# Spectral Guarantees for Adversarial Streaming PCA

Eric Price
UT Austin

Zhiyang Xun
UT Austin

August 21, 2024

**Abstract**

In streaming PCA, we see a stream of vectors $x_1, \dots, x_n \in \mathbb{R}^d$ and want to estimate the top eigenvector of their covariance matrix. This is easier if the spectral ratio $R = \lambda_1/\lambda_2$ is large. We ask: how large does $R$ need to be to solve streaming PCA in $\widetilde{O}(d)$ space? Existing algorithms require $R = \widetilde{\Omega}(d)$. We show:

- For all mergeable summaries, $R = \widetilde{\Omega}(\sqrt{d})$ is necessary.
- In the insertion-only model, a variant of Oja's algorithm gets $o(1)$ error for $R = O(\log n \log d)$.
- No algorithm with $o(d^2)$ space gets $o(1)$ error for $R = O(1)$.

Our analysis is the first application of Oja's algorithm to adversarial streams. It is also the first algorithm for adversarial streaming PCA that is designed for a *spectral*, rather than *Frobenius*, bound on the tail; and the bound it needs is exponentially better than is possible by adapting a Frobenius guarantee.

# 1 Introduction

Principal Component Analysis (PCA) is a fundamental primitive for handling high-dimensional data by finding the highest-variance directions. At its most simple, given a data set $X \in \mathbb{R}^{n \times d}$ of $n$ data points in $d$ dimensions, we want to find the top unit eigenvector $v^*$ of the covariance matrix $\Sigma = \frac{1}{n} X^T X$.

One common way to approximate $v^*$ is the power method: start with a random vector $u_0$, then repeatedly multiply by $\Sigma$ and renormalize. This converges to $v^*$ at a rate that depends on the ratio of the top two eigenvalues of $\Sigma$, denoted $R := \lambda_1/\lambda_2$. In particular, after $O(\log_R \frac{d}{\varepsilon})$ iterations we have $\|Pu_k\|^2 = 1 - \langle u_k, v^* \rangle^2 = \sin^2(u_k, v^*) \leq \varepsilon$ with high probability, where $P = I - v^*(v^*)^T$ projects away from $v^*$.

But what if the data points $x_1, x_2, \ldots, x_n \in \mathbb{R}^d$ arrive in a streaming fashion? Directly applying the power method requires either $nd$ space to store $X$, or $d^2$ space to store $\Sigma$. What can be done in smaller space? The question of streaming PCA has been extensively studied, in two main settings: *adversarial* and *stochastic* streams.

In the adversarial streaming setting, we want to solve PCA for an arbitrary set of data points in arbitrary order. Many of these algorithms store linear sketches of the data, such as $AX$ and $XB$ for Gaussian matrices $A, B$ [CW09; BWZ16; Woo14b; Upa18; TYUC17]. These results give a Frobenius guarantee for rank-$k$ approximation of $X$. Specialized to $k = 1$, the result direction $\widehat{u}$ satisfies

$$\left\|X(I - \widehat{u}\widehat{u}^T)\right\|_F^2 \leq (1 + \varepsilon) \|XP\|_F^2$$

which is equivalent to

$$\widehat{u}^T \Sigma \widehat{u} \geq \lambda_1 - \varepsilon \sum_{i>1} \lambda_i.$$

The best result here is FrequentDirections [Lib13; GLPW16], which is a deterministic insertion-only algorithm rather than a linear sketch. It uses $O(d/\varepsilon)$ space to get the guarantee, which is optimal [CW09]. Unfortunately, this Frobenius guarantee can be quite weak: if the eigenvalues do not decay and we only have a bound on $R = \lambda_1/\lambda_2$, to get $\|P\widehat{u}\|^2 \leq 0.1$ we need $\varepsilon < \frac{R}{d}$, which means $\Theta(d^2/R)$ space. The well-known spiked covariance mode [Joh01], where the $x_i$ are iid Gaussian with covariance that has eigenvalues $\lambda_2 = \lambda_3 = \cdots = \lambda_d$, is one example where this quadratic space bound appears.

In the stochastic streaming setting, the $x_i$ are drawn iid from a somewhat nice distribution. The goal is to converge to the principal component of the true distribution using little space and few samples. Algorithms for the stochastic setting are typically iterative, using $O(d)$ space and converging to the true solution with a sample complexity depending on how "nice" the distribution is. Examples include Oja's algorithm [Oja82; BDF13; JJKNS16; AL17; HNTW21; HNW21; LSW21] and the block power method [ACLS12; MCJ13; HP14; BDWY16]. Oja's algorithm starts with a random $v_0$, then repeatedly sets

$$v_i = v_{i-1} + \eta_i x_i x_i^T v_{i-1}$$

for some small learning rate $\eta_i$. These analyses depend heavily on the data points being iid[1]. In return, they can get a stronger *spectral* guarantee than the sketching algorithms. The bounds are not directly comparable to the sketching algorithms (not only does the sample complexity depend on the data distribution, but the convergence is to the principal component of the true distribution

[1] Or nearly so; for example, [JJKNS16] requires that the $x_i$ are independent with identical covariance matrices.

rather than the empirical $\Sigma$), but in the spiked covariance setting they just need $n \geq \widetilde{O}((1+\frac{1}{R-1})^2 d)$ rather than $O(d^2/R)$. That is, they use near-linear samples down to $R = 1 + \varepsilon$.

So the situation is: algorithms that handle arbitrary data need $O(d^2/R)$ space for a spectral guarantee. Iterative methods have a good spectral guarantee—linear space and often near-linear samples for constant $R$—but only handle iid data. Is this separation necessary, or can we get a good spectral guarantee in the arbitrary-data setting? In this paper we ask:

*Is a polynomial spectral gap necessary to guarantee a near-linear space algorithm?*

### 1.1 Our results

Our main result is that linear space *is* possible for polylogarithmic spectral gaps. In fact, Oja's method essentially achieves this:

**Theorem 1.1** (Performance of Oja's method in adversarial streams)**.** *For any sufficiently large universal constant $C$, suppose $\eta$ is such that $\eta n \lambda_1 > C \log d$ and $\eta n \lambda_2 < \frac{1}{C \log n}$. If $\eta \|x_i\|^2 \leq 1$ for every $i$, then Oja's algorithm with learning rate $\eta$ returns $\widehat{v}$ satisfying $\|P\widehat{v}\| \leq \sqrt{\eta n \lambda_2} + d^{-9}$ with $1 - d^{-\Omega(C)}$ probability.*

*Moreover, Oja's method can be modified (Algorithm 1) so that in addition, regardless of $\lambda_1$ and $\lambda_2$, if $\eta \|x_i\|^2 \leq 1$ for all $i$ then either $\|P\widehat{v}\| \leq \sqrt{\eta n \lambda_2} + d^{-9}$ or $\widehat{v} = \perp$.*

If $R > O(\log n \log d)$, there exists an $\eta$ that satisfies the eigenvalue condition. However, Theorem 1.1 requires knowing $\eta$ and that no single $\|x_i\|$ is too large. It's fairly easy to extend the algorithm to remove both restrictions, as well as describe the performance with respect to finite precision. Algorithm 2 simply runs Oja's method for different learning rates and picks the smallest one that works; unless any single $x_i$ has too large $\|x_i\|^2$ violating Theorem 1.1, in which case it outputs that $x_i$. For $X \in \mathbb{R}^{n \times d}$ with $b$-bits entries, where each $X_{i,j}$ is either 0 or falls within $2^{-b} \leq |X_{i,j}| \leq 2^b$, it suffices to test roughly $O(b)$ different learning rates in parallel. We say an algorithm $\varepsilon$-approximates PCA if it returns $u$ with $\|Pu\|^2 \leq \varepsilon$, and we have the following theorem.

**Theorem 1.2** (Full algorithm)**.** *For $X \in \mathbb{R}^{n \times d}$ have $b$-bit entries for $b > \log(dn)$. Whenever the spectral gap $R = \lambda_1/\lambda_2 > O(\log n \log d)$, Algorithm 2 uses $O(b^2 d)$ bits of space and $O(\frac{\log d}{R} + d^{-9})$-approximates PCA with high probability.*

**Algorithm 1** Oja's Algorithm, checking the growth of $\|v_n\|$ to identify too-small learning rates.

**procedure** OjaCheckingGrowth($X$, $\eta$)
  Choose $\widehat{v}_0 \in S^{d-1}$ uniformly. ▷ All numbers stored to $O(\log(nd))$ bits of precision
  Set $s_0 = 0$.
  **for** $i = 1, \ldots, n$ **do**
    $v_i' \leftarrow (1 + \eta x_i x_i^T)\widehat{v}_{i-1}$.
    $\widehat{v}_i \leftarrow \frac{v_i'}{\|v_i'\|}$.
    $s_i \leftarrow s_{i-1} + \log \|v_i'\|$.
  **end for**
  **if** $s_n \leq 10 \log d$, **return** $\perp$. ▷ Returns $\perp$ rather than a wrong answer if $\eta$ is too small.
  **else return** $\widehat{v}_n$
**end procedure**

**Algorithm 2** Algorithm handling unknown learning rate and large-norm entries

**procedure** AdversarialPCA($X$, $b$) ▷ $X \in \mathbb{R}^{n\times d}$ has $X_{i,j} = 0$ or $2^{-b} \le |X_{i,j}| \le 2^b$
  Define $\eta_i = 2^i$ for integer $i$, $|i| \le 4b + \log(nd^2) + O(1)$.
  In parallel run OjaCheckingGrowth($X, \eta_i$) for all $i$, getting $v^{(i)}$.
  In parallel record $\overline{x}$, the single $x_i$ of maximum $\|x_i\|$.
  Let $i^*$ be the smallest $i$ with $v^{(i)} \neq \perp$.
  **if** $\eta_{i^*} \|\overline{x}\|_2 \ge 1$, **return** $\frac{\overline{x}}{\|\overline{x}\|}$.
  **else return** $v^{(i^*)}$.
**end procedure**

**Lower bound for mergeable summaries.** Existing algorithms for adversarial PCA, including linear sketching or FrequentDirections, fall under the category of mergeable summaries [ACHPWY13]. These algorithms enable processing of disjoint data inputs on separate machines, producing summaries that can be combined to address the problem using the full dataset. By contrast, our algorithm is not mergeable and requires the data to appear in one long sequence.

Considering the benefits of mergeable summaries, a natural goal would be to get a good spectral guarantee with a mergeable summary. As discussed above, existing algorithms require $\Omega(d^2/R)$ space, so $R = \widetilde{\Theta}(d)$ is needed for them to achieve near-linear space. Is it possible to get near-linear space and logarithmic $R$, like Theorem 1.2 achieves in the insertion-only model?

Existing lower bounds [LW16] imply that $\Omega(d^2/R^2)$ space is necessary for linear sketching (see Appendix B for discussion). We show that the same bound applies to *all* mergeable algorithms: all mergeable summaries require $\Omega(d^2/R^2)$ bits of space to 0.1-approximate PCA, making $R = \widetilde{\Omega}(\sqrt{d})$ necessary for near-linear space.

**Theorem 1.3** (Mergeable Lower Bound)**.** *For all mergeable summaries, 0.1-approximate PCA on streams with spectral gap $R$ requires at least $\Omega(d^2/R^2)$ bits of space.*

**Dependence on Accuracy.** Theorem 1.2 shows that it is possible to solve $O(\frac{\log d}{R})$-approximate PCA in near-linear space. This is $o(1)$, but cannot be driven towards 0 in the way that other settings allow (in the iid setting, the accuracy improves exponentially in the number of samples; in the existing $O(d^2/R)$-space worst-case algorithms, the space grows as $\frac{1}{\varepsilon}$ for accuracy $\varepsilon$). Unfortunately, we show that this is inherent: there is a phase transition where aiming for more than $\text{poly}(1/R)$ accuracy requires quadratic rather than near-linear space.

**Theorem 1.4** (Accuracy Lower Bound)**.** *There exists a universal constant $C > 1$ such that: for any $R > 1$, $\frac{1}{CR^2}$-approximate PCA on streams with spectral gap $R$ requires at least $\frac{d^2}{CR^3}$ bits of space for sufficiently large $d > \text{poly}(R)$.*

Specializing to constant $R$ gives the following corollary:

**Theorem 1.5.** *For any constant $R > 1$, there exists a constant $\varepsilon > 0$ such that $\varepsilon$-approximate PCA on streams of spectral gap $R$ requires $\Omega(d^2)$ bits of space.*

This shows that for constant $R$, storing the entire covariance matrix is essentially the only thing one can do to achieve $o(1)$ accuracy. By contrast, Theorem 1.2 shows that for $R = \Theta(\log n \log d)$, $\varepsilon$-solving PCA for any constant $\varepsilon > 0$ is possible in $\widetilde{O}(d)$ bits of space. This is a much lower threshold than the $R = \widetilde{\Theta}(d)$ needed for near-linear space by existing analyses.

