Approximating the Top Eigenvector in Random Order Streams

Praneeth Kacham[*]
Google Research
pkacham@google.com

David P. Woodruff
Carnegie Mellon University
dwoodruf@cs.cmu.edu

Abstract

When rows of an $n \times d$ matrix A are given in a stream, we study algorithms for approximating the top eigenvector of the matrix $A^\mathsf{T}A$ (equivalently, the top right singular vector of A). We consider worst case inputs A but assume that the rows are presented to the streaming algorithm in a uniformly random order. We show that when the gap parameter $R = \sigma_1(A)^2/\sigma_2(A)^2 = \Omega(1)$, then there is a randomized algorithm that uses $O(h \cdot d \cdot \mathrm{polylog}(d))$ bits of space and outputs a unit vector v that has a correlation $1 - O(1/\sqrt{R})$ with the top eigenvector v_1. Here h denotes the number of *heavy rows* in the matrix, defined as the rows with Euclidean norm at least $\|A\|_\mathsf{F}/\sqrt{d \cdot \mathrm{polylog}(d)}$. We also provide a lower bound showing that any algorithm using $O(hd/R)$ bits of space can obtain at most $1 - \Omega(1/R^2)$ correlation with the top eigenvector. Thus, parameterizing the space complexity in terms of the number of heavy rows is necessary for high accuracy solutions.

Our results improve upon the $R = \Omega(\log n \cdot \log d)$ requirement in a recent work of Price and Xun (FOCS 2024). We note that the algorithm of Price and Xun works for arbitrary order streams whereas our algorithm requires a stronger assumption that the rows are presented in a uniformly random order. We additionally show that the gap requirements in their analysis can be brought down to $R = \Omega(\log^2 d)$ for arbitrary order streams and $R = \Omega(\log d)$ for random order streams. The requirement of $R = \Omega(\log d)$ for random order streams is nearly tight for their analysis as we obtain a simple instance with $R = \Omega(\log d/\log\log d)$ for which their algorithm, with any fixed learning rate, cannot output a vector approximating the top eigenvector v_1.

1 Introduction

We consider the problem of approximating the top eigenvector in the streaming setting. In this problem, we are given vectors $a_1, \ldots, a_n \in \mathbb{R}^d$ one at a time in a stream. Let A be an $n \times d$ matrix with rows $a_1, \ldots, a_n$. The task is to approximate the top eigenvector of the matrix $A^\mathsf{T}A$. Throughout the paper, we use $v_1 \in \mathbb{R}^d$ to denote the top eigenvector of $A^\mathsf{T}A$. We focus on obtaining streaming algorithms that use a small amount of space and can output a unit vector $\hat{v}$ such that $\langle \hat{v}, v_1 \rangle^2 \geq 1 - f(R)$, where $f(R)$ is a decreasing function in the gap $R = \lambda_1(A^\mathsf{T}A)/\lambda_2(A^\mathsf{T}A)$. Here $\lambda_1(\cdot), \lambda_2(\cdot)$ denote the two largest eigenvalues. As the gap R becomes larger, the eigenvector approximation problem becomes easier and we want more accurate approximations to the eigenvector v_1.

If one is allowed to use $\tilde{O}(d^2)$[2] bits of space, we can maintain the matrix $A^\mathsf{T}A = \sum_i a_i a_i^\mathsf{T}$ as we see the rows a_i in the stream, and at the end of processing the stream, we can compute the exact top eigenvector v_1. When the dimension d is large, the requirement of $\Omega(d^2)$ bits of memory can be

[*]Work done while the author was a student at Carnegie Mellon University.

[2]The notation $\tilde{O}(f(n))$ is used to denote the set of functions in $O(f(n) \cdot \mathrm{polylog}(n))$.

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

impractical (see e.g., applications that require a large value of d in Mitliagkas et al. (2013).) Hence, an interesting question is to study non-trivial streaming algorithms that use less memory. In this work, we focus on obtaining algorithms that use $\tilde{O}(d)$ bits of space.

In the offline setting (where the entire matrix A is available to us), fast iterative algorithms such as Gu (2015); Musco and Musco (2015); Musco et al. (2018) can be used to quickly obtain accurate approximations to the top eigenvector when the gap $R = \Omega(1)$. In a single pass streaming setting, we cannot run these algorithms as these iterative algorithms need to *see* the entire matrix multiple times.

There have been two major lines of work studying the problem of eigenvector approximation and the related Principal Component Analysis (PCA) problem in the streaming setting with near-linear in d memory. In the first line of work, each row encountered in the stream is assumed to be sampled independently from an unknown distribution with mean 0 and covariance Σ and the task is to approximate the top eigenvector of Σ using the samples. In this line of work, the sample complexity required for algorithms using $O(d \cdot \mathrm{polylog}(d))$ bits of space to output an approximation to v_1, is the main question. The algorithms are usually a variant of Oja's algorithm (Oja, 1982; Jain et al., 2016; Allen-Zhu and Li, 2017; Huang et al., 2021; Kumar and Sarkar, 2023) or the block power method (Hardt and Price, 2014; Balcan et al., 2016). We note that Kumar and Sarkar (2023) relax the i.i.d. assumption and analyze the sample complexity of Oja's algorithm for estimating the top eigenvector in the Markovian data setting.

