Proof of Lemma 3

Proof. Let $\mathbf{q}_\tau = \sqrt{1-\delta_\tau}\mathbf{u} + \sqrt{\delta_\tau}\mathbf{u}_\tau^\perp$, where $\mathbf{u}_\tau^\perp$ is the component of $\mathbf{q}_\tau$ that is orthogonal to $\mathbf{u}$. Now,

$$\mathbf{u}^\top\mathbf{s}_{\tau+1} = \frac{1}{B}\sum_t (\mathbf{u}^\top\mathbf{x}_t)(\mathbf{x}_t^\top\mathbf{q}_t)$$
$$= \frac{1}{B}\sum_t (z_t + \mathbf{u}^\top\mathbf{w}_t)(\sqrt{1-\delta_\tau}(z_t + \mathbf{u}^\top\mathbf{w}_t) + \sqrt{\delta_\tau}\mathbf{w}_t^\top\mathbf{u}_\tau^\perp)$$
$$= \frac{\sqrt{1-\delta_\tau}}{B}\sum_t (z_t + \mathbf{u}^\top\mathbf{w}_t)^2 + \frac{\sqrt{\delta_\tau}}{B}\sum_t (z_t + \mathbf{u}^\top\mathbf{w}_t)\mathbf{w}_t^\top\mathbf{u}_\tau^\perp. \quad (21)$$

Now, the first term above is a summation of B i.i.d. chi-square variables and hence using standard results (see Lemma 13), w.p. $(1 - C/T)$:

$$\frac{1}{B}\sum_t (z_t + \mathbf{u}^\top \mathbf{w}_t)^2 \geq (1+\sigma^2)(1 - \sqrt{\frac{C\log(2T)}{B}}). \tag{22}$$

Also, $\mathbf{w}_t^\top \mathbf{u}$ and $\mathbf{w}_t^\top \mathbf{u}_\tau^\perp$ are independent random variables, as both $\mathbf{w}_t^\top \mathbf{u}$, $\mathbf{w}_t^\top \mathbf{u}_\tau^\perp$ are Gaussians and $E[\mathbf{w}_t^\top \mathbf{u}_\tau^\perp \mathbf{u}^\top \mathbf{w}_t] = 0$. Hence, using Lemma 15, the following holds with probability $\geq 1 - 4C/T$:

$$\|\frac{1}{B}\sum_t (z_t + \mathbf{u}^\top \mathbf{w}_t)\mathbf{w}_t^\top \mathbf{u}_\tau^\perp\|_2 \leq \sigma\sqrt{1+\sigma^2}\sqrt{\frac{C\log(T)}{B}} \overset{(i)}{\leq} \sigma\sqrt{1+\sigma^2}\sqrt{\frac{C_1 p\log(T)}{B(1-\delta_0)}}\sqrt{1-\delta_\tau}, \tag{23}$$

where (i) follows by using inductive hypothesis (i.e., $\sqrt{1-\delta_\tau} > \sqrt{1-\delta_{\tau-1}}$, induction step follows as we show that the error decreases at each step) and Lemma 4.

The lemma now follows by using (21), (22), (23) and by setting B, T appropriately. □

D Proof of Lemma 8

Proof. Using the generative model (2), we get:

$$U^\top F_{\tau+1} Q_\tau \mathbf{v} = \Lambda\left(\frac{1}{B}\sum_t \mathbf{z}_t\mathbf{z}_t^\top\right)\Lambda U^\top Q_\tau \mathbf{v} + \left(\frac{1}{B}\sum_t U^\top \mathbf{w}_t\mathbf{w}_t^\top U\right)U^\top Q_\tau \mathbf{v}$$
$$+(\frac{1}{B}\sum_t U^\top \mathbf{w}_t\mathbf{z}_t^\top)\Lambda U^\top Q_\tau \mathbf{v} + \Lambda(\frac{1}{B}\sum_t \mathbf{z}_t\mathbf{w}_t^\top U)U^\top Q_\tau \mathbf{v} + \left(\frac{1}{B}\sum_t (\Lambda\mathbf{z}_t + U^\top \mathbf{w}_t)\mathbf{w}_t^\top U_\perp U_\perp^\top Q_\tau\right)\mathbf{v}. \tag{24}$$

Note that in the equation and rest of the proof, t varies from $B\tau < t \leq B(\tau+1)$.