Our results are summarized in Table 1, which gives upper and lower bounds for the requirements for near-linear space.

| Setting | Method | Mergeable? | Requirement for $\widetilde{O}(d)$ space | Citation |
|---|---|---|---|---|
| Distributional | Oja's algorithm | No | $\lambda_1 > \lambda_2$ | [Oja82] |
| | Block power method | No | $\lambda_1 > \lambda_2$ | [HP14; BDWY16] |
| Adversarial | Linear Sketching | Yes | $\lambda_1 > (\lambda_2 + \ldots + \lambda_d) \cdot \Omega(\frac{1}{\log^C d})$ | [Upa18; TYUC17] |
| | FrequentDirections | Yes | $\lambda_1 > (\lambda_2 + \ldots + \lambda_d) \cdot \Omega(\frac{1}{\log^C d})$ | [Lib13; GLPW16] |
| | Algorithm 2 | No | $\lambda_1 > \lambda_2 \cdot O(\log d \log n)$ | Theorem 1.2 |
| Adversarial | Impossibility | Yes | $\lambda_1 \leq \lambda_2 \cdot d^{0.49}$ | Theorem 1.3 |
| | | No | $\lambda_1 \leq \lambda_2 \cdot 100$ | Theorem 1.5 |

Table 1: In various settings, the requirement on the eigenvectors $\lambda_i$ of the covariance matrix for the algorithm to get small constant approximate PCA in $\widetilde{O}(d)$ space. In the last two rows, we instead state a setting of $\lambda_1, \lambda_2$ for an instance in which $\widetilde{O}(d)$ space is impossible.

### 1.2 Related Work

Oja's algorithm has been extensively studied in the stochastic setting where the data streams are sampled iid; see, e.g., [BDF13; JJKNS16; AL17; HNTW21; HNW21; LSW21]. Since the goal in this setting is to approximate the underlying distribution's principal components, there is a minimum sample complexity for even an offline algorithm to estimate the principal component. This line of work [JJKNS16] can show that Oja's algorithm has a similar sample complexity to the optimal offline algorithm, even for spectral ratios $R$ close to 1. Recent work of [KS24] extends Oja's algorithm to data sampled from a Markov chain instead of iid samples. They showed that, despite the data dependency inherent in Markovian data, the performance of Oja's algorithm is as good as the iid case when the Markov chain has large second eigenvalue.

Our analysis of Oja's algorithm is by necessity quite different from these stochastic-setting analyses. Oja's algorithm returns $v_n = B_n v_0$ for a transformation matrix $B_n = (I + \eta_n x_n x_n^T)(I + \eta_{n-1} x_{n-1} x_{n-1}^T) \cdots (I + \eta_1 x_1 x_1^T)$. In the stochastic setting, $B_n$ is a random variable, with $\mathbb{E}[B_i \mid B_{i-1}] = (I + \eta\Sigma)B_i$; the analyses focus on matrix concentration of $B_n$, essentially to bound the deviation of $B_n$ around the "expected" $(I+\eta\Sigma)^n$. In our arbitrary-data setting, $B_n$ is not a random variable at all. The only randomness is the initialization $v_0$. This makes our analysis quite different, instead tracking how much $\widehat{v}_i$ can move under the covariance constraints.

Our lower bound construction for high accuracy is closely related to one in [Woo14a], which shows an $\Omega(dk/\varepsilon)$ lower bound for a $(1+\varepsilon)$-approximate rank-$k$ approximation of $\Sigma$ in Frobenius norm. The [Woo14a] construction for $k = 1$ and $\varepsilon = \Theta(\frac{1}{n})$ is very similar to ours, and would give an $\Omega(d^2)$ lower bound for a small constant approximation when $R < 2$. Our construction has a more careful analysis in terms of $R$.

Much of the prior work on streaming PCA, for both the adversarial and stochastic settings, is focused on solving $k$-PCA not just the single top direction. We leave the extension of our upper bound to general $k$ as an open question.

## 2 Proof Overview

### 2.1 Upper Bound

For our application of Oja's algorithm we use a fixed learning rate $\eta$ throughout the stream. The $x_i$ correlated with $v^*$ could all arrive at the beginning or the end of the stream, and we want to weight them equally so that at least we can solve the commutative case where Oja's algorithm is

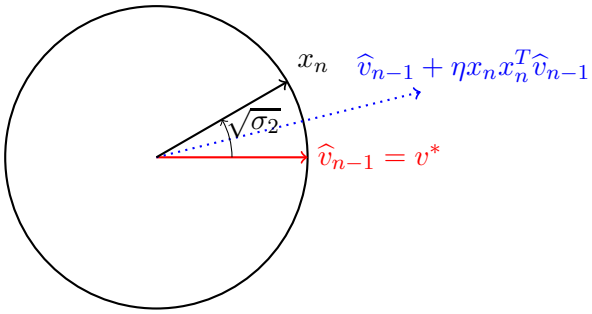


Figure 1: Suppose $\eta = 1$. Then even after convergence to $v^*$ exactly, a single final sample can skew the result by $\Theta(\sqrt{\sigma_2})$. For smaller $\eta$, the same can happen with $\frac{1}{\eta}$ final samples.

relatively simple.

As a basic intuition, Oja's algorithm returns $\widehat{v}_n = \frac{v_n}{\|v_n\|}$, where

$$\begin{aligned} v_n &= (I + \eta x_n x_n^T)(I + \eta x_{n-1} x_{n-1}^T) \cdots (I + \eta x_1 x_1^T) v_0 \\ &\approx e^{\eta x_n x_n^T} e^{\eta x_{n-1} x_{n-1}^T} \cdots e^{\eta x_1 x_1^T} v_0 \end{aligned}$$

where the approximation is good when $\eta \|x_i\|^2 \ll 1$. Imagine that these matrix exponentials commute (e.g., each $x_i$ is $e_j$ for some $j$). Then we would have

$$v_n \approx e^{\eta X^T X} v_0. \tag{1}$$

This suggests that the important property of the learning rate $\eta$ is the spectrum of $\eta X^T X$. Let $\eta X^T X$ have top eigenvalue $\sigma_1 = n\eta\lambda_1$, with corresponding eigenvector $v^*$, and all other eigenvalues at most $\sigma_2 = n\eta\lambda_2$. For Theorem 1.1, we would like to show that Oja's algorithm works if $\sigma_1 > O(\log d)$ and $\sigma_2 < \frac{1}{O(\log n)}$.

For (1) to converge to $v^*$, as in the power method, we want the $v^*$ coefficient of $v_0$ to grow by a poly$(d)$ factor more than any other eigenvalue, i.e., $e^{\sigma_1} \geq \text{poly}(d) e^{\sigma_2}$ or $\sigma_1 \geq \sigma_2 + O(\log d)$. So we certainly need to set $\eta$ such that $\sigma_1 \geq O(\log d)$. But how large a spectral gap do we need, i.e., how small does $\sigma_2$ need to be?

One big concern for adversarial-order Oja's algorithm is: even if most of the stream clearly emphasizes $v^*$ so $v_i$ converges to it, a small number of inputs at the end could cause $v_n$ to veer away from $v^*$ to a completely wrong direction. This cannot happen in the commutative setting, but it can happen in general: $v_n$ can rotate by $\Theta(\sqrt{\sigma_2})$, by ending the stream with $\frac{1}{\eta}$ copies of $v^* + \sqrt{\sigma_2} v'$ (see Figure 1). But this is the worst that can happen. We show:

**Lemma 2.1** (Growth implies correctness)**.** *For any $v_0$ and all $i$, $\|P\widehat{v}_i\| \leq \sqrt{\sigma_2} + \frac{\|Pv_0\|}{\|v_i\|}$.*

This lemma has two useful implications: first, if we ever get close to $v^*$, the final solution will be at most $\sqrt{\sigma_2}$ further from $v^*$. Second, no matter where we start, the final output is good if $\|v_n\|$ is very large. This is how Algorithm 1 can return either a correct answer or $\perp$: it just observes whether $\|v_n\|$ has grown by poly$(d)$.

So it suffices to show that $\|v_n\|$ is large for a random $v_0$; and since $v_0$ starts with a random $\frac{1}{\text{poly}(d)}$ component in the $v^*$ direction, it in fact suffices to show that $\|v_n\|$ would grow by poly$(d)$ if Oja's algorithm started at $v_0 = v^*$. Now, one can show that

$$\|v_n\|^2 \geq e^{\sum_{i=1}^n \eta \langle x_i, \widehat{v}_{i-1} \rangle^2}. \tag{2}$$

So if $v_i$ were always exactly $v^*$, we would have $\|v_n\|^2 \geq e^{\eta(v^*)^T X^T X v^*} = e^{\sigma_1} \geq \text{poly}(d)$ as needed. In addition, if we start at $v^*$, then Lemma 2.1 implies $\|P\widehat{v}_i\| \leq \sqrt{\sigma_2}$ for all $i$, so we never deviate *much* from $v^*$. However, $v_i$ can deviate a little bit, which could decrease $\langle x_i, \widehat{v}_{i-1}\rangle^2$. The question is, by how much? Well, it's easy to show

$$\eta\langle x_i, \widehat{v}_{i-1}\rangle^2 \geq \eta\frac{1-\sigma_2}{2}\langle x_i, v^*\rangle^2 - \eta\langle x_i, P\widehat{v}_{i-1}\rangle^2 \tag{3}$$

so we just need to show that

$$\eta\sum_i \langle x_i, P\widehat{v}_{i-1}\rangle^2 \ll \sigma_1. \tag{4}$$

We know that $\|P\widehat{v}_{i-1}\|^2 \leq \sigma_2$, and $\eta\sum_i\langle x_i, w\rangle^2 \leq \sigma_2$ for any fixed unit vector $w \perp v^*$, but the worry is that $P\widehat{v}_{i-1}$ could rotate through many different orthogonal directions; each direction $w$ can only contribute $\sigma_2^2$ to $\eta\sum_i\langle x_i, P\widehat{v}_{i-1}\rangle^2$, but the total could conceivably be up to $\sigma_2^2 d$.

Our main technical challenge is to rule this out, so $\eta\sum_i\langle x_i, P\widehat{v}_{i-1}\rangle^2$ is small. For intuition, in this overview we just rule out $P\widehat{v}_{i-1}$ moving through many *standard* basis vectors by showing

$$\sum_{j=1}^{d}\max_i\langle e_j, P\widehat{v}_{i-1}\rangle^2 \lesssim \sigma_2\log^2 n\log\|v_n\|. \tag{5}$$

That is, $P\widehat{v}_{i-1}$ cannot rotate through $\sqrt{\sigma_2}$ correlation with each of the $d$ different basis vectors (which would give a value of $\sigma_2 d$) unless $\|v_n\|$ is large (which is what we wanted to show in the first place).

First, we show that $\|v_n\|$ grows proportional to the *squared* movement of $P\widehat{v}_i$:

**Lemma 2.2.** *Suppose $Pv_0 = 0$. For any two time steps $0 \leq a < b \leq n$,*

$$\|P\widehat{v}_b - P\widehat{v}_a\|^2 \leq 4\sigma_2\log\frac{\|v_b\|}{\|v_a\|}$$

As a result, for any subsequence $i_0, \ldots, i_k$ of iterations, the sum of squared movement has

$$\sum_{j=1}^{k}\left\|P\widehat{v}_{i_j} - P\widehat{v}_{i_{j-1}}\right\|^2 \lesssim \sigma_2\log\|v_n\|.$$

We use a combinatorial lemma to turn this bound on squared distances over subsequences into (5). For any set of vectors $A$ the following holds (see Figure 2):

**Lemma 2.3** (Simplified version of Lemma 3.2)**.** *Let $A_0 = 0$, and $A_1, \ldots, A_n \in \mathbb{R}^d$ satisfy that every subsequence $S$ of $\{0, \ldots, n\}$ has*

$$\sum_i \left\|A_{S_i} - A_{S_{i-1}}\right\|_2^2 \leq B.$$

*for some $B > 0$. Then*

$$\sum_{j=1}^{d}\max_{i\in[n]}(A_i)_j^2 \leq B(1+\log_2 n)^2.$$

Applying Lemma 2.3 to $A_i := P\widehat{v}_i$ immediately gives (5).

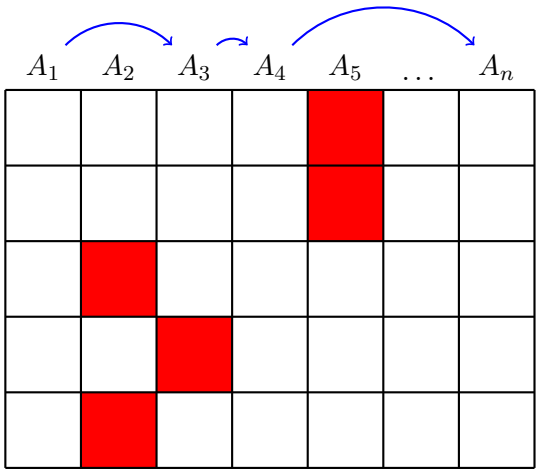


Figure 2: Lemma 2.3 states that, if the sum of squared distances across any subsequence of vectors $A_i$ is at most $B$, then the vector selecting the maximum value in each coordinate has squared norm $B \log^2 n$.