The other line of work studies algorithms for arbitrary streams appearing in an arbitrary order. In this setting, we want algorithms to work for *any* input stream given in *any* order. A problem closely related to the eigenvector estimation problem is the Frobenius-norm Low Rank Approximation (Clarkson and Woodruff, 2017; Boutsidis et al., 2016; Upadhyay, 2016; Ghashami et al., 2016). The deterministic Frequent Directions sketch of Ghashami et al. (2016) can, using $\tilde{O}(d/\varepsilon)$ bits of space, output a unit vector u such that

$$\|A(I - uu^{\mathsf{T}})\|_{\mathsf{F}}^2 \leq (1+\varepsilon)\|A(I - v_1 v_1^{\mathsf{T}})\|_{\mathsf{F}}^2.$$

Although the vector u is a $1+\varepsilon$ approximate solution to the Frobenius norm Low Rank Approximation problem, it is possible that the vector u may be (nearly) orthogonal to the top eigenvector v_1. Hence the Frequent Directions sketch does not guarantee top eigenvector approximation. Recently, Price and Xun (2024) study the eigenvector approximation problem in arbitrary streams and obtain results in terms of the gap R of the instance. Price and Xun prove that when $R = \Omega(\log n \cdot \log d)$, a variant of Oja's algorithm outputs a unit vector $\hat{v}$ such that

$$\langle \hat{v}, v_1 \rangle^2 \geq 1 - \frac{C \log d}{R} - \frac{1}{\mathrm{poly}(d)}$$

where C is a large enough universal constant. On the lower bound side, Price and Xun showed that any algorithm that outputs a vector $\hat{v}$ satisfying

$$\langle \hat{v}, v_1 \rangle^2 \geq 1 - \frac{1}{CR^2},$$

must use $\Omega(d^2/R^3)$ bits of space while processing the stream. This lower bound shows that in the important case of $R = O(1)$, the *correlation*[3] that can be obtained by an algorithm using $\tilde{O}(d)$ bits of space is at most a constant less than 1. Thus, the current best algorithms for arbitrary streams work only when $R = \Omega(\log n \cdot \log d)$ and for the important case of $R = O(1)$, there are no existing algorithms requiring significantly fewer than d^2 bits of memory. They also give a lower bound on the size of *mergeable* summaries for approximating the top eigenvector.

We identify an instance with $R = \Theta(\log d / \log \log d)$ where the algorithm of Price and Xun fails to produce a vector with even a constant correlation with the vector v_1. This shows that their algorithm or other variants of Oja's algorithm may fail to extend to the case when $R = O(1)$. We further show that the algorithm of Price and Xun fails to produce such a vector even when the rows in our hard instance are ordered uniformly at random, showing that even randomly ordered streams can be hard to solve for variants of Oja's algorithm.

In this work, we focus on algorithms that work on worst case inputs A while assuming that the rows of A are *uniformly randomly ordered*. This model is mid-way between the i.i.d. setting and the

[3] We say that the value $\langle u, v \rangle^2$ denotes the correlation between unit vectors u and v.

arbitrary order stream setting in terms of the generality of streams that can be modeled. We note that a number of works (Munro and Paterson, 1980; Guha et al., 2005; Chakrabarti et al., 2008; Guha and McGregor, 2009; Assadi and Sundaresan, 2023) have previously considered streaming algorithms and lower bounds for worst case inputs with random order streams, as it is a natural model often arising in practical settings. Our algorithms are parameterized in terms of the number of **heavy** rows in the stream. See Gupta and Singla (2021) for a gentle introduction to the random-order model. We define a row a_i to be *heavy* if $\|a_i\|_2 \geq \|A\|_F/\sqrt{d \cdot \text{polylog}(d)}$. Note that in any stream of rows, by definition, there are at most $d \cdot \text{polylog}(d)$ heavy rows. We state our theorem informally below:

Theorem 1.1. *Let $a_1, \ldots, a_n \in \mathbb{R}^d$ be a randomly ordered stream and let A denote the $n \times d$ matrix with rows given by $a_1, \ldots, a_n$. If $R = \lambda_1(A^\mathsf{T}A)/\lambda_2(A^\mathsf{T}A) > C$ for a large enough constant C and the number of heavy rows in the stream is at most h, then there is a streaming algorithm using $O(h \cdot d \cdot \text{polylog}(d))$ bits of space and outputting a unit vector $\hat{v}$ satisfying*

$$\langle \hat{v}, v_1 \rangle^2 \geq 1 - O(1/\sqrt{R})$$

with a probability $\geq 4/5$.

Our algorithm is a variant of the block power method. Along the way, we also improve the gap requirements in the results of Price and Xun (2024). We show that by subsampling a stream of rows, the algorithm of Price and Xun can be made to work even when the gap R is $\Omega(\log^2 d)$ in arbitrary order streams, improving on the $\Omega(\log n \cdot \log d)$ requirement in their analysis. We also show that in random order streams, a gap of $\Omega(\log d)$ is sufficient for their algorithm, though our algorithm improves on this and works for even a constant gap.

Similar to the lower bound of Price and Xun, we show that any algorithm for random order streams must use $\Omega(h \cdot d/R)$ bits of space to output a vector $\hat{v}$ satisfying $\langle \hat{v}, v_1 \rangle^2 \geq 1 - 1/CR^2$ where C is a constant. We summarize the theorem below.