We now show that each of the five terms in the above given equation concentrate around their respective means. Also, let $\mathbf{y}_t = U^\top \mathbf{w}_t$ and $\mathbf{y}_t^\perp = U_\perp^\top \mathbf{w}_t$. Note that, $\mathbf{y}_t \sim N(0, \sigma^2 I_{k\times k})$ and $\mathbf{y}_t^\perp \sim N(0, \sigma^2 I_{(p-k)\times(p-k)})$.

(a): Consider the first term in (24). Using $\|A\mathbf{v}\|_2 \leq \|A\|_2\|\mathbf{v}\|_2$ and the assumption that $\lambda_1 = 1$, we get: $\|\Lambda\left(\frac{1}{B}\sum_t \mathbf{z}_t\mathbf{z}_t^\top - I\right)\Lambda U^\top Q_\tau \mathbf{v}\|_2 \leq \|\left(\frac{1}{B}\sum_t \mathbf{z}_t\mathbf{z}_t^\top - I\right)\|_2 \|U^\top Q_\tau \mathbf{v}\|_2$. Using Lemma 12 we get (w.p. $1 - C/T$):

$$\|\frac{1}{B}\sum_t \mathbf{z}_t\mathbf{z}_t^\top - I\|_2 \leq \sqrt{\frac{C_1 k\log(T)}{B}}.$$

That is,

$$\|\Lambda(\frac{1}{B}\sum_t \mathbf{z}_t\mathbf{z}_t^\top - I)\Lambda U^\top Q_\tau \mathbf{v}\|_2 \leq \sqrt{\frac{C_1 k\log(T)}{B}}\|U^\top Q_\tau \mathbf{v}\|_2. \tag{25}$$

(b): Similarly, the second term in (24) can be bounded as (w.p. $1 - C/T$):

$$\|\left(\frac{1}{B}\sum_t U^\top \mathbf{w}_t\mathbf{w}_t^\top U - \sigma^2 I\right)U^\top Q_\tau \mathbf{v}\|_2 \leq \sigma^2\sqrt{\frac{C_1 k\log(T)}{B}}\|U^\top Q_\tau \mathbf{v}\|_2. \tag{26}$$

(c): Now consider the third and the fourth term. Now $\mathbf{w}_t$ and $\mathbf{z}_t$ are independent 0-mean Gaussians, hence using Lemma 15, we get: $\|\frac{1}{B}\sum_t U^\top \mathbf{w}_t \mathbf{z}_t^\top\|_2 \leq \sigma\sqrt{\frac{C_1 k\log(T)}{B}}$. Hence, w.p. $1-2C/T$,

$$\|\Lambda(\frac{1}{B}\sum_t \mathbf{z}_t\mathbf{w}_t^\top U)U^\top Q_\tau \mathbf{v}\| + \|(\frac{1}{B}\sum_t U^\top \mathbf{w}_t\mathbf{z}_t)\Lambda U^\top Q_\tau \mathbf{v}\| \leq 2\sigma\sqrt{\frac{C_1 k\log(T)}{B}}\|U^\top Q_\tau \mathbf{v}\|_2. \tag{27}$$

(d): Finally, we consider the last term in (24). Note that, $(\Lambda\mathbf{z}_t + U^\top \mathbf{w}_t) \sim N(0, D)$ where D is a diagonal matrix with $D_{ii} = \lambda_i^2 + \sigma^2$. Also, $Q^\top U_\perp U_\perp^\top \mathbf{w}_t \sim N(0, \sigma^2 I_{(p-k)\times(p-k)})$ and is independent of $(\Lambda\mathbf{z}_t + U^\top \mathbf{w}_t)$ as $E[Q^\top U_\perp U_\perp^\top \mathbf{w}_t \mathbf{w}_t^\top U] = 0$; recall that for Gaussian RVs, covariance is zero iff RVs are independent. Hence, using Lemma 15, w.p. $\geq 1 - C/T$:

$$\|(\frac{1}{B}\sum_t (\Lambda\mathbf{z}_t + U^\top \mathbf{w}_t)\mathbf{w}_t^\top U_\perp U_\perp^\top Q_\tau)\mathbf{v}\|_2 \leq \sqrt{1+\sigma^2}\sigma\sqrt{\frac{C_1 k\log(T)}{B}}. \tag{28}$$