**Remark 2.4.** *The $\log^2 n$ factor in Lemma 2.3 is why we need $R > O(\log d \log n)$, rather than just $R > O(\log d)$. The factor in Lemma 2.3 is tight for $n = \Theta(d)$: $A_{i,j} := \log \frac{n}{1+|i-j|}$ has $B = \Theta(n)$ while $\sum_{j=1}^{d} \max_{i \in [n]} (A_i)_j^2$ is $\Theta(n \log^2 n)$.*

A similar approach, applied to $A_{i,j} = x_i^T P \widehat{v}_j$, lets us bound our actual target (4):

**Lemma 2.5.** *If $v_0 = v^*$, then*

$$\eta \sum_{i=1}^{n} \langle x_i, P\widehat{v}_{i-1} \rangle^2 \lesssim \sigma_2^2 \log^2 n \log \|v_n\|$$

Combined with (2) and (3), this implies that $\|v_n\| \geq e^{\Omega(\sigma_1)}$ if $\sigma_2 \ll \frac{1}{\log n}$:

**Lemma 2.6** (The right direction grows)**.** *Suppose $\sigma_2 < \frac{1}{2}$. Then if $v_0 = v^*$ we have*

$$\log \|v_n\| \gtrsim \frac{\sigma_1}{1 + \sigma_2^2 \log^2 n}.$$

Since the initial $v_0$ is random, it with high probability has a $\frac{1}{\text{poly}(d)}$ component in the $v^*$ direction; then linearity of the unnormalized algorithm means $\|v_n\|$ is large with high probability. By Lemma 2.1, this means the angle between $v^*$ and final answer $\widehat{v}_n$ is bounded by $\sqrt{\sigma_2} + d^{-C}$, so the algorithm succeeds.

## 2.2 Lower bound for mergeable summaries

In this section, we outline the proof of the mergeable summary lower bound. Specifically, we show with spectral gap $R = o(\sqrt{d})$, no mergeable summary algorithm can 0.1-approximate PCA using $O(d)$ space. Consider the scenario where there are $R$ players each possessing $d/R - 1$ vectors that are i.i.d. drawn from $\mathcal{N}(0, I_d)$. We then insert another vector $v^* \sim \mathcal{N}(0, I_d)$ into a random location in each player's data list. Consequently, from the viewpoint of each participant, their dataset

consists of $d/R$ i.i.d. Gaussian vectors, making it impossible for them to individually identify the shared vector $v^*$.

Now we consider the spectral properties of the overall data. With high probability, the variance in the direction of $v^*$ will be $\Theta(R)$ times larger than orthogonal directions. This implies that

1. The spectral gap of the data is at least $\Theta(R)$.

2. The principal component of the data is very close to $v^*$.

Therefore, we can employ a mergeable summary algorithm that approximates PCA to approximate $v^*$. Now we only need to prove that each player's summary must have at least $\omega(d)$ bits.

Suppose each player runs this algorithm and writes an $s$ bits long summary $S$ of their data to help approximate $v^*$. We measure the amount of information $S$ contains about $v^*$, i.e., $I(v^*; S)$. Because each player cannot distinguish $v^*$ from the other $d/R - 1$ vectors they have, we can prove that $I(S; v^*) \leq H(S)/(d/R) = R \cdot O(s/d)$. With $R = o(\sqrt{d})$ players, the combined summaries have at most $R \cdot I(S; v^*) \leq R^2 \cdot O(s/d) = o(s)$ bits of information about $v^*$. Since an approximation of $v^*$ has $\Theta(d)$ bits of information about $v^*$, this requires that $s = \omega(d)$.

### 2.3 Lower bound for high accuracy in insertion-only streams

To give an $\Omega(d^2)$ lower bound for constant $R$, we construct a two-player one-way communication game, where Alice feeds a uniformly random stream into the algorithm and passes the state to Bob. Bob then repeatedly takes this state, adds a few more vectors, and extracts the PCA estimate. We will show that Bob is able to learn $\Omega(d^2)$ bits about Alice's input, and therefore the stream state must have $\Omega(d^2)$ bits. Our approach is illustrated in Figure 3.

Suppose that Alice feeds in a random binary stream $x_1, x_2, \ldots, x_n \in \{-1, 1\}^d$. What can Bob insert so the PCA solution reveals information about (say) $x_1$?

First, suppose Bob inserted $k-1$ more copies of $x_1$ for some constant $k$. Then (if $n < d/100$) the PCA solution would be very close to $x_1$: $v = \frac{x_1}{\|x_1\|}$ has $\|Xv\|^2 \geq kd$ from just the copies of $x_1$, while every orthogonal direction has variance at most $(\sqrt{n}+\sqrt{d})^2 \approx 1.1d$ by standard bounds on singular values of subgaussian matrices [RV10]. Thus the spectral ratio $R = \frac{\lambda_1}{\lambda_2} > \frac{k}{1.1}$, so the streaming algorithm should return a vector highly correlated with $x_1$. The problem with this approach is that Bob cannot insert $x_1$ without knowing $x_1$, so the streaming PCA solution does not reveal any *new* information to him.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 1 | -1 | -1 | 1 | -1 | -1 | 1 | 1 |
| **1** | **1** | **1** | **-1** | **1** | **1** | **-1** | **-1** |
| -1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 |
| -1 | -1 | 1 | -1 | -1 | -1 | 1 | -1 |
| **1** | **1** | **1** | **-1** | 0 | 0 | 0 | 0 |
| **1** | **1** | **1** | **-1** | 0 | 0 | 0 | 0 |
| **1** | **1** | **1** | **-1** | 0 | 0 | 0 | 0 |

Figure 3: High-accuracy lower bound approach: Alice inserts a sequence of random bits (all but the last row). Bob knows the left side and wants to approximate the right side. To estimate the **blue** bits on the right, he adds $O(1)$ vectors using the corresponding **red** bits on the left and random bits on the right. With high probability, the principal component has constant correlation with the blue bits.

But what if Bob inserts $z_2, \ldots, z_k$ that match $x_1$ on the first half of bits, and are all 0 on the second half? The top principal component $u^*$ will still be highly correlated with $x_1$: the vector $v$ that matches $x_1, z_2, \ldots, z_k$ on the first half of bits and is zero on the rest has variance that is a $O(k)$ factor larger than any orthogonal direction.

A more careful analysis shows that the top principal component $v^*$ is not only correlated with the half of bits of $x_1$ shared with the $z_i$, but (on the remaining bits) is very highly correlated with the average $\frac{1}{k}(x_1 + z_2 + \cdots + z_k)$. In fact, it is *so* highly correlated with the average that $v^*$ must be at least somewhat—$\Theta(1/k^2)$—correlated with the last 10% of bits in $x_1$. This analysis is robust to a PCA approximation, so the streaming PCA algorithm lets Bob construct $\widehat{v}$ with constant correlation with the last half of bits in $x_1$.

Thus Bob can learn $\Omega(d)$ bits about the first row by inserting $z_2, \ldots, z_k$ that match the first half of bits and looking at the PCA solution on the last half of bits. If he does this for every row, he learns $\Omega(nd) = \Omega(d^2)$ bits about Alice's input. Therefore the algorithm state Alice sent needs $\Omega(d^2)$ space.

This construction is very similar to the one in [Woo14a] for lower-bounding low-rank Frobenius approximation. The difference in [Woo14a] is that Bob only inserts one row, so necessarily $R < 2$. Our main contribution here is the more careful analysis in terms of $R$.

# 3 Proof of Upper Bound

For most of this section we focus on Oja's method (Theorem 1.1), then in Section 3.4 we show Theorem 1.2. For simplicity, the proof is given assuming exact arithmetic. In Section 3.5 we discuss why $O(\log(nd))$ bits of precision suffice.

**Setup.** $\widehat{v}_i$ is the normalized state at time $i$, $v_i$ is the unnormalized state, $x_i$ is the sample, $\eta$ is the learning rate, $v^*$ is the direction of maximum variance, $P = I - v^*(v^*)^T$ to be the projection matrix that removes the $v^*$ component. Let $\sigma_1 = \eta \left\|X^T X\right\|$ and $\sigma_2 = \eta \left\|PX^T XP\right\|$, so:

$$\sum_{i=1}^{n} \langle v^*, x_i \rangle^2 = \sigma_1 \tag{6}$$

$$\eta \sum_{i=1}^{n} \langle w, x_i \rangle^2 \leq \sigma_2 \qquad (\forall w \perp v^*) \tag{7}$$

For much of the proof we will also need $\sigma_1 \geq C \log d$ and $\sigma_2 \leq \frac{1}{C \log n}$, but this will be stated as needed.

Oja's algorithm works by starting with a (typically random) vector $v_0$, then repeatedly applying Hebb's update rule that "neurons that fire together, wire together":

$$v_i = v_{i-1} + \eta \langle x_i, v_{i-1} \rangle x_i = (I + \eta x_i x_i^T) v_{i-1}. \tag{8}$$

The algorithm only keeps track of the normalized vectors $\widehat{v}_i = v_i / \|v_i\|$, but for analysis purposes we will often consider the unnormalized vectors $v_i$.

The norm $\|v_i\|$ grows in each step, according to

$$\|v_i\|^2 = \|v_{i-1}\|^2 \left(1 + (2\eta + \eta^2 \|x\|^2) \langle x_i, \widehat{v}_{i-1} \rangle^2\right), \tag{9}$$

and in particular (since Theorem 1.1 assumes $\eta \|x_i\|^2 \le 1$)

$$\log \frac{\|v_i\|^2}{\|v_{i-1}\|^2} \ge \eta \langle x_i, \widehat{v}_{i-1}\rangle^2. \tag{10}$$

Our goal is to show that $\widehat{v}_n \approx v^*$, or equivalently, that $\|P\widehat{v}_n\|$ is small.

### 3.1 Initial Lemmas

**Claim 3.1.** *Let $0 \le a_1, a_2, \dots, a_n$ and define $b_i = e^{\sum_{j\le i} a_i}$ for $i \in \{0, 1, \dots, n\}$. Then:*

$$\sum_{i=1}^{n} a_i b_{i-1} \le b_n - 1.$$

*Proof.* This follows from induction on $n$. $n = 0$ is trivial, and then

$$\sum_{i=1}^{n} a_i b_{i-1} \le b_{n-1} - 1 + a_n b_{n-1} = (1 + a_n) b_{n-1} - 1 \le e^{a_n} b_{n-1} - 1 = b_n - 1.$$

□

Define $B_i = \frac{\|v_i\|^2}{\|v_0\|^2}$, and $A_i = \log \frac{B_i}{B_{i-1}}$ which satisfies $A_i \ge \eta \langle x_i, \widehat{v}_{i-1}\rangle^2$ by (10). Therefore

$$\eta \sum_{i=1}^{n} \langle x_i, v_{i-1}\rangle^2 \le \|v_0\|^2 \sum_{i=1}^{n} A_i B_{i-1} \le \|v_0\|^2 (B_n - 1) = \|v_n\|^2 - \|v_0\|^2 \tag{11}$$

by Claim 3.1. Then for any unit vector $w$ with $Pw = w$,

$$\begin{aligned}
\langle v_n - v_0, w\rangle^2 &= \left(\eta \sum_{i=1}^{n} \langle x_i, v_{i-1}\rangle \langle x_i, w\rangle\right)^2 && \text{by (8)} \\
&\le \eta \sum_{i=1}^{n} \langle x_i, v_{i-1}\rangle^2 \cdot \eta \sum_{i=1}^{n} \langle x_i, w\rangle^2 && \text{by Cauchy-Schwarz} \\
&\le (\|v_n\|^2 - \|v_0\|^2)\sigma_2. && \text{by (11) and (7)}
\end{aligned}$$

There's nothing special about the start and final indices, giving the following bound for general indices $a \le b$:

$$\langle v_b - v_a, w\rangle^2 \le (\|v_b\|^2 - \|v_a\|^2)\sigma_2. \tag{12}$$

**Lemma 2.1** (Growth implies correctness)**.** *For any $v_0$ and all $i$, $\|P\widehat{v}_i\| \le \sqrt{\sigma_2} + \frac{\|Pv_0\|}{\|v_i\|}$.*

*Proof.* By (12), for any $w$ with $w = Pw$,

$$\langle v_i - v_0, w\rangle \le \sqrt{\sigma_2}\, \|v_i\|.$$

Hence

$$\langle \widehat{v}_i, w\rangle = \frac{\langle v_i - v_0, w\rangle + \langle v_0, w\rangle}{\|v_i\|} \le \sqrt{\sigma_2} + \frac{\langle v_0, w\rangle}{\|v_i\|}.$$

Setting $w = P\widehat{v}_i / \|P\widehat{v}_i\|$, we have $\langle \widehat{v}_i, w\rangle = \|P\widehat{v}_i\|$ and $\langle v_0, w\rangle \le \|Pv_0\|$, giving the result. □

Lemma 2.1 implies that, if we start at $v^*$, we never move by more than $\sqrt{\sigma_2}$ from it. We now show that you cannot even move $\sqrt{\sigma_2}$ without increasing the norm of $v$.

**Lemma 2.2.** *Suppose $Pv_0 = 0$. For any two time steps $0 \le a < b \le n$,*

$$\|P\widehat{v}_b - P\widehat{v}_a\|^2 \le 4\sigma_2 \log \frac{\|v_b\|}{\|v_a\|}$$

*Proof.* Define $w$ to be the unit vector in direction $P(\widehat{v}_b - \widehat{v}_a)$. By (12) we have

$$\langle v_b - v_a, w\rangle^2 \le \sigma_2(\|v_b\|^2 - \|v_a\|^2).$$

Therefore

$$\begin{aligned}
\|P\widehat{v}_b - P\widehat{v}_a\|^2 = \langle P(\widehat{v}_b - \widehat{v}_a), w\rangle^2 &= \langle \widehat{v}_b - \widehat{v}_a, w\rangle^2 \\
&\le 2\langle \widehat{v}_b - \frac{\|v_a\|}{\|v_b\|}\widehat{v}_a, w\rangle^2 + 2\langle \frac{\|v_a\|}{\|v_b\|}\widehat{v}_a - \widehat{v}_a, w\rangle^2 \\
&\le 2\frac{1}{\|v_b\|^2}\langle v_b - v_a, w\rangle^2 + 2(\frac{\|v_a\|}{\|v_b\|} - 1)^2 \|P\widehat{v}_a\|^2 \\
&\le 2\sigma_2(1 - \frac{\|v_a\|^2}{\|v_b\|^2}) + 2(1 - \frac{\|v_a\|}{\|v_b\|})^2\sigma_2 \\
&= 4\sigma_2(1 - \frac{\|v_a\|}{\|v_b\|}).
\end{aligned}$$

Finally, $(1 - 1/x) \le \log x$ for all $x > 0$. ∎

### 3.2 Results on Sequences

The following combinatorial result is written in a self-contained fashion, independent of the streaming PCA application.