Theorem 1.2. *Consider an arbitrary random order stream $a_1, \ldots, a_n$ with the gap parameter $\frac{\sigma_1(A)^2}{\sigma_2(A)^2} = R$. Let h be the number of* heavy *rows in the stream. Any streaming algorithm that outputs a unit vector $\hat{v}$ such that*

$$\langle \hat{v}, v_1 \rangle^2 \geq 1 - 1/CR^2$$

for a large enough constant C, with a probability $\geq 1 - (1/2)^{R+1}$ over the ordering of the stream and its internal randomness, must use $\Omega(h \cdot d/R)$ bits of space.

Techniques. The randomized power method (Gu, 2015) algorithm to approximate the top eigenvector samples a random Gaussian vector $\boldsymbol{g}$ and iteratively computes the vector $v = (A^\mathsf{T}A)^t \boldsymbol{g}$[4] for $t = \Theta(\log d)$ iterations and shows that when the gap R is large, $v/\|v\|_2$ is a good approximation for v_1. Thus, the algorithm needs to *see* the quadratic form $A^\mathsf{T}A$ multiple times and hence, it cannot be implemented in the single-pass streaming setting of this paper.

Assume that the stream is randomly ordered and that there are no heavy rows. Our key observation is that if the stream is long enough, then we can see t approximations $\boldsymbol{B}_j^\mathsf{T}\boldsymbol{B}_j$[5] of the quadratic form $A^\mathsf{T}A$. Here the matrices $\boldsymbol{B}_1, \ldots, \boldsymbol{B}_t$ are formed by sampling and rescaling the rows of the matrix A and importantly, the rows of $\boldsymbol{B}_1, \ldots, \boldsymbol{B}_t$ do not overlap in the stream, that is, they appear one after the other. Thus we can compute $v' = (\boldsymbol{B}_t^\mathsf{T}\boldsymbol{B}_t) \cdots (\boldsymbol{B}_1^\mathsf{T}\boldsymbol{B}_1) \cdot \boldsymbol{g}$ for the starting vector $\boldsymbol{g}$ in a single pass over the stream. We prove that such matrices $\boldsymbol{B}_j$ exist using the row norm sampling result of Magdon-Ismail (2010). Now, the main issue is to show that $v'/\|v'\|_2$ is a good approximation to the top eigenvector v_1. We crucially use a singular value inequality of Wang and Xi (1997) to prove that $\|\boldsymbol{B}_j^\mathsf{T}\boldsymbol{B}_j - A^\mathsf{T}A\|_2 \leq \varepsilon\|A\|_2^2$ for all j suffices for $v'/\|v'\|_2$ to be a good approximation to v_1.

The above analysis assumes that there are no heavy rows. Indeed, suppose that a matrix A has a row a with a large Euclidean norm which is orthogonal to all the other rows. Also assume that the top eigenvector of the matrix A is in this direction. Since, the matrices $\boldsymbol{B}_1, \ldots, \boldsymbol{B}_t$ are non-overlapping substreams of the matrix A, at most one of the matrices $\boldsymbol{B}_j$ can have the row a and hence the vector $v'/\|v'\|_2$ will not be a good approximation to $a/\|a\|_2$, the top eigenvector. Thus, we need to handle

[4] Note that $A^\mathsf{T}A \cdot v = \sum_i \langle a_i, v \rangle a_i$.

[5] We use bold symbols to denote random variables.

the heavy rows separately. We show that, by storing all the rows with a Euclidean norm larger than $\|A\|_{\mathsf{F}}/\sqrt{d \cdot \mathrm{polylog}(d)}$ and running the above described algorithm on the remaining set of rows, we can obtain a good approximation to the top eigenvector.

Our lower bound (Theorem 1.2) shows that any single-pass streaming algorithm must use space proportional to the number of heavy rows, and therefore our procedure that handles the heavy rows separately gives near-optimal bounds.

Finally, the row norm sampling technique of Magdon-Ismail (2010) serves as a general technique to reduce the number of rows in the stream while (approximately) preserving the top eigenvector. We use this observation to improve the $R = \Omega(\log n \cdot \log d)$ for arbitrary streams in Price and Xun (2024) to $R = \Omega(\log^2 d)$. We then show that assuming a uniformly random order, the analysis of Price and Xun (2024) can be improved to show that $R = \Omega(\log d)$ suffices. Thus, for random order streams, techniques before our work can be used to approximate the top eigenvector when the gap $R = \Omega(\log d)$. Our work improves upon this to give an algorithm that works for streams with $R = \Omega(1)$.

Implications to practice. Often, in practical situations, we can assume that the rows being streamed are sampled independently from a nice-enough distribution, in which case Oja's algorithm, as discussed, can approximate the top eigenvector accurately given enough samples. However, *independence* and assumptions on the covariance matrix can be very strong assumptions in some cases and in such cases, our algorithm only requires that the order of the rows in the stream be uniformly random, in which case we output an approximation with provable guarantees.