Now, using (24), (25), (26), (27), (28) (w.p. $\geq 1 - 5C/T$)

$$\|U^\top F_{\tau+1} Q_\tau \mathbf{v}\|_2 \geq \|(\Lambda^2+\sigma^2 I)U^\top Q_\tau \mathbf{v}\|_2 - \sqrt{\frac{C_1 k\log(T)}{B}}\|U^\top Q_\tau \mathbf{v}\|_2 \left((1+\sigma)^2 + \frac{\sigma\sqrt{1+\sigma^2}}{\|U^\top Q_\tau \mathbf{v}\|_2}\right). \tag{29}$$

Now, $\|U^\top Q_\tau \mathbf{v}\|_2 \geq \sigma_k(U^\top Q_\tau \mathbf{v})$. Next, by using the inductive hypothesis (i.e., $\sigma_k(U^T Q_\tau) \geq \sigma_k(U^T Q_{\tau-1})$, induction step follows as we show that the error decreases at each step) and Lemma 10, we have $\|U^\top Q_\tau \mathbf{v}\|_2 \geq \sigma_k(U^\top Q_0) \geq \frac{C}{\sqrt{pk}}$ with probability ≥ 0.99.

Also, $\|(\Lambda^2+\sigma^2 I)U^\top Q_\tau \mathbf{v}\|_2 \geq (\lambda_k^2+\sigma^2)\|U^\top Q_\tau \mathbf{v}\|_2$. Additionally, $\|U^\top Q_\tau \mathbf{v}\|_2 \geq \sqrt{1 - \|U_\perp^\top Q_\tau\|_2^2}$. Hence, lemma follows by using these facts with (29) and by selecting B as given in Theorem 7. □

E Proof of Lemma 9

Proof. Similar to our proof for Lemma 8, we separate out the "error" or deviation terms in $\|U_\perp^\top F_{\tau+1} Q_\tau\|_2$ and bound them using concentration bounds. Now,

$$\begin{aligned}
\|U_\perp^\top F_{\tau+1} Q_\tau \mathbf{v}\|_2 &= \|U_\perp^\top (U\Lambda^2 U^\top + \sigma^2 I + E_\tau) Q_\tau \mathbf{v}\|_2 \\
&\leq \|\sigma^2 U_\perp^\top Q_\tau \mathbf{v}\|_2 + \|U_\perp^\top E_\tau Q_\tau \mathbf{v}\|_2 \\
&\leq \sigma^2 \|U_\perp^\top Q_\tau \mathbf{v}\|_2 + \|E_\tau\|_2,
\end{aligned} \tag{30}$$

where E_τ is the error matrix representing deviation of the estimate $F_{\tau+1}$ from its mean. That is,

$$\begin{aligned}
E &= \frac{1}{B}\sum_t \mathbf{x}_t\mathbf{x}_t^\top - U\Lambda^2 U^\top - \sigma^2 I \\
&= U\Lambda(\frac{1}{B}\sum_t \mathbf{z}_t\mathbf{z}_t^\top - I)\Lambda U^\top + (\frac{1}{B}\sum_t \mathbf{w}_t\mathbf{w}_t^\top - \sigma^2 I) \\
&+ U\Lambda\frac{1}{B}\sum_t \mathbf{z}_t\mathbf{w}_t^\top + \frac{1}{B}\sum_t \mathbf{w}_t\mathbf{z}_t^\top \Lambda U.
\end{aligned} \tag{31}$$

Note that the above given four terms correspond to similar four terms in (24) and hence can be bounded in similar fashion. In particular, the following holds with probability $1 - 4C/T$:

$$\|E\|_2 \leq \sqrt{\frac{C_1 k\log(T)}{B}} + \sigma^2\sqrt{\frac{C_1 p\log(T)}{B}} + 2\sigma\sqrt{\frac{C_1 p\log(T)}{B}} \leq \lambda_k^2\epsilon/2, \tag{32}$$

where the second inequality follows by setting B as required by Theorem 7. The lemma now follows using (30), (31), (32). □