**Lemma 3.2.** *Let $A \in \mathbb{R}^{d\times n}$ have first column all zero. Define $b_i^{(k)}$ to be column $1+2^k i$ of $A$. Then:*

$$\sum_i \max_j A_{ij}^2 \le (1 + \log_2 n) \sum_{k=0}^{\log_2 n} \sum_{j>0} \left\| b_j^{(k)} - b_{j-1}^{(k)} \right\|^2 .$$

*Proof.* We will show this separately for each row $i$; the result is just the sum over these rows. For fixed $i$, let $j^* = \arg\max_j A_{ij}^2$.

Let $j^{(k)} = 1 + 2^k \lfloor \frac{j^*-1}{2^k} \rfloor$ set the last $k$ bits of $j^* - 1$ to zero. We have that $j^{(0)} = j^*$ and $j^{\log_2 n} = 0$. Therefore

$$A_{ij^*} = \sum_{k=0}^{\log_2 n} (A_{i,j^{(k)}} - A_{i,j^{(k+1)}}).$$

Now, $j^{(k)}$ is either $j^{(k+1)}$ or $j^{(k+1)} + 2^k$. Each value in the right sum is either zero (if $j^{(k)}$ is $j^{(k+1)}$) or the $i$th coordinate of $b_{j'}^{(k)} - b_{j'-1}^{(k)}$ for some $j'$ (if $j^{(k)} = j^{(k+1)} + 2^k$, using $j' = j^{(k)}/2^k$). Thus, by Cauchy-Schwarz,

$$\begin{aligned}
A_{ij^*}^2 &\le (1 + \log_2 n) \cdot \sum_{k=0}^{\log_2 n} (A_{i,j^{(k)}} - A_{i,j^{(k+1)}})^2 \\
&\le (1 + \log_2 n) \cdot \sum_{k=0}^{\log_2 n} \sum_{j>0} ((b_j^{(k)})_i - (b_{j-1}^{(k)})_i)^2.
\end{aligned}$$

Summing over $i$,

$$\sum_i \max_j A_{ij}^2 \le (1+\log_2 n) \sum_{k=0}^{\log_2 n} \sum_{j>0} \left\| b_j^{(k)} - b_{j-1}^{(k)} \right\|^2.$$

□

### 3.3 Proof of Growth

We return to the streaming PCA setting. The goal of this section is to show that, if $v_0 = v^*$, then $\|v_n\|$ is large.

**Lemma 2.5.** *If $v_0 = v^*$, then*

$$\eta \sum_{i=1}^n \langle x_i, P\widehat{v}_{i-1}\rangle^2 \lesssim \sigma_2^2 \log^2 n \log \|v_n\|$$

*Proof.* Define $u_i = P\widehat{v}_i$. We apply Lemma 3.2 to the matrix $A_{ij} = \langle x_i, u_{j-1}\rangle$ for $i, j \in [n]$, getting:

$$\sum_{i=1}^n \max_{j \le n-1} \langle x_i, u_j\rangle^2 \le (1+\log_2 n) \sum_{k=0}^{\log_2 n} \sum_{j>0} \sum_{i=1}^n (\langle x_i, u_{2^k j}\rangle - \langle x_i, u_{2^k(j-1)}\rangle)^2.$$

Now,

$$\begin{aligned}\sum_{i=1}^n (\langle x_i, u_{2^k j}\rangle - \langle x_i, u_{2^k(j-1)}\rangle)^2 &= (u_{2^k j} - u_{2^k(j-1)}) X^T X (u_{2^k j} - u_{2^k(j-1)}) \\ &\le \frac{\sigma_2}{\eta} \left\| u_{2^k j} - u_{2^k(j-1)} \right\|^2.\end{aligned}$$

by the assumption (7) on $X$ and that every $u_j \perp v^*$. Then, for each $k$, Lemma 2.2 shows that

$$\sum_{j>0} \left\| u_{2^k j} - u_{2^k(j-1)} \right\|^2 \le 4\sigma_2 \log \frac{\|v_n\|}{\|v_0\|} = 4\sigma_2 \log \|v_n\|$$

and thus

$$\eta \sum_i \langle x_i, P\widehat{v}_{i-1}\rangle^2 \le \eta \sum_i \max_j \langle x_i, u_j\rangle^2 \le (1+\log_2 n) \sum_{k=0}^{\log_2 n} 4\sigma_2^2 \log \|v_n\| \lesssim \sigma_2^2 \log^2 n \log \|v_n\|$$

as desired. □

**Lemma 2.6** (The right direction grows)**.** *Suppose $\sigma_2 < \frac{1}{2}$. Then if $v_0 = v^*$ we have*

$$\log \|v_n\| \gtrsim \frac{\sigma_1}{1+\sigma_2^2 \log^2 n}.$$

*Proof.* We will show that $\eta \sum_{i=1}^n \langle x_i, \widehat{v}_{i-1}\rangle^2 \gtrsim \sigma_1$, giving the result by (10).

Recall that $(x+y)^2 \ge \frac{1}{2}x^2 - y^2$ for all $x, y$. Thus, if $\widehat{v}_i = a_i v^* + u_i$ for $u_i \perp v^*$, we have

$$\langle x_i, \widehat{v}_{i-1}\rangle^2 \ge \frac{a_{i-1}^2}{2} \langle x_i, v^*\rangle^2 - \langle x_i, u_{i-1}\rangle^2.$$

Lemma 2.1 shows that $a_i^2 \geq 1 - \sigma_2 \geq \frac{1}{2}$, so summing up over $i$,

$$\eta \sum_{i=1}^{n} \langle x_i, \widehat{v}_{i-1} \rangle^2 \geq \frac{1}{4}\sigma_1 - \eta \sum_{i=1}^{n} \langle x_i, u_{i-1} \rangle^2.$$

Then (10) and Lemma 2.5 give

$$\log \|v_n\| \geq \frac{1}{2}\eta \sum_{i=1}^{n} \langle x_i, \widehat{v}_{i-1} \rangle^2 \geq \frac{1}{8}\sigma_1 - O(\sigma_2^2 \log^2 n \log \|v_n\|),$$

or

$$\log \|v_n\| \gtrsim \frac{\sigma_1}{1 + \sigma_2^2 \log^2 n}.$$

□

**Claim 3.3.** *Let $a \sim N(0,1)$. For any two vectors $u$ and $v$, with probability $1 - \delta$,*

$$\|au + v\| \geq \delta\sqrt{\pi/2}\,\|u\|.$$

*Proof.* First, without loss of generality $v$ is collinear with $u$; any orthogonal component only helps. So we can only consider real-valued $u$ and $v$, and in fact rescale so $u = 1$. The claim is then: with probability $1 - \delta$, a sample from $N(v, 1)$ has absolute value at least $\delta\sqrt{\pi/2}$. This follows from the standard Gaussian density being at most $1/\sqrt{2\pi}$. □

**Theorem 1.1** (Performance of Oja's method in adversarial streams)**.** *For any sufficiently large universal constant $C$, suppose $\eta$ is such that $\eta n \lambda_1 > C \log d$ and $\eta n \lambda_2 < \frac{1}{C \log n}$. If $\eta \|x_i\|^2 \leq 1$ for every $i$, then Oja's algorithm with learning rate $\eta$ returns $\widehat{v}$ satisfying $\|P\widehat{v}\| \leq \sqrt{\eta n \lambda_2} + d^{-9}$ with $1 - d^{-\Omega(C)}$ probability.*

*Moreover, Oja's method can be modified (Algorithm 1) so that in addition, regardless of $\lambda_1$ and $\lambda_2$, if $\eta \|x_i\|^2 \leq 1$ for all $i$ then either $\|P\widehat{v}\| \leq \sqrt{\eta n \lambda_2} + d^{-9}$ or $\widehat{v} = \perp$.*

*Proof.* We assume that $\eta \|x_i\|^2 \leq 1$ for all $i$, since the theorem is otherwise vacuous.

We begin with the last statement. Algorithm 1 only returns $\widehat{v} \neq \perp$ if $s_n = \log \frac{\|v_n\|}{\|v_0\|} > 10 \log d$. But then by Lemma 2.1,

$$\|P\widehat{v}_n\| \leq \sqrt{\sigma_2} + \frac{\|v_0\|}{\|v_n\|} \leq \sqrt{\sigma_2} + d^{-10}.$$

All that remains is to show that, if $\sigma_1 > C \log d$ and $\sigma_2 < \frac{1}{C \log n}$, $\widehat{v} \neq \perp$ with at least $1 - d^{-\Omega(C)}$ probability. And of course, $\widehat{v} \neq \perp$ if $\frac{\|v_n\|}{\|v_0\|} \geq d^{10}$.

Oja's algorithm starts with $\widehat{v}_0$ uniformly on the sphere, and is indifferent to the initial scale $\|v_0\|$, so $v_0$ could be constructed as $\frac{v_0}{\|v_0\|}$ for $v_0 \sim N(0, I_d)$.

Let $v_0 = av^* + u$ for $u \perp v^*$. Let $B = \prod_{i=1}^{n}(I + \eta x_i x_i^T)$, so $v_n = Bv_0$.

By Lemma 2.6 and the bound on $\sigma_2$, we know $\|Bv^*\| \geq e^{c'\sigma_1}$ for some constant $c'$. Then by Claim 3.3, with probability $1 - \delta$,

$$\|v_n\| = \|aBv^* + Bu\| \geq \delta\sqrt{\pi/2}\,\|Bv^*\| \geq \delta e^{c'\sigma_1}.$$

The (very naive) Markov bound from $\mathbb{E}[\|v_0\|^2] = d$ gives that

$$\frac{\|v_n\|}{\|v_0\|} \geq \frac{\delta^{3/2} e^{c'\sigma_1}}{\sqrt{d}}$$

with probability $1 - 2\delta$. For sufficiently large $C$ in $\sigma_1 \geq C \log d$, this gives

$$\frac{\|v_n\|}{\|v_0\|} \geq d^{10}$$

with probability $1 - d^{-\Omega(C)}$. □

### 3.4 Proof of Theorem 1.2

**Theorem 1.2** (Full algorithm). *For $X \in \mathbb{R}^{n \times d}$ have $b$-bit entries for $b > \log(dn)$. Whenever the spectral gap $R = \lambda_1/\lambda_2 > O(\log n \log d)$, Algorithm 2 uses $O(b^2 d)$ bits of space and $O(\frac{\log d}{R} + d^{-9})$-approximates PCA with high probability.*

*Proof.* Let $C$ be the constant in Theorem 1.1. For $R$ to be well defined, $\lambda_1 \neq 0$ so some $x_i \neq 0$. Therefore $2^{-2b} \leq \lambda_1 \leq nd^2 2^{2b}$. Thus one of the $\eta_i$ considered in Algorithm 2 is such that $\eta n \lambda_1 \in [C \log d, 2C \log d]$. Let $\widehat{i}$ be this $i$. For sufficiently large constant in the choice of $R$, we have

$$\eta_i n \lambda_2 \leq \frac{1}{C \log n}$$

for all $i \leq \widehat{i}$.

Let $\overline{x}$ be the $x_i$ of maximum norm, as computed by the algorithm. We now show that $\widehat{x} := \frac{\overline{x}}{\|\overline{x}\|}$ is a sufficiently good answer if $\eta_{\widehat{i}} \|\overline{x}\|^2 \geq 1$. Decompose $\overline{x} = av^* + bw$ for $w \perp v^*$ a unit vector. By (7), $b$ is fairly small:

$$b^2 \leq \eta_{\widehat{i}} \sum_i \langle x_i, w \rangle^2 \leq \left\| PX^T XP \right\| = n\lambda_2 \leq \frac{2C \log d}{R \eta_{\widehat{i}}}.$$

The unit vector $\widehat{x}$ in direction $\overline{x}$ has error

$$\|P\widehat{x}\|^2 = \frac{b^2}{\|\overline{x}\|^2} \leq \frac{2C \log d}{R \eta_{\widehat{i}} \|\overline{x}\|^2} \lesssim \frac{\log d}{R \eta_{\widehat{i}} \|\overline{x}\|^2}. \tag{13}$$

Therefore if $\eta_{\widehat{i}} \|\overline{x}\|^2 \geq 1$, $\widehat{x}$ is a sufficiently accurate answer.

The last statement in Theorem 1.1 shows that, if $\eta_{i^*} \|\overline{x}\|^2 \leq 1$ and $i^* \leq \widehat{i}$, then

$$\left\| Pv^{(i^*)} \right\|^2 \leq (\sqrt{\eta_{i^*} n \lambda_2} + d^{-9})^2 \lesssim \eta_{i^*} n \lambda_2 + d^{-18} \leq \frac{2C \log d}{R} + d^{-18} \tag{14}$$

which is sufficiently accurate. We now split into case analysis.