Organization. We first introduce the row-norm sampling procedure to obtain approximate quadratic forms. The proof is a slight modification of that of Magdon-Ismail (2010). The only difference is that we instead consider a version that samples each row in the input independently with some appropriate probability and keeps the rows that are sampled after scaling appropriately. We then introduce and analyze our block power iteration algorithm when all rows have roughly the same Euclidean norm, and then extend it to the general case, which is our main result. Finally, we provide a lower bound showing that $\Omega(td/R)$ bits of space is necessary to obtain constant correlation with the top eigenvector. Due to space constraints, all of our proofs are placed in the appendix.

2 Power Method with Approximate Quadratic Forms

In this section, we present and analyze our algorithm for approximating the top eigenvector of $A^{\mathsf{T}}A$ when the rows of A are presented to the algorithm in a uniformly random order.

We first show a row sampling technique that reduces the number of rows in the stream. The row-norm sampling technique for approximating the quadratic form $A^{\mathsf{T}}A$ with spectral norm guarantees was given by Magdon-Ismail (2010). The technique works irrespective of the order of the rows.

2.1 Sampling for Row Reduction

Theorem 2.1. *Let A be an arbitrary $n \times d$ matrix. Given $p \in [0,1]^n$, let $\boldsymbol{Q}$ be an $n \times n$ diagonal matrix such that for each $i \in [n]$, we independently set $\boldsymbol{Q}_{ii} = 1/\sqrt{p_i}$ with probability p_i and 0 otherwise. If for all i,*

$$p_i \geq \min\left(1, C\frac{\|a_i\|_2^2}{\varepsilon^2\|A\|_2^2}\log d\right),$$

then with probability $1 - 1/\operatorname{poly}(d)$, $\|A^{\mathsf{T}}A - A^{\mathsf{T}}\boldsymbol{Q}^{\mathsf{T}}\boldsymbol{Q}A\|_2 \leq \varepsilon\|A\|_2^2$. With probability at least $1 - 1/\operatorname{poly}(d)$, the matrix $\boldsymbol{Q}$ has at most $O(\varepsilon^{-2}\rho\log d)$ non-zero entries, where $\rho = \|A\|_{\mathsf{F}}^2/\|A\|_2^2$ denotes the stable rank of matrix A.

Note that given the value of $\|A\|_2$, the sampling procedure in this theorem can be performed in a stream. Additionally, as the original stream is uniformly randomly ordered, the sub-sampled stream is also uniformly randomly ordered assuming that the sampling is independent of the order of the rows.

Given that all of the non-zero entries of the matrix have absolute value at least $1/\operatorname{poly}(nd)$ and at most $\operatorname{poly}(nd)$, we have that $\|A\|_2^2$ lies in the interval $[1/\operatorname{poly}(nd), \operatorname{poly}(nd)]$. Thus, we can guess the value of $\|A\|_2^2$ as $2^i/\operatorname{poly}(nd)$ for $i = 0, \ldots, O(\log(nd))$ and one of these values must be a 2-approximation to $\|A\|_2^2$, and thus sub-sampling the rows using that guess satisfies the conditions in the above theorem. We can run the streaming algorithms on all the streams simultaneously to obtain $O(\log nd)$ vectors $u_1, \ldots, u_{O(\log nd)}$ as the candidates for being an approximation to the top eigenvector. From Theorem 2.1, the candidate vector u_j computed on the stream obtained by sampling the rows with the correct probabilities is a good approximation to the top eigenvector, and therefore $\|A \cdot u_j\|_2$ is large for that value of j. Thus, the vector u_j with the largest value $\|A \cdot u_j\|_2$ is a good approximation to the top eigenvector v_1. If $\boldsymbol{G}$ is a Gaussian matrix with $O(\varepsilon^{-2} \log d)$ rows, then for all u_j, we can approximate $\|A \cdot u_j\|_2$ up to a $1 \pm \varepsilon$ factor using $\|\boldsymbol{G} \cdot A \cdot u_j\|_2$ by the Johnson-Lindenstrauss lemma. Additionally, the matrix $\boldsymbol{G} \cdot A$ can be maintained in the stream using $O(\varepsilon^{-2} \cdot d \log d)$ bits (when we see a row a_i, we sample an independent Gaussian vector $\boldsymbol{g}_i$ and add $\boldsymbol{g}_i a_i^\mathsf{T}$ to an accumulator to maintain $\boldsymbol{G} \cdot A$). Thus, at the end of processing the stream, we can compute a vector u_j that has a large value $\|A \cdot u_j\|_2$, and hence is a good approximation for v_1.

If we can process each created stream using s bits of space, then the overall space requirement is $O(s \cdot \log(nd) + d \cdot \operatorname{polylog}(d))$ bits, using $O(s)$ bits for each guess for the value of $\|A\|_2^2$ and $O(d \cdot \operatorname{polylog}(d))$ bits for storing a Gaussian sketch of the matrix with $\varepsilon = 1/\operatorname{polylog}(d)$.