**In one case**, suppose $\eta_{\widehat{i}} \|\overline{x}\|^2 < 1$. Therefore the main body of Theorem 1.1 states that $v^{(\widehat{i})} \neq \perp$ with high probability. In particular, this means $i^* \leq \widehat{i}$, so $\eta_{i^*} \|\overline{x}\|^2 < 1$, and the algorithm's answer is $v^{(i^*)}$ which is sufficiently accurate by (14).

**Otherwise**, $\eta_{\widehat{i}} \|\overline{x}\|^2 \geq 1$. Then outputting $\overline{x}$ is sufficiently accurate by (13). If $i^* \geq \widehat{i}$, the algorithm will definitely output $\overline{x}$; if $i^* < \widehat{i}$, the algorithm might output $v^{(i^*)}$, but only if $\eta_{i^*} \|\overline{x}\|^2 < 1$, in which case this is sufficiently accurate by (14). □

### 3.5 Precision

Finally, we discuss why $O(\log(nd))$ bits of precision suffice for the algorithm. Algorithm 1 tracks two values: a unit vector $\widehat{v}_i$ and the log-norm $s_i$ of the unnormalized $v_i$. The main concern is that the error in $\widehat{v}_i$ could compound.

Consider $\widehat{v}_i$ and $s_i$ to be the values computed by the algorithm, which has some $\varepsilon = \frac{1}{\text{poly}(nd)}$ error (in $\ell_2$) added in each iteration. We can enforce $s_i \geq s_{i-1}$ despite the error. Redefine $v_i$ to $2^{s_i}\widehat{v}_i$.

We now redo the proof of (12) with $\varepsilon$ error in each step. Define $B_i = \frac{\|v_i\|^2}{\|v_0\|^2} = 2^{s_i}$, and $A_i = \log \frac{B_i}{B_{i-1}} = (s_i - s_{i-1})$ which satisfies $A_i \geq \eta\langle x_i, \widehat{v}_{i-1}\rangle^2 - O(\varepsilon)$ by (10). Therefore

$$\eta \sum_{i=1}^{n} \langle x_i, v_{i-1}\rangle^2 \leq \|v_0\|^2 \sum_{i=1}^{n} (A_i + O(\varepsilon))B_{i-1} \leq \|v_0\|^2 \left((B_n - 1) + O(\varepsilon n B_n)\right) = \|v_n\|^2 - \|v_0\|^2 + O(\varepsilon n \|v_n\|^2) \tag{15}$$

by Claim 3.1. Then for any unit vector $w$ with $Pw = w$,

$$\begin{aligned}
\langle v_n - v_0, w\rangle^2 &= (\sum_{i=1}^{n} \langle v_i - v_{i-1}, w\rangle)^2 \\
&= (\sum_{i=1}^{n} \eta\langle x_i, v_{i-1}\rangle\langle x_i, w\rangle + O(\varepsilon)\|v_{i-1}\|)^2 \\
&= \left(O(n\varepsilon\|v_n\|) + \eta\sum_{i=1}^{n}\langle x_i, v_{i-1}\rangle\langle x_i, w\rangle\right)^2 && \text{by (8)} \\
&\leq \eta\sum_{i=1}^{n}\langle x_i, v_{i-1}\rangle^2 \cdot \eta\sum_{i=1}^{n}\langle x_i, w\rangle^2 + O(n^2\varepsilon\|v_n\|^2) && \text{by Cauchy-Schwarz} \\
&\leq (\|v_n\|^2 - \|v_0\|^2)\sigma_2 + O(\varepsilon n^2\|v_n\|^2). && \text{by (15) and (7)}
\end{aligned}$$

There's nothing special about the start and final indices, giving the following bound for general indices $a \leq b$:

$$\langle v_b - v_a, w\rangle^2 \leq (\|v_b\|^2 - \|v_a\|^2)\sigma_2 + O(\varepsilon n^2\|v_b\|^2). \tag{16}$$

Given (16), the error tolerance flows through the rest of the proof easily. Lemmas 2.1 and 2.2 follow immediately with $O(\varepsilon n^2)$ additive error. Lemma 2.5 gets additive error $O(\sigma_2\varepsilon n^3 \log^2 n)$, so both the numerator and denominator of Lemma 2.6 change by $\varepsilon\,\text{poly}(n)$. Both the conditions and result of Theorem 1.1 only change by an additive $\varepsilon\,\text{poly}(n)$ error, which for sufficiently small polynomial $\varepsilon$ are absorbed by the constant factors and $\frac{1}{d^9}$ additive error. And Algorithm 2 does nothing that could compound the error by more than a constant factor, so Theorem 1.2 holds as well.

## 4 Lower Bound for Mergeable Summaries

In this section, we show that any all the mergeable summaries require $\Omega(d^2/R^2)$ bits of space, even just to approximate PCA with 0.1 error. This is significantly worse than our upper bound for streaming algorithms.

**Theorem 1.3** (Mergeable Lower Bound)**.** *For all mergeable summaries,* 0.1*-approximate PCA on streams with spectral gap* $R$ *requires at least* $\Omega(d^2/R^2)$ *bits of space.*

To prove the theorem, for $p > 1$, we define distribution $\mathcal{D}_p$ over $\mathbb{R}^{d\times d}$ such that $X \sim \mathcal{D}_p$ is drawn according to the following randomized procedure:

1. Sample $v^* \sim \mathcal{N}(0, I_d)$.
2. Define $k := d/p$. For each $i \in [p]$, we sample a $X^{(i)} \in \mathbb{R}^{k\times d}$ as follows: Randomly choose a $j_i^* \in [k]$ and set $X^{(i)}_{j_i^*}$ to be $v^*$. For $j \in [k] \setminus \{j_i^*\}$, independently sample $X^{(i)}_j \sim \mathcal{N}(0, I_d)$. Let $X^{(i)} := (X^{(i)}_1, X^{(i)}_2, \ldots, X^{(i)}_k)^T$.
3. Finally, let $X$ be the concatenation of all the $X^{(i)}$'s, i.e.,

$$X := \begin{pmatrix} X^{(1)} \\ X^{(2)} \\ \vdots \\ X^{(p)} \end{pmatrix}.$$

We first show that for $X \sim \mathcal{D}_p$, $X^TX$ has a large spectral gap with high probability:

**Lemma 4.1.** *For* $X \sim \mathcal{D}$*, with* $1 - o(1)$ *probability,*

$$\min_{\substack{v'\perp v^* \\ \|v'\|=1}} \frac{\|X\widehat{v^*}\|^2}{\|Xv'\|^2} \geq 0.1p.$$

*Furthermore, the spectral gap of* $X^TX$ *is at least* $0.1p$*.*

*Proof.* We can decompose the rows of $X$ into two parts: repetitions of $v^*$ and other randomly sampled rows. Define $\widetilde{X} \in \mathbb{R}^{(d-p)\times d}$ as $X$ excluding the $v^*$'s in each $X^{(i)}$.

For an arbitrary unit vector $u \in \mathbb{R}^d$, $X$'s variance on $u$ is equal to

$$\|Xu\|^2 = p\langle v^*, u\rangle^2 + \|\widetilde{X}u\|^2.$$

Therefore, $X$'s variance at the direction of $v^*$ is

$$\|X\widehat{v^*}\|^2 = p\langle v^*, \widehat{v^*}\rangle^2 + \|\widetilde{X}\widehat{v^*}\|^2 \geq p\langle v^*, \widehat{v^*}\rangle^2 = p\,\|v^*\|^2.$$

For any unit vector $u$ orthogonal to $v^*$, we have

$$\|Xu\|^2 = p\langle v^*, u\rangle^2 + \|\widetilde{X}u\|^2 = \|\widetilde{X}u\|^2 \leq \|\widetilde{X}\|^2.$$

Utilizing this, we have

$$\min_{\substack{v'\perp v^* \\ \|v'\|=1}} \frac{\|X\widehat{v^*}\|^2}{\|Xv'\|^2} \geq \frac{p\|v^*\|^2}{\|\widetilde{X}\|^2}.$$

Note that $v^* \sim \mathcal{N}(0, I_d)$, by Lemma A.5, $\|v^*\|^2 \geq 0.9d$ holds with probability at least $1 - o(1)$. In addition, since every instance in $\widetilde{X}$ follows $\mathcal{N}(0, 1)$ independently, by Lemma A.1, we have

$$\Pr\left[\|\widetilde{X}\| \geq 3\sqrt{d}\right] \leq o(1).$$

This states that with probability $1 - o(1)$,

$$\min_{\substack{v' \perp v^* \\ \|v'\|=1}} \frac{\|X\widehat{v^*}\|^2}{\|Xv'\|^2} \geq \frac{p\|v^*\|^2}{\|\widetilde{X}\|^2} \geq \frac{0.9pd}{9d} = 0.1p.$$

Furthermore, the spectral gap of $X^TX$ is given by

$$\max_{\|v\|=1} \min_{\substack{v' \perp v \\ \|v'\|=1}} \frac{\|Xv\|^2}{\|Xv'\|^2} \geq \min_{\substack{v' \perp v^* \\ \|v'\|=1}} \frac{\|X\widehat{v^*}\|^2}{\|Xv'\|^2} \geq 0.1p.$$

□

**Lemma 4.2.** *For $X \sim \mathcal{D}$, let $v$ be the top eigenvalue of $X^TX$. Let $\alpha$ be an arbitrarily small positive constant. There exists a constant $C$ such that when $p \geq C$, with $1 - o(1)$ probability, $\sin^2(\widehat{v^*}, v) \leq \alpha^2$.*

*Proof.* Without loss of generality, we express $v$ as $v = \sqrt{1-\varepsilon^2}\widehat{v^*} + \varepsilon u$ for some $\varepsilon > 0$ and unit vector $u$ orthogonal to $v^*$. We only need to prove that $\varepsilon \leq \alpha$.

We have

$$\|Xv\|^2 \leq (1-\varepsilon^2)\|X\widehat{v^*}\|^2 + \varepsilon^2\|Xu\|^2 + 2\sqrt{1-\varepsilon^2}\varepsilon\|X\widehat{v^*}\| \cdot \|Xu\|.$$

By Lemma 4.1, we have with probability $1 - o(1)$, $\|X\widehat{v^*}\|^2 \geq 0.1p\|Xu\|^2$. Therefore,

$$\|Xv\|^2 \leq \|X\widehat{v^*}\|^2(1 - \varepsilon^2 + \varepsilon^2\frac{10}{p} + 2\varepsilon\sqrt{\frac{10}{p}}).$$

When $\varepsilon > \alpha$, there exists a constant $C > 0$ such that for $p \geq C$,

$$\|X\widehat{v^*}\|^2(1 - \varepsilon^2 + \varepsilon^2\frac{10}{p} + 2\varepsilon\sqrt{\frac{10}{p}}) \leq \|X\widehat{v^*}\|^2(1 - \frac{\varepsilon^2}{2}) \leq \|X\widehat{v^*}\|^2.$$

This contradicts the assumption that the direction of $v$ has larger variance than $v^*$. This proves the lemma. □

Let $\mathcal{A}$ be an arbitrary deterministic mergeable summary for PCA that satisfies

$$\Pr_{X \sim \mathcal{D}_p}[\sin^2(\mathcal{A}(X), v^*) \leq 0.105] \geq 0.9. \tag{17}$$

**Lemma 4.3.** *$\mathcal{A}$ requires $\Omega(d^2/p^2)$ bits of space.*

Before proving Lemma 4.3, we first show the proof of Theorem 1.3 assuming Lemma 4.3 is true.

*Proof of Theorem 1.3.* Suppose we have a mergeable summary $\mathcal{S}$ that 0.1-approximates PCA uses $o(d^2/R^2)$ bits of space with high probability. Let $\alpha > 0$ be a constant such that

$$\sin^2(\arcsin\alpha + \arcsin\sqrt{0.1}) \leq 0.105.$$

Let $C$ be the constant in Lemma 4.2 corresponding to $\alpha$. By Lemma 4.1, for $X \sim \mathcal{D}_{10R+C}$, $X^TX$ has spectral gap $R$ with high probability. Therefore, $\mathcal{S}$ succeeds in 0.1-approximating PCA with high probability. Combining with Lemma 4.2, we have $\mathcal{S}$ succeeds in 0.105-approximating $v^*$ on $X \sim \mathcal{D}_{10R+C}$ with high probability.

By Yao's minimax principle, there must be a *deterministic* mergeable summary $\mathcal{A}$ that also uses $o(d^2/R^2)$ bits of space and 0.105-approximates PCA on $X \sim \mathcal{D}_{10R+C}$ with high probability, i.e., it satisfies (17). By Lemma 4.3, $\mathcal{A}$ must require $\Omega(d^2/R^2)$ bits of space, which is a contradiction.

□

We use $s$ to denote the bits of space of $\mathcal{A}$. To prove Lemma 4.3, we will show that $s = \Omega(d^2/p^2)$. We use $m_i$ to denote $\mathcal{A}$'s summary for $X^{(i)}$. The key property we use here is that each $m_i$ is a deterministic function of $X^{(i)}$, so $m_i$'s are independent except for the shared vector $v^*$. We start with the following classical result:

**Proposition 4.4** (Chain Rule for Mutual Information)**.** *Let $X$, $Y$, $Z$ be random variables. We have*

$$I(X;Y \mid Z) = I(X;Y) - (I(X;Z) - I(X;Z \mid Y)).$$

**Corollary 4.5.** *Let $X$, $Y$, $Z$ be random variables. If $X$ and $Z$ are independent,*

$$I(X;Y) \leq I(X;Y \mid Z).$$

*Proof.* Since $X$ and $Z$ are independent, $I(X;Z) = 0$. Applying proposition 4.4 gives the result. □

Using these results, we can prove the next two lemmas bounding the mutual information between $v^*$ and $m_i$'s in terms of $s$.