2.2 Random-Order Streams with bounds on Norms

Algorithm 1: Approximate Eigenvector for Streams with no Large Norms

Input: An $n \times d$ matrix A with $n = \Omega(\eta \cdot \rho(A) \cdot \log^2 d/\varepsilon^2)$, $\max_i \|a_i\|_2^2 / \min_i \|a_i\|_2^2 \le \eta$
Output: A vector $\boldsymbol{z}$

$t \leftarrow \lceil C_1 \log d \rceil$
Compute $\boldsymbol{G} \cdot A$ in the stream where $\boldsymbol{G}$ is a Gaussian matrix with $O(\varepsilon^{-2} \log d)$ rows
for $\rho = 1, 2, 4, \ldots, d$ *simultaneously* **do**
  $p \leftarrow C_2 \eta \rho \log d / n\varepsilon^2$ // $p \le 1/(5t)$ for $\rho \le 2 \cdot \rho(A)$
  $\boldsymbol{z}_\rho \sim N(0,1)^d$
  for $j = 1, \ldots, t$ **do**
    $\boldsymbol{y}_j \leftarrow \mathrm{Bin}(n, p)$
    if $\boldsymbol{y}_j > 2np$ **then**
      return $\perp$
    end
    // The matrix $A_{j\cdot(2np):j\cdot(2np)+\boldsymbol{y}_j}$ corresponds to $\boldsymbol{B}_j$ in the analysis.
    $acc \leftarrow 0$
    for $i = (j-1)\cdot(2np)+1, \ldots, (j-1)\cdot(2np) + \boldsymbol{y}_j$ **do**
      $acc \leftarrow acc + \langle a_i, \boldsymbol{z}_\rho \rangle \cdot a_i$
    end
    // Here $acc = \boldsymbol{B}_j^\mathsf{T} \boldsymbol{B}_j \boldsymbol{z}_\rho$
    $\boldsymbol{z}_\rho \leftarrow acc$
    $\boldsymbol{z}_\rho \leftarrow \boldsymbol{z}_\rho / \|\boldsymbol{z}_\rho\|_2$
  end
end
return $\arg\max_{\boldsymbol{z} \in \{\boldsymbol{z}_1, \boldsymbol{z}_2, \boldsymbol{z}_4, \ldots, \boldsymbol{z}_d\}} \|(\boldsymbol{G} \cdot A)\boldsymbol{z}\|_2$

We now present the analysis of the block power method for random order streams assuming that the Euclidean norms of all the rows in A are close to each other. We later remove this assumption. Suppose there exists a parameter η such that $(\max_i \|a_i\|_2^2)/(\min_i \|a_i\|_2^2) \le \eta$. If η is close to 1 then all the rows in the stream have roughly the same norm.

Let $p = C\eta\rho \log(d)/\varepsilon^2 n$. We can see that for any row a_i in the stream,

$$C \frac{\|a_i\|_2^2}{\varepsilon^2 \|A\|_2^2} \log d \le C \frac{\eta \|A\|_\mathsf{F}^2/n}{\varepsilon^2 \|A\|_2^2} \log d \le \frac{C\eta\rho \log d}{n\varepsilon^2} = p.$$

Thus, p is greater than the probability with which we need to sample each row in the row-norm sampling result in Theorem 2.1. Now if we perform such a sampling of the rows of A, we sample $\mathrm{Bin}(n,p)$[6] number of rows, which is tightly concentrated around $np = \varepsilon^{-2} C\eta\rho \log d$. Thus, if we first sample $\boldsymbol{y} \sim \mathrm{Bin}(n,p)$ and then consider the first $\boldsymbol{y}$ number of rows in the random order stream, then we will have sampled from a distribution satisfying the requirements in Theorem 2.1 and can therefore obtain a matrix $\boldsymbol{B}$ such that

$$\|\boldsymbol{B}^\mathsf{T}\boldsymbol{B} - A^\mathsf{T}A\|_2 \le \varepsilon \|A\|_2^2.$$

Thus, assuming that the rows appear in a uniformly random order lets us show that the first $\boldsymbol{y}$ rows of the stream can be used to compute an approximation to the quadratic form $A^\mathsf{T}A$. We will now show that we can obtain $O(\log d)$ such quadratic forms in the stream given that the stream is long enough.

Assume that the number of rows in the stream $n = \Omega(\eta\rho \log^2 d/\varepsilon^2)$. We partition the stream into $t = \Theta(\log d)$ groups as follows: the first $2np$ rows are placed in the group 1, the second $2np$ rows are placed in the group 2, and so on. Note that since $n = \Omega(\eta\rho \log^2 d/\varepsilon^2)$, we can form t such groups. Since the rows are uniformly randomly ordered, the joint distribution of the rows appearing in group 1 is the same as that of the joint distribution of the rows appearing in group 2 and so on. Let $\boldsymbol{y}_1, \dots, \boldsymbol{y}_t \sim \mathrm{Bin}(n,p)$ be drawn independently. With probability $\ge 1 - 1/\operatorname{poly}(d)$, we have $\boldsymbol{y}_i \le (3/2)np$ for all i. For $i = 1, \dots, t$, let $\boldsymbol{B}_i$ be the matrix formed by the first $\boldsymbol{y}_i$ rows in group i. Using a union bound, we have that with probability $\ge 1 - 1/\operatorname{poly}(d)$, for all $i = 1, \dots, t$,

$$\|A^\mathsf{T}A - \frac{1}{p}\boldsymbol{B}_i^\mathsf{T}\boldsymbol{B}_i\|_2 \le \varepsilon \|A\|_2^2.$$

Conditioned on the above event, we will now show that running the power method on the blocks $\boldsymbol{B}_1, \dots, \boldsymbol{B}_t$ lets us approximate the top singular vector of the matrix A.

Assumption 2.2. We assume that $\sigma_1(A)/\sigma_2(A) \ge 2$.

Lemma 2.3. *Let $\varepsilon > 1/\operatorname{poly}(d)$ be an accuracy parameter and $t = \Omega(\log d)$ be the number of iterations. Let $\varepsilon \le c/t^2$ for a small constant c. Suppose $B_1, \dots, B_t$ all satisfy $\|A^\mathsf{T}A - B_j^\mathsf{T}B_j\|_2 \le \varepsilon\|A\|_2^2$ for $\varepsilon < 1/5$. If $\boldsymbol{g}$ is a random vector sampled from the Gaussian distribution, then the unit vector*

$$\hat{v} := \frac{(B_t^\mathsf{T}B_t)\cdots(B_1^\mathsf{T}B_1)\boldsymbol{g}}{\|(B_t^\mathsf{T}B_t)\cdots(B_1^\mathsf{T}B_1)\boldsymbol{g}\|_2}$$

satisfies

$$\langle \hat{v}, v_1\rangle^2 \ge \frac{1}{1 + C' t\sqrt{\varepsilon}}$$

with probability $\ge 9/10$ for a large enough constant C'. Here v_1 denotes the top right singular vector of the matrix A.

To prove this lemma, our strategy is to show that the matrix product $M := (B_t^\mathsf{T}B_t)\cdots(B_1^\mathsf{T}B_1)$ has a stable rank close to 1 — meaning it has one very large singular value and the rest of the singular values are small. We can then argue that the vector $\hat{v} = M\boldsymbol{g}/\|M\boldsymbol{g}\|_2$ is in the direction of the top singular vector M. Using the fact that $v_1^\mathsf{T}(B_j^\mathsf{T}B_j)v_1 \ge (1-\varepsilon)\|A\|_2^2$ for all j, we show that the top singular vector of M must have a large correlation with v_1. Therefore, it follows that the vector $\hat{v}$ has a large correlation with v_1 as well. As part of the proof, we crucially use an inequality from Wang and Xi (1997).

If $t = \Theta(\log d)$ and $1/\operatorname{poly}(d) \le \varepsilon \le c/(\log d)^2$, then the above lemma shows that $\hat{v}$ has a large correlation with the top singular vector v_1. Using this lemma, we show that Algorithm 1 can be used to obtain an approximation for v_1 in random order streams with bounded norms.

Theorem 2.4. *Let $\alpha \ge 1/\operatorname{poly}(d)$ be an accuracy parameter. Let η be a parameter such that $\frac{\max_i \|a_i\|_2^2}{\min_i \|a_i\|_2^2} \le \eta$. If the number of rows in the stream $n = \Omega(\alpha^{-4} \cdot \rho(A) \cdot \eta \cdot \log^6 d)$, where $\rho(A) =$*

[6] $\mathrm{Bin}(n,p)$ denotes the binomial distribution with parameters n and p.

$\|A\|_{\mathsf{F}}^2/\|A\|_2^2$ and the rows in the stream are ordered uniformly at random, then we can compute a vector $\hat{v}$ using the block power method that satisfies

$$|\langle v_1, \hat{v}\rangle|^2 \geq 1 - 3\alpha$$

with probability $\geq 4/5$ if $\sigma_1(A)/\sigma_2(A) \geq 2$. The algorithm uses $O(d \cdot \mathrm{polylog}(d)/\alpha^4)$ bits of space.

Proof. Set $\varepsilon = \alpha^2/C\log^2 d$ for a large enough constant C. Assuming $n = \Omega(\alpha^{-4}\rho\eta\log^6 d)$, we have $n = \Omega(\varepsilon^{-2}\rho\eta\log^2 d)$. Now consider the execution of Algorithm 1 on matrix A, with parameters η and ε. Let $\rho = 2^j$ be such that $\rho(A)/2 \leq \rho \leq \rho(A)$, and consider the execution in the algorithm with parameter ρ. Using Theorem 2.1, with probability $\geq 1 - 1/\mathrm{poly}(d)$, the algorithm computes t matrices $\boldsymbol{B}_1, \ldots, \boldsymbol{B}_t$ such that for all $j \in [t]$,

$$\|\frac{1}{p}\boldsymbol{B}_j^{\mathsf{T}}\boldsymbol{B}_j - A^{\mathsf{T}}A\|_2 \leq \varepsilon\|A\|_2^2.$$

Noting that $\boldsymbol{z}_\rho = (\boldsymbol{B}_t^{\mathsf{T}}\boldsymbol{B}_t)\cdots(\boldsymbol{B}_1^{\mathsf{T}}\boldsymbol{B}_1)\boldsymbol{g}/\|(\boldsymbol{B}_t^{\mathsf{T}}\boldsymbol{B}_t)\cdots(\boldsymbol{B}_1^{\mathsf{T}}\boldsymbol{B}_1)\boldsymbol{g}\|_2$, by Lemma 2.3, we have with probability $\geq 9/10$ that

$$\langle \boldsymbol{z}_\rho, v_1\rangle^2 \geq \frac{1}{1 + C't\sqrt{\varepsilon}} \geq 1 - \alpha.$$