**Lemma 4.6.** *For every $i \in [p]$, $I(m_i; v^*) \leq s/k$.*

*Proof.* Since $m_i$ and $j_i^*$ are independent, by corollary 4.5 we have

$$I(m_i; X^{(i)}_{j_i^*}) \leq I(m_i; X^{(i)}_{j_i^*} \mid j_i^*).$$

Thus,

$$I(m_i; v^*) = I(m_i; X^{(i)}_{j_i^*}) \leq I(m_i; X^{(i)}_{j_i^*} \mid j_i^*) = \frac{1}{k}\sum_{j\in[k]} I(m_i; X^{(i)}_j \mid j_i^* = j) = \frac{1}{k}\sum_{j\in[k]} I(m_i; X^{(i)}_j).$$

Furthermore, since each $X^{(i)}_j$ is sampled independently, by applying corollary 4.5, we have for each $j \in [k]$,

$$I(m_i; X^{(i)}_j) \leq I(m_i; X^{(i)}_j \mid X^{(i)}_1, \ldots, X^{(i)}_{j-1}).$$

We have

$$\sum_{j\in[k]} I(m_i; X^{(i)}_j) \leq \sum_{j\in[k]} I(m_i; X^{(i)}_j \mid X^{(i)}_1, \ldots, X^{(i)}_{j-1}) = I(m_i; X^{(i)}_1, \ldots, X^{(i)}_n) \leq H(m_i).$$

Since $\mathcal{A}$ only has $s$ bits of space, $H(m_i) \leq s$. Therefore,

$$I(m_i; v^*) \leq \frac{1}{k} H(m_i) \leq \frac{s}{k}.$$

□

**Lemma 4.7.** $I(v^*; m_1, m_2, \ldots, m_p) \leq p^2 s/d$

*Proof.* We first prove that for $i \in [p]$, $I(v^*; m_i \mid m_1, \ldots, m_{i-1}) \leq I(v^*; m_i)$. Note that

$$I(m_i; m_1, \ldots, m_{i-1} \mid v^*) \leq I(X^{(i)}; X^{(1)}, \ldots, X^{(i-1)} \mid v^*) = 0.$$

By proposition 4.4, we have

$$I(v^*; m_i \mid m_1, \ldots, m_{i-1}) = I(v^*; m_i) - I(m_i; m_1, \ldots, m_{i-1}) + I(m_i; m_1, \ldots, m_{i-1} \mid v^*) \leq I(v^*; m_i).$$

Then by Lemma 4.6, we have

$$I(v^*; m_1, \ldots, m_p) = \sum_{i\in[p]} I(v^*; m_i \mid m_1, \ldots, m_{i-1}) \le \sum_{i\in[p]} I(v^*; m_i) \le \frac{ps}{k} = \frac{p^2 s}{d}.$$

□

Next, we show that the mutual information between $v^*$ and the output of $\mathcal{A}$ must be at least $\Omega(d)$. For this purpose, we refer to a special case of lemma 4.4 from [JKDP21]:

**Lemma 4.8** (Lemma 4.4 of [JKDP21])**.** *Consider random variable $x$ uniformly distributed over $D \subseteq \mathbb{R}^d$ and random variable $\tilde{x}$ in $\mathbb{R}^d$. If the joint distribution of $(x, \tilde{x})$ satisfies*

$$\Pr[\|x - \tilde{x}\| \le \eta] \ge 0.9,$$

*then we have*

$$\frac{1}{8}\log \mathrm{Cov}_{3\eta,1/2}(D) \le I(x; \tilde{x}) + 1.98,$$

*where $\mathrm{Cov}_{3\eta,1/2}$ denotes the minimum number of $d$-dimensional balls of radius $3\eta$ required to cover at least half of $D$.*

**Lemma 4.9.** *Let unit vector $\tilde{v}$ be an approximation of $\widehat{v^*}$ such that*

$$\Pr[\sin^2(v^*, \tilde{v}) \le 0.105] \ge 0.9.$$

*Then,*

$$I(\widehat{v^*}; \tilde{v}) \gtrsim d.$$

*Proof.* We define $\tilde{v}' := \mathrm{sign}(\tilde{v}^T v^*)\tilde{v}$. Then it is easy to verify that $\sin^2(v^*, \tilde{v}) \le 0.105$ implies that $\|\widehat{v^*} - \tilde{v}'\| \le 1/3 - c$ for some constant $c > 0$. Therefore, by Lemma 4.8, we have

$$I(\widehat{v^*}; \tilde{v}') \ge \frac{1}{8}\log \mathrm{Cov}_{1-3c,1/2}(S_d) - 1.98,$$

where $S_d$ denotes the $d$-dimensional unit sphere. Note that each ball of radius $1 - 3c$ can cover a spherical cap with height at most $1 - 3c$ on the unit sphere, and the union of these caps need to cover at least half of the surface area of a unit sphere. Using a bound on the area of a spherical cap (Lemma A.6), we have

$$\log \mathrm{Cov}_{1-3c,1/2} \ge \log \frac{\text{area of } S_d}{\text{area of height-}(1-3c)\text{ spherical cap}} \gtrsim d.$$

Therefore,

$$I(\widehat{v^*}; \tilde{v}') \gtrsim d.$$

This implies that

$$d \lesssim I(\widehat{v^*}; \tilde{v}') = I(\widehat{v^*}; \mathrm{sign}(\tilde{v}^T v^*)\tilde{v}) \le I(\widehat{v^*}; \tilde{v}, \mathrm{sign}(\tilde{v}^T v^*)) = I(\widehat{v^*}; \tilde{v}) + I(\widehat{v^*}; \mathrm{sign}(\tilde{v}^T v^*) \mid \tilde{v}).$$

In addition, since

$$I(\widehat{v^*}; \mathrm{sign}(\tilde{v}^T v^*) \mid \tilde{v}) \le H(\mathrm{sign}(\tilde{v}^T v^*)) \le 1,$$

we have

$$I(\widehat{v^*}; \tilde{v}) \gtrsim d.$$

□

This gives us a lower bound for $s$:

*Proof of Lemma 4.3.* Let $\tilde{v}$ be the output of $\mathcal{A}$. By Lemma 4.9, we have

$$I(\tilde{v}; \widehat{v^*}) \gtrsim d.$$

By Lemma 4.7,

$$I(m_1, \ldots, m_p; v^*) \leq \frac{p^2 s}{d}.$$

Using the data processing inequality, we get

$$d \lesssim I(\tilde{v}; \widehat{v^*}) \leq I(m_1, \ldots, m_p; v^*) \leq \frac{p^2 s}{d}.$$

Therefore,

$$s \gtrsim \frac{d^2}{p^2}.$$

□

# 5 Lower Bound for Accuracy

Our lower bound is based on the PartialDuplicate instance, where an instance is a matrix $X \in \{0, -1, 1\}^{(k+n+1)\times d}$ can be expressed as follows:

- The first row equals $x + y$, where $x, y \in \{0, -1, 1\}^d$ have $\mathrm{supp}(x) = \{1, 2, \ldots, d/2\}$ and $\mathrm{supp}(y) = \{d/2 + 1, \ldots, d\}$.
- For $i \in \{2, \ldots, k+1\}$, the $i$-th row equals $x$.
- The last $n$ rows form a uniformly random matrix $X' \in \{-1, 1\}^{n\times d}$.

That is, the entries look like:

$$X = \begin{array}{|cc|}\hline x & y \\ x & 0 \\ \vdots & \vdots \\ x & 0 \\ \hline \multicolumn{2}{|c|}{X'} \\ \hline \end{array}$$

except that $x, y$ are zero-padded to $d$ dimensions. Without loss of generality, we assume $d$ is superconstant and $k = o(d)$.

## 5.1 Spectral properties of PartialDuplicate

Let $v^*$ be the top unit eigenvector of $X^T X$. We can decompose $v^*$ into three components: the $x$ direction, the $y$ direction, and the component orthogonal to both of these. This is:

$$v^* = a\widehat{x} + b\widehat{y} + c\widetilde{w},$$

where $a^2 + b^2 + c^2 = 1$ and $\widetilde{w}$ is an arbitrary unit vector orthogonal to $x$ and $y$.

We have that

$$\begin{aligned}\|Xv^*\|^2 &= \|X'v^*\|^2+k\langle x,v^*\rangle^2+\langle x+y,v^*\rangle^2\\ &= \|X'v^*\|^2+a^2k\|x\|^2+(a\|x\|+b\|y\|)^2\\ &= \|X'v^*\|^2+a^2(k+1)\|x\|^2+2ab\|x\|\|y\|+b^2\|y\|^2\\ &= \|X'v^*\|^2+\frac{a^2(k+1)d}{2}+abd+\frac{b^2d}{2}. \end{aligned} \tag{18}$$

**Lemma 5.1.** *Suppose $n \le d$. Then $|c| \le O(1/k)$ with high probability.*

*Proof.* Since $\text{sign}(b)\cdot v^*$ is also a top eigenvector of $X^TX$, without loss of generality, we assume $b>0$. We consider unit vector $v' = \sqrt{a^2+c^2}\widehat{x}+b\widehat{y}$, we have

$$\begin{aligned}&\|Xv'\|^2-\|Xv^*\|^2\\ =&\|X'v'\|^2-\|X'v^*\|^2+\frac{c^2(k+1)d}{2}+(\sqrt{a^2+c^2}-a)bd\\ \ge&(v'+v^*)^TX'^TX'(v'-v^*)+\frac{c^2(k+1)d}{2}.\end{aligned}$$

By Lemma A.1, with high probability,

$$|(v'+v^*)^TX'^TX'(v'-v^*)| \le \left\|X'\right\|^2\left\|v'-v^*\right\|\left\|v'+v^*\right\| \le O(|cd|).$$

Thus,

$$\|Xv'\|^2-\|Xv^*\|^2 \ge -O(|cd|)+\frac{c^2(k+1)d}{2}.$$

Since $v^*$ is the top eigenvector, this implies that

$$-O(|cd|)+\frac{c^2(k+1)d}{2} \le 0.$$

Hence,

$$|c| \le O(1/k).$$

□

**Lemma 5.2.** *Suppose $k \ge C$ and $n \le \frac{d}{9k}$ for a sufficiently large constant $C$. Then $|b| \ge \frac{1}{3k}$ with high probability.*

*Proof.* Suppose $|b| < \frac{1}{3k}$ with non-negligible probability. Combining with Lemma 5.1, this implies that with non-negligible probability, $|b| < \frac{1}{3k}$ and $|c| \le O(1/k)$. We will show that with high probability, any unit vector $v = a\widehat{x}+b\widehat{y}+c\widetilde{w}$ satisfying $|b| < \frac{1}{3k}$ and $|c| < \frac{\log k}{k}$ is not the top eigenvector of $X^TX$. This contradicts the assumption and proves the lemma.

Without loss of generality, we only consider the case when $b>0$. Let $t := \sqrt{a^2+b^2} = \sqrt{1-c^2} > 2/3$. Therefore, we have $a = \sqrt{t^2-b^2} = t-\Theta(b^2)$. Taking this into (18), we have

$$\|Xv\|^2 = \|X'v\|^2+\frac{(t^2-b^2)kd}{2}+b(t-\Theta(b^2))d.$$

We consider vector $v' = \sqrt{t^2 - (b+\varepsilon)^2}\widehat{x} + (b+\varepsilon)\widehat{y} + c\widetilde{w}$ for $\varepsilon = \frac{1}{3k}$. Now we prove that with high probability $\|Xv'\|^2 - \|Xv\|^2 > 0$. We have

$$\begin{aligned}
\left\|Xv'\right\|^2 - \|Xv\|^2 &= \left\|X'v'\right\|^2 - \left\|X'v\right\|^2 + \frac{kd}{2}(b^2 - (b+\varepsilon)^2) + \varepsilon td + \Theta(b^3)d - \Theta((b+\varepsilon)^3)d \\
&= \left\|X'v'\right\|^2 - \left\|X'v\right\|^2 - bkd\varepsilon - \frac{kd\varepsilon^2}{2} + \varepsilon td + \Theta(b^3)d - \Theta((b+\varepsilon)^3)d \\
&\geq \left\|X'v'\right\|^2 - \left\|X'v\right\|^2 - \frac{d}{9k} - \frac{d}{18k} + \frac{d}{3k} \pm O(\frac{1}{k^3})d \\
&\geq -\left\|X'v\right\|^2 + \frac{d}{8k}
\end{aligned}$$

with our choice of $k$. Note that

$$\left\|X'v\right\|^2 \leq \left\|aX'\widehat{x}\right\|^2 + \left\|bX'\widehat{y}\right\|^2 + \left\|cX'\widetilde{w}\right\|^2 \leq \left\|X'\widehat{x}\right\|^2 + \frac{1}{9k^2}\left\|X'\widehat{y}\right\|^2 + \frac{\log^2 k}{k^2}\left\|X'\right\|^2.$$