Thus, for ρ which satisfies $\rho(A)/2 \leq \rho \leq \rho(A)$, the algorithm computes a vector $\boldsymbol{z}_\rho$ that has a large correlation with the vector v_1. Since the algorithm does not know the exact value of ρ, it computes an approximation for $\|A\boldsymbol{z}\|_2^2$ for all $\boldsymbol{z} \in \{\boldsymbol{z}_1, \boldsymbol{z}_2, \boldsymbol{z}_4, \ldots, \boldsymbol{z}_d\}$. First, we condition on the fact that with probability $\geq 1 - 1/\mathrm{poly}(d)$, for all $\boldsymbol{z}_i$, $\|\boldsymbol{G}A\boldsymbol{z}_i\|_2^2 = (1 \pm \varepsilon)\|A\boldsymbol{z}_i\|_2^2$. Since $\langle \boldsymbol{z}_\rho, v_1\rangle^2 \geq (1-\alpha)$, we note that $\|\boldsymbol{G}A\boldsymbol{z}_\rho\|_2^2 \geq (1-\varepsilon)(1-\alpha)\sigma_1(A)^2$. Now, for the vector $\boldsymbol{z}$ returned by the algorithm, we have $\|\boldsymbol{A}\boldsymbol{z}\|_2^2 \geq (1 - O(\varepsilon))(1-\alpha)\sigma_1(A)^2$ which implies that

$$\langle \boldsymbol{z}, v_1\rangle^2 \cdot \sigma_1(A)^2 + (1 - \langle \boldsymbol{z}, v_1\rangle^2)\frac{\sigma_1(A)^2}{R} \geq \|A\boldsymbol{z}\|_2^2 \geq (1 - \alpha - O(\varepsilon))\sigma_1(A)^2$$

and therefore $\langle \boldsymbol{z}, v_1\rangle^2 \geq 1 - 3\alpha$ since $R \geq 2$. □

2.3 Random Order Streams without Norm Bounds

Assuming that the random order streams are long enough, Theorem 2.4 shows that if all the squared row norms are within an η factor, then the block power method outputs a vector with a large correlation with the top eigenvector of the matrix $A^{\mathsf{T}}A$. For general streams, the factor η could be quite large and hence the algorithm requires very long streams to output an approximation to v_1.

If there are no *heavy* rows, i.e., rows with a Euclidean norm larger than $\|A\|_{\mathsf{F}}/\sqrt{d \cdot \mathrm{polylog}(d)}$, then the row norm sampling procedure in Theorem 2.1 can be used to convert any randomly ordered stream of rows into a uniformly random stream of rows that all have the same norm. The row norm sampling procedure computes a probability $p_i = \min(1, C\varepsilon^{-2}\|a_i\|_2^2 \log d/\|A\|_2^2)$ and samples the row a_i with probability p_i. If sampled, then the row a_i is scaled by $1/\sqrt{p_i}$. From Theorem 2.1, we have that the top eigenvector of the *quadratic form* of the sampled-and-rescaled submatrix is a good approximation to the top eigenvector $A^{\mathsf{T}}A$ when the gap R is large enough. Suppose $p_i < 1$. If the row a_i is sampled, we then have

$$\|a_i/\sqrt{p_i}\|_2 = \frac{\varepsilon\|A\|_2}{\sqrt{C\log d}}.$$

Thus, if $p_i < 1$ for all i, then all the sampled-and-rescaled rows have the same Euclidean norm and therefore, we can run the algorithm from Theorem 2.4 by setting $\eta = 1$. Note that $p_i = 1$ only if $\|a_i\|_2^2 \geq \varepsilon^2\|A\|_2^2/C\log(d)$. Since we assumed that there are no heavy rows, there is no row with $p_i = 1$ as long as $\varepsilon \geq 1/\mathrm{polylog}(d)$. Thus, using Theorem 2.4 on the row norm sampled substream directly gives us a good approximation to the top eigenvector. However, in general, the streams can have rows with large Euclidean norm. We will now state our theorem and describe how such streams can be handled.

Theorem 2.5. *Let A be an $n \times d$ matrix with its non-zero entries satisfying $1/\operatorname{poly}(d) \le |A_{i,j}| \le \operatorname{poly}(d)$, and hence representable using $O(\log d)$ bits of precision. Let $R = \sigma_1(A)^2/\sigma_2(A)^2$. Assume $2 \le R \le C_1 \log^2 d$. Let h be the number of rows in A with norm at most $\|A\|_F/\sqrt{d \cdot \operatorname{polylog}(d)}$, where $\operatorname{polylog}(d) = \log^{C_2} d$ for a large enough universal constant C_2. Given the rows of the matrix A in a uniformly random order, there is an algorithm using $O((h+1)\cdot d \cdot \operatorname{polylog}(d) \cdot \log n)$ bits of space and which outputs a vector $\hat{v}$ such that with probability $\ge 4/5$, $\hat{v}$ satisfies $\langle \hat{v}, v_1 \rangle^2 \ge 1 - 8/\sqrt{R}$, where v_1 is the top eigenvector of the matrix $A^\mathsf{T}A$.*