By Lemma A.1, with high probability,

$$\left\|X'\right\| \leq 2\sqrt{d}.$$

Furthermore, since $\widehat{x}$ and $\widehat{y}$ are independent of $X'$, by Claim A.2, with high probability,

$$\left\|X'\widehat{x}\right\|^2 \leq n + o(n) \quad \text{and} \quad \left\|X'\widehat{y}\right\|^2 \leq n + o(n).$$

Therefore, with high probability,

$$\left\|X'v\right\|^2 \leq 1.1n < \frac{d}{8k}.$$

Hence, with high probability,

$$\left\|Xv'\right\|^2 - \|Xv\|^2 > 0.$$

□

**Lemma 5.3.** *Suppose $k \geq C$, $n \leq \frac{d}{9k}$ and $\varepsilon \leq \frac{1}{Ck^2}$ for a sufficiently large constant $C$. For any $\varepsilon$-approximate PCA solution $w$, $\langle w, y\rangle \geq \Omega(\sqrt{d}/k)$ with high probability.*

*Proof.* By Lemma 5.2, with high probability,

$$\langle v^*, y\rangle = \langle b\widehat{y}, y\rangle \geq \Omega(\sqrt{d}/k).$$

Therefore, for $w = v^* + \sqrt{\varepsilon}u$ for some unit vector $u$, we have

$$\langle w, y\rangle = \langle v^* + \sqrt{\varepsilon}u, y\rangle = \langle v^*, y\rangle + \sqrt{\varepsilon}\langle u, y\rangle \geq \Omega(\sqrt{d}/k) - \sqrt{\varepsilon}\,\|y\| \geq \Omega(\sqrt{d}/k).$$

□

**Lemma 5.4.** *Suppose $n \leq d$. The spectral gap $R$ is at least $k/20$ with high probability.*

*Proof.* The first eigenvalue $\lambda_1$ of $X^TX$ satisfies

$$\lambda_1 = \max_{\|v\|=1} \|Xv\|^2 \geq \|X\widehat{x}\|^2 \geq (k+1)\langle x, \widehat{x}\rangle^2 \geq \frac{kd}{2}.$$

The second eigenvalue $\lambda_2$ of $X^TX$ satisfies

$$\begin{aligned}\lambda_2 &= \min_{v} \max_{\substack{v' \perp v \\ \|v'\|=1}} \left\|Xv'\right\|^2 \\ &\leq \max_{\substack{v' \perp x \\ \|v'\|=1}} \left\|Xv'\right\|^2 \\ &= \max_{\substack{v' \perp x \\ \|v'\|=1}} \left(\left\|X'v'\right\|^2 + \langle y, v'\rangle^2\right) \\ &\leq \left\|X'\right\|^2 + \frac{d}{2}.\end{aligned}$$

By Lemma A.1 , with high probability, $\|X'\| \leq 3\sqrt{d}$. Therefore,

$$\lambda_2 \leq 10d.$$

Hence the spectral ratio

$$R = \frac{\lambda_1}{\lambda_2} \geq \frac{k}{20}.$$

□

## 5.2 Accuracy lower bound

**Theorem 1.4** (Accuracy Lower Bound)**.** *There exists a universal constant $C > 1$ such that: for any $R > 1$, $\frac{1}{CR^2}$-approximate PCA on streams with spectral gap $R$ requires at least $\frac{d^2}{CR^3}$ bits of space for sufficiently large $d > \text{poly}(R)$.*

*Proof.* Suppose that we have such an $\varepsilon$-approximate streaming PCA algorithm. We set up a two player one-way communication protocol. Let $A_1 \in \{-1, 1\}^{n \times \frac{d}{2}}$ and $A_2 \in \{-1, 1\}^{n \times \frac{d}{2}}$ be chosen uniformly at random. Let $A = [A_1, A_2] \in \{-1, 1\}^{n \times d}$ be their concatenation. Let $A' = [A_1, 0] \in \{0, -1, 1\}^{n \times d}$ be the matrix that pads $A_1$ to $d$ columns with 0.

In this protocol, Alice receives $A = [A_1, A_2]$ and Bob receives $A_1$. Alice feeds $A$ to the streaming algorithm, reaching some stream state $S$, which she sends to Bob. Bob uses $A_1$ and $S$ to construct an approximation $\widehat{A}$ to $A_2$ in the following fashion. For each $i \in [n]$, Bob sets the streaming algorithm's state to $S$, inserts the $i$-th row of $A'$ for $k$ times and computes the algorithm's approximate PCA solution $\widehat{v}_i$. Let $\widehat{V} \in \mathbb{R}^{n \times d}$ be the matrix with the $i$-th row being $\widehat{v}_i$. Let $\widehat{V}_2 \in R^{n \times \frac{d}{2}}$ be the last $d/2$ columns of $\widehat{V}$. We will show that $I(A_2; \widehat{V}) \gtrsim d^2/R^3$ for an appropriate choice of parameters.

Note that when Bob produces $\widehat{v}_i$, the streaming algorithm has effectively seen the stream $A$ followed by $k$ vectors that match the $i$th row of $A$. Up to reordering of rows, this is distributed identically to PartialDuplicate. Reordering the rows, of course, does not change the covariance matrix.

We choose $k = \max(20R, C)$, $n = \frac{d}{9k}$ and $\varepsilon = \frac{1}{Ck^2}$ for the constant $C$ in Lemma 5.3. By Lemma 5.4, with high probability the stream has spectral gap at least $k/20 \geq R$. Therefore the streaming algorithm's PCA solution should be $\varepsilon$-approximate with at least $2/3$ probability. Then Lemma A.3 says that

$$I(\widehat{V}; A_2) \geq \Omega\left(\frac{1}{k^2} \cdot \frac{d}{k} \cdot \frac{d}{2}\right) - d = \Omega\left(d^2/R^3\right).$$

Now, $\widehat{V}$ is independent of $A_2$ conditioned on $(S, A_1)$ so by the data processing inequality,

$$I(\widehat{V}; A_2) \leq I(A_1, S; A_2) \leq I(A_1; A_2) + I(S; A_2 \mid A_1) \leq 0 + H(S).$$

Thus, if the state $S$ contains $|S|$ bits, we have

$$\Omega(d^2/R^3) \le H(S) = H(|S|) + H(S \mid |S|) \le \mathbb{E}[|S|] + H(|S|)$$

Now, for any random variable $X$ over positive integers,

$$\begin{aligned} H(X) &= \sum_{i=1}^{\infty} p(i) \log \frac{1}{p(i)} \\ &= \left( \sum_{i:p(i)\le 2^{-i}} p(i) \log \frac{1}{p(i)} \right) + \left( \sum_{i:p(i)>2^{-i}} p(i) \log \frac{1}{p(i)} \right) \\ &\le \left( \sum_{i:p(i)\le 2^{-i}} 2^{-i} \cdot i \right) + \left( \sum_{i:p(i)>2^{-i}} ip(i) \right) \\ &= 2 + \mathbb{E}[X] \end{aligned}$$

so $\Omega(d^2/R^3) \le 2\,\mathbb{E}[|S|] + 2$, or

$$\mathbb{E}[|S|] \ge \Omega(d^2/R^3).$$

Thus the streaming algorithm must store $\Omega(d^2/R^3)$ bits on average after Alice has finished feeding in her part of the stream. □

## Acknowledgments

We thank David Woodruff and anonymous reviewers for helpful comments. Eric Price and Zhiyang Xun are supported by NSF award CCF-1751040 (CAREER) and the NSF AI Institute for Foundations of Machine Learning (IFML).

## References

[ACHPWY13] Pankaj K Agarwal, Graham Cormode, Zengfeng Huang, Jeff M Phillips, Zhewei Wei, and Ke Yi. "Mergeable summaries". In: *ACM Transactions on Database Systems (TODS)* 38.4 (2013), pp. 1–28.

[ACLS12] Raman Arora, Andrew Cotter, Karen Livescu, and Nathan Srebro. "Stochastic optimization for PCA and PLS". In: *2012 50th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*. IEEE. 2012, pp. 861–868.

[AL17] Zeyuan Allen-Zhu and Yuanzhi Li. "First efficient convergence for streaming k-PCA: a global, gap-free, and near-optimal rate". In: *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE. 2017, pp. 487–492.

[BDF13] Akshay Balsubramani, Sanjoy Dasgupta, and Yoav Freund. "The fast convergence of incremental PCA". In: *Advances in neural information processing systems* 26 (2013).

[BDGL15] Anja Becker, Léo Ducas, Nicolas Gama, and Thijs Laarhoven. *New directions in nearest neighbor searching with applications to lattice sieving*. Cryptology ePrint Archive, Paper 2015/1128. https://eprint.iacr.org/2015/1128. 2015. URL: https://eprint.

[BDWY16] Maria-Florina Balcan, Simon Shaolei Du, Yining Wang, and Adams Wei Yu. "An improved gap-dependency analysis of the noisy power method". In: *Conference on Learning Theory*. PMLR. 2016, pp. 284–309.

[BWZ16] Christos Boutsidis, David P Woodruff, and Peilin Zhong. "Optimal principal component analysis in distributed and streaming models". In: *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*. 2016, pp. 236–249.

[CW09] Kenneth L Clarkson and David P Woodruff. "Numerical linear algebra in the streaming model". In: *Proceedings of the forty-first annual ACM symposium on Theory of computing*. 2009, pp. 205–214.

[FS10] Ohad N Feldheim and Sasha Sodin. "A universality result for the smallest eigenvalues of certain sample covariance matrices". In: *Geometric And Functional Analysis* 20.1 (2010), pp. 88–123.

[GLPW16] Mina Ghashami, Edo Liberty, Jeff M Phillips, and David P Woodruff. "Frequent directions: Simple and deterministic matrix sketching". In: *SIAM Journal on Computing* 45.5 (2016), pp. 1762–1792.

[HNTW21] De Huang, Jonathan Niles-Weed, Joel A Tropp, and Rachel Ward. "Matrix concentration for products". In: *Foundations of Computational Mathematics* (2021), pp. 1–33.

[HNW21] De Huang, Jonathan Niles-Weed, and Rachel Ward. "Streaming k-PCA: Efficient guarantees for Oja's algorithm, beyond rank-one updates". In: *Conference on Learning Theory*. PMLR. 2021, pp. 2463–2498.

[HP14] Moritz Hardt and Eric Price. "The noisy power method: A meta algorithm with applications". In: *Advances in neural information processing systems* 27 (2014).

[JJKNS16] Prateek Jain, Chi Jin, Sham M Kakade, Praneeth Netrapalli, and Aaron Sidford. "Streaming PCA: Matching matrix bernstein and near-optimal finite sample guarantees for Oja's algorithm". In: *Conference on learning theory*. PMLR. 2016, pp. 1147–1164.

[JKDP21] A Jalal, S Karmalkar, A Dimakis, and E Price. "Instance-Optimal Compressed Sensing via Posterior Sampling". In: *International Conference on Machine Learning (ICML)*. 2021.

[JL84] William B. Johnson and Joram Lindenstrauss. "Extensions of Lipschitz mappings into Hilbert space". In: *Contemporary mathematics* 26 (1984), pp. 189–206. URL: https://api.semanticscholar.org/CorpusID:117819162.

[Joh01] Iain M Johnstone. "On the distribution of the largest eigenvalue in principal components analysis". In: *The Annals of statistics* 29.2 (2001), pp. 295–327.

[KS24] Syamantak Kumar and Purnamrita Sarkar. "Streaming PCA for Markovian Data". In: *Advances in Neural Information Processing Systems* 36 (2024).

[Lib13] Edo Liberty. "Simple and deterministic matrix sketching". In: *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*. 2013, pp. 581–588.

[LM00] B. Laurent and Pascal Massart. "Adaptive estimation of a quadratic functional by model selection". In: *Annals of Statistics* 28 (Oct. 2000). DOI: 10.1214/aos/1015957395.

[LSW21] Robert Lunde, Purnamrita Sarkar, and Rachel Ward. "Bootstrapping the error of Oja's Algorithm". In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 6240–6252.

[LW16] Yi Li and David P. Woodruff. "Tight Bounds for Sketching the Operator Norm, Schatten Norms, and Subspace Embeddings". In: *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2016)*. Ed. by Klaus Jansen, Claire Mathieu, José D. P. Rolim, and Chris Umans. Vol. 60. Leibniz International Proceedings in Informatics (LIPIcs). Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2016, 39:1–39:11. ISBN: 978-3-95977-018-7. DOI: 10.4230/LIPIcs.APPROX-RANDOM.2016.39. URL: https://drops.dagstu

[MCJ13] Ioannis Mitliagkas, Constantine Caramanis, and Prateek Jain. "Memory limited, streaming PCA". In: *Advances in neural information processing systems* 26 (2013).

[MV10] Daniele Micciancio and Panagiotis Voulgaris. "Faster Exponential Time Algorithms for the Shortest Vector Problem". In: *Proceedings of the Twenty-First Annual ACM-SIAM Symposium on Discrete Algorithms*. SODA '10. Austin, Texas: Society for Industrial and Applied Mathematics, 2010, pp. 1468–1480. ISBN: 9780898716986.

[Oja82] Erkki Oja. "Simplified neuron model as a principal component analyzer". In: *Journal of mathematical biology* 15.3 (1982), pp. 267–273.

[RV10] Mark Rudelson and Roman Vershynin. "Non-asymptotic theory of random matrices: extreme singular values". In: *Proceedings of the International Congress of Mathematicians 2010 (ICM 2010) (In 4 Volumes) Vol. I: Plenary Lectures and Ceremonies Vols. II–IV: Invited Lectures*. World Scientific. 2010, pp. 1576–1602.

[TYUC17] Joel A Tropp, Alp Yurtsever, Madeleine Udell, and Volkan Cevher. "Practical sketching algorithms for low-rank matrix approximation". In: *SIAM Journal on Matrix Analysis and Applications* 38.4 (2017), pp. 1454–1485.

[Upa18] Jalaj Upadhyay. "Fast and space-optimal low-rank factorization in the streaming model with application in differential privacy." In: *NeurIPS* (2018).

[Woo14a] David Woodruff. "Low rank approximation lower bounds in row-update streams". In: *Advances in neural information processing systems* 27 (2014).

[Woo14b] David P Woodruff. "Sketching as a tool for numerical linear algebra". In: *Foundations and Trends® in Theoretical Computer Science* 10.1–2 (2014), pp. 1–157.

# A Utility lemmas for the Lower Bounds

We use the following bound on the maximum singular value of an iid subgaussian matrix:

**Lemma A.1** (Feldheim and Sodin [FS10], see also (2.4) of [RV10])**.** *Let $A$ be an $n \times N$ random matrix with independent subgaussian entries of zero mean and variance 1, for $n \leq N$. There exists a universal constant $c > 0$ such that*

$$\Pr[\|A\| \geq \sqrt{n} + \sqrt{N} + \tau\sqrt{N}] \lesssim e^{-cn\tau^{3/2}}$$

*for any $\tau > 0$.*

The following is essentially a restatement of the JL lemma for $\pm 1$ matrices:

**Claim A.2.** *Let $u \in \mathbb{R}^d$ be a unit vector, and $X \in \{-1,1\}^{n\times d}$ independently and uniformly. Then*

$$\mathbb{E}[\|Xu\|^2] = n$$

*and with $1-\delta$ probability*

$$|\|Xu\|^2 - n| \lesssim \sqrt{n \log \frac{1}{\delta}} + \log \frac{1}{\delta}.$$

*Proof.* Let $z = Xu$. The coordinates $z_i$ are independent, mean zero, variance 1, and subgaussian with variance parameter 1. The expectation bound is trivial: sum the variance over $n$ independent coordinates. For concentration, each coordinate $z_i^2$ is a squared subgaussian, and hence subgamma with $(\sigma, c)$ parameters $(O(1), O(1))$. Then $\sum_i z_i^2$ is subgamma with parameters $(O(\sqrt{n}), O(1))$. Hence with probability $1-\delta$ we have

$$|\|Xu\|^2 - n| \lesssim \sqrt{n \log \frac{1}{\delta}} + \log \frac{1}{\delta}.$$

□

**Lemma A.3.** *Let $X \in \{-1,1\}^{n\times d}$ be uniformly distributed, and let $Y \in \mathbb{R}^{n\times d}$ have rows of norm at most 1 such that each row $i \in [n]$ has $|\langle x_i, y_i\rangle| > a\sqrt{d}$ with at least 50% probability, for $a > 0$. Then*

$$I(X;Y) \geq \Omega(a^2 nd) - n.$$

*Proof.* For any row $y$, when $x \in \{-1,1\}^d$ uniformly at random, $\langle x, y\rangle$ is subgaussian with variance parameter $\|y\|^2 \leq 1$, so

$$\Pr[|\langle x, y\rangle| > a\sqrt{d}] \leq 2e^{-a^2 d/2},$$

so the number of $x$ with $|\langle x, y\rangle| > a\sqrt{d}$ is at most $2^{(1-\Omega(a^2))d}$. Let $b \in \{0,1\}^n$ denote the indicator vector with $b_i = 1$ if $|\langle x_i, y_i\rangle| > a\sqrt{d}$ and $b_i = 0$ otherwise.

For any $Y, b$, let $S_{Y,b} \subseteq \{-1,1\}^{n\times d}$ be the set of possible $X$ that satisfy the inner product condition $|\langle x_i, y_i\rangle| > a\sqrt{d}$ for all rows $i \in [n]$ with $b_i = 1$. Each row with $b_i = 1$ has at most $2^{(1-\Omega(a^2))d}$ values of $x_i$ in the support, so

$$|S_{Y,b}| \leq 2^{nd - \Omega(a^2 \|b\|_1 d)}.$$

We have $\mathbb{E}[\|b\|_1] \geq \frac{n}{2}$, so

$$H(X \mid Y) \leq H(X \mid Y, b) + H(b) \leq (\mathop{\mathbb{E}}_{Y,b} \log |S_{Y,b}|) + n \leq (1 - \Omega(\frac{1}{2}a^2))nd + n$$

so

$$I(X;Y) = H(X) - H(X \mid Y) \geq \Omega(a^2 nd) - n.$$

□

**Claim A.4.** *Let $A, B > 0$. Then*

$$Aa^2 + Bab \leq \frac{a^2 + b^2}{2}(A + \sqrt{A^2 + B^2}),$$

*with equality if $\frac{a^2}{a^2+b^2} = \frac{1+\sqrt{\frac{A^2}{A^2+B^2}}}{2}$.*

*Proof.* Just ask a computer. By hand, though: the equations are homogeneous, so WLOG we can assume $a^2 + b^2 = 1$. We then maximize over $a \in [0, 1]$. Taking the derivative, the maximum is achieved when

$$2Aa + B(\sqrt{1-a^2} - \frac{a^2}{\sqrt{1-a^2}}) = 0$$

or

$$\begin{aligned} 2Aa\sqrt{1-a^2} &= B(2a^2-1) \\ 4A^2a^2(1-a^2) &= B^2(4a^4 - 4a^2 + 1) \\ a^4(4B^2+4A^2) - a^2(4A^2+4B^2) + B^2 &= 0 \\ a^2 &= \frac{1 \pm \sqrt{\frac{A^2}{A^2+B^2}}}{2} \end{aligned}$$

the first squaring preserved equality only when $a^2 \geq \frac{1}{2}$, so the optimum is at

$$a^2 = \frac{1 + \sqrt{\frac{A^2}{A^2+B^2}}}{2}.$$

Then

$$\begin{aligned} Aa^2 + Ba\sqrt{1-a^2} &= A\frac{1+\sqrt{\frac{A^2}{A^2+B^2}}}{2} + B\sqrt{\frac{1+\sqrt{\frac{A^2}{A^2+B^2}}}{2}\frac{1-\sqrt{\frac{A^2}{A^2+B^2}}}{2}} \\ &= A\frac{1+\sqrt{\frac{A^2}{A^2+B^2}}}{2} + B\sqrt{\frac{\frac{B^2}{A^2+B^2}}{4}} \\ &= \frac{1}{2}(A + \sqrt{A^2+B^2}). \end{aligned}$$

□

**Lemma A.5** (Laurent-Massart Bounds[LM00])**.** *Let $v \sim \mathcal{N}(0, I_n)$. For any $t > 0$,*

$$\Pr[\|v\|^2 - n \geq 2\sqrt{nt} + 2t] \leq e^{-t},$$

$$\Pr[\|v\|^2 - n \leq -2\sqrt{nt}] \leq e^{-t}.$$

**Lemma A.6** ([MV10], see also [BDGL15])**.** *Consider a $d$-dimensional unit sphere $S_d$. Let $C_h$ be a spherical cap on $S_d$ with height $h < 1$, i.e.,*

$$C_h := \{v \in S_d \mid \langle u, v\rangle \geq 1 - h\}$$

*for some $u \in S_d$. Then the ratio of the area of $C_h$ to the area of $S_d$ is given by $d^{\Theta(1)} \cdot (2h - h^2)^{d/2}$.*

# B Lower Bound for Linear Sketching

When establishing lower bounds for approximating operator norms using linear sketching, Li and Woodruff [LW16] constructed a lower bound instance with the following properties:

**Lemma B.1.** *For any $\alpha > 1.01$, there exist two distributions $\mathcal{D}_1$ and $\mathcal{D}_2$ over $\mathbb{R}^{d\times d}$ and $s > 0$ such that*

1. *For $X \sim \mathcal{D}_1$, $\|X\|_2 > \sqrt{\alpha}s$ with 0.99 probability.*
2. *For $X \sim \mathcal{D}_2$. $\|X\|_2 < s$ with 0.99 probability.*
3. *For $X \sim \mathcal{D}_1$, the spectral gap $\lambda_1(X^TX)/\lambda_2(X^TX)$ is at least $\alpha$ with 0.99 probability.*
4. *Let $\mathcal{L}_1$ and $\mathcal{L}_2$ be the corresponding distribution of the linear sketch of dimension $k$ on $\mathcal{D}_1$ and $\mathcal{D}_2$. Then $d_{TV}(\mathcal{L}_1, \mathcal{L}_2) < 0.1$ whenever $k \le o(d^2/\alpha^2)$.*

This implies a space lower bound for PCA using linear sketching. We present the Johnson-Lindenstrauss lemma first.

**Lemma B.2** (Johnson-Lindenstrauss Lemma [JL84])**.** *For any positive integer $d$ and $\varepsilon, \delta \in (0,1)$, there exists a distribution $\mathcal{S}$ over $\mathbb{R}^{m \times d}$ where $m = \Theta\left(\varepsilon^{-2} \log \frac{1}{\delta}\right)$ such that for every $x \in \mathbb{R}^d$,*

$$\Pr_{A \sim \mathcal{S}}\left[\left|\|Ax\|_2^2 - \|x\|_2^2\right| \le \varepsilon \|x\|_2^2\right] \ge 1 - \delta.$$

Using these lemmas, we can prove the following lower bound, which implies any sketching algorithm for adversarial streaming PCA needs at least $\Omega(d^2/R^2)$ bits of space.

**Theorem B.3.** *For all linear sketching algorithms, 0.1-approximate PCA on streams with spectral gap $R = o(\sqrt{d})$ requires sketches of dimension $\Omega(d^2/R^2)$,*

*Proof.* Let $\mathcal{D}_1$ and $\mathcal{D}_2$ be the distributions described in Lemma B.1 with $\alpha = R$. Suppose there exists a linear sketching algorithm that 0.1-approximates PCA using $o(d^2/R^2)$ space with success probability 0.99. We will show that this leads to a contradiction by constructing a linear sketch of dimension $o(d^2/R^2)$ that distinguishes $\mathcal{D}_1$ and $\mathcal{D}_2$ with 0.9 probability whenever $4 \le R \le o(\sqrt{d})$.

Let $\mathcal{S}$ be the distribution in Lemma B.2 with parameters $\delta = \varepsilon = 0.01$, and $\mathcal{S}$ is a distribution over $\mathbb{R}^{O(1) \times d}$. Let $s$ be the corresponding parameter for $\mathcal{D}_1$ and $\mathcal{D}_2$ in Lemma B.1. Our algorithm proceeds as follows: Given a matrix $X$, run the PCA approximation algorithm, which is a linear sketching of dimension $o(d^2/R^2)$, to obtain an approximation $\widetilde{v}$. In parallel, sample $A \sim \mathcal{S}$ and compute $AX$, which is a matrix of dimension $O(1) \times d$; that is, it is a linear sketch with $O(d) = o(d^2/R^2)$ dimensions. Suppose $\|AX\widetilde{v}\|_2 > 1.1s$, output that $X$ is from $\mathcal{D}_1$; otherwise, output that $X$ is from $\mathcal{D}_2$.

We first show that for $X \sim \mathcal{D}_1$, $\|AX\widetilde{v}\|_2 > 1.1s$ with 0.9 probability. Let $v^*$ be the true principal component of $X$ in the direction that $\langle \widetilde{v}, v^* \rangle \ge 0$. By a union bound, we have that with probability at least 0.96, the following events happen simultaneously:

1. $X$ has a spectral gap of $R$.
2. The PCA approximation $\widetilde{v}$ satisfies $\sin^2(\widetilde{v}, v^*) \le 0.1$.
3. $\|AX\widetilde{v}\|_2 \ge 0.99\|X\widetilde{v}\|_2$.
4. $\|X\|_2 > \sqrt{R}s$.

When all of these hold, we can prove that $\|AX\widetilde{v}\|_2 > 0.6\sqrt{R}s$. We have

$$\|AX\widetilde{v}\|_2 \ge 0.99\|X\widetilde{v}\|_2 \ge 0.99(\|Xv^*\|_2 - \|X(v^* - \widetilde{v})\|_2) \ge 0.99(\|X\|_2 - \|v^* - \widetilde{v}\|_2\|X\|_2).$$

Since $\widetilde{v}$ and $v^*$ are unit vectors, $\sin^2(\widetilde{v}, v^*) \le 0.1$ implies that $\|v^* - \widetilde{v}\|_2 = \sqrt{2 - 2\cos(\widetilde{v}, v^*)} \le 0.35$. Thus,

$$\|AX\widetilde{v}\|_2 \ge 0.6\|X\|_2 > 0.6\sqrt{R}s.$$

Therefore, $\|AX\widetilde{v}\|_2 > 1.1s$ with probability at least 0.9 whenever $R \geq 4$.

Next, we show that for $X \sim \mathcal{D}_2$, $\|AX\widetilde{v}\|_2 \leq 1.1s$ with 0.9 probability. Again, by a union bound, we have with at least 0.9 probability,

$$\|AX\widetilde{v}\|_2 \leq 1.01\|X\widetilde{v}\|_2 \leq 1.01\|X\|_2 \leq 1.1s.$$

This proves that our sketching algorithm distinguishes $\mathcal{D}_1$ and $\mathcal{D}_2$ with probability at least 0.9, contradicting Lemma B.1, and therefore proves the theorem. □