The key idea in proving this theorem is to partition the matrix A into A_{heavy} and A_{light}, where A_{heavy} denotes the matrix with the heavy rows and A_{light} denotes the matrix with the rest of the rows of A. Since we assume that there are at most h heavy rows, we can store the matrix A_{heavy} using $O(h \cdot d \cdot \operatorname{polylog}(d))$ bits of space. Now consider the following two cases: (i) $\|A_{\text{heavy}}\|_2 \ge (1-\beta)\|A\|_2$ or (ii) $\|A_{\text{heavy}}\|_2 < (1-\beta)\|A\|_2$ for some parameter β. In the first case, we can show that the top eigenvector u of $A_{\text{heavy}}^\mathsf{T} A_{\text{heavy}}$ is a good approximation for v_1. Since, we store the full matrix A_{heavy}, we can compute u exactly at the end of the stream. Suppose $\|A_{\text{heavy}}\|_2 < (1-\beta)\|A\|_2$. By the triangle inequality, we have $\|A_{\text{light}}\|_2 > \beta\|A\|_2$. If we set β large enough compared to $1/R$, then we can show that the top eigenvector u' of $A_{\text{light}}^\mathsf{T} A_{\text{light}}$ is a good approximation of v_1. From the above discussion, since all the rows of A_{light} are *light*, we can obtain a stream using Theorem 2.1 such that all the rows have the same norm and additionally, the top eigenvector of this stream is a good approximation for u' and therefore v_1. We then approximate the top eigenvector of the new stream using Theorem 2.4. Setting β appropriately, we show that this procedure can be used to compute a vector $\hat{v}$ satisfying $\langle \hat{v}, v_1 \rangle^2 \ge 1 - O(1/\sqrt{R})$ proving the theorem.

3 Lower Bounds

Our algorithm uses $\tilde{O}(h \cdot d)$ space when the number of heavy rows in the stream is h. We want to argue that it is nearly tight. We show the following theorem.

Theorem 3.1. *Given a dimension d, let h and R be arbitrary with $R \le h \le d$ and $R^2 \cdot h = O(d)$. Consider an algorithm $\mathcal{A}$ with the following property:*

Given any fixed matrix $n \times d$ matrix A with $O(h)$ heavy rows and gap $\sigma_1(A)^2/\sigma_2(A)^2 \ge R$, in the form of a uniform random order stream, the algorithm $\mathcal{A}$ outputs a unit vector $\hat{v}$ such that, with probability $\ge 1 - (1/2)^{4R+4}$ over the randomness of the stream and the internal randomness of the algorithm, $|\langle \hat{v}, v_1 \rangle|^2 \ge 1 - c/R^2$.

If c is a small enough constant, then the algorithm $\mathcal{A}$ must use $\Omega(h \cdot d/R)$ bits of space.

The theorem shows that a streaming algorithm must use $\Omega(hd/R)$ bits of space assuming that with high probability, it outputs a vector with a large enough correlation with the top eigenvector of $A^\mathsf{T}A$ when the rows are given in a random order stream.

Our proof uses the same lower bound instance as that of Price and Xun (2024). The key difference from their proof is that our lower bound must hold against random order streams.

4 Improving the Gap Requirements in the Algorithm of Price and Xun

4.1 Arbitrary Order Streams

As discussed in Section 2.1, we can guess an approximation of $\|A\|_2^2$ in powers of 2 and sample at most $O(d \log d/\varepsilon^2)$ rows in the stream to obtain a matrix $\boldsymbol{B}$, in the form of a stream, satisfying $\|\boldsymbol{B}^\mathsf{T}\boldsymbol{B} - A^\mathsf{T}A\|_2 \le \varepsilon\|A\|_2^2$, with a large probability. Using Weyl's inequalities, we obtain that

$$\sigma_2(\boldsymbol{B}^\mathsf{T}\boldsymbol{B}) \le \sigma_2(A^\mathsf{T}A) + \varepsilon\|A\|_2^2 \quad \text{and} \quad \sigma_1(\boldsymbol{B}^\mathsf{T}\boldsymbol{B}) \ge (1-\varepsilon)\sigma_1(A^\mathsf{T}A)$$

implying $R' = \sigma_1(\boldsymbol{B})^2/\sigma_2(\boldsymbol{B})^2 \ge (1-\varepsilon)/(1/R+\varepsilon)$. For $\varepsilon = 1/(2R) \le 1/2$, we note $R' \ge R/3$. Let $n' = O(R^2 \cdot d \log d)$ be the number of rows in the matrix $\boldsymbol{B}$ and note that $R' = \Omega(\log n' \cdot \log d)$ assuming $R = \Omega(\log^2 d)$. Hence, running the algorithm of Price and Xun on the rows of the matrix

$\boldsymbol{B}$, we compute a vector $\hat{v}$ for which

$$|\langle \hat{v}, v_1' \rangle|^2 \geq 1 - \frac{\log d}{CR'} - \frac{1}{\text{poly}(d)}$$

with a large probability, where v_1' is the top eigenvector of the matrix $\boldsymbol{B}^\mathsf{T}\boldsymbol{B}$. We now note that if v_1 denotes the top eigenvector of the matrix $A^\mathsf{T}A$, then $|\langle v_1, v_1' \rangle|^2 \geq 1 - O(1/R)$ which therefore implies that with a large probability,

$$|\langle \hat{v}, v_1 \rangle|^2 \geq 1 - \frac{\log d}{CR}.$$

Thus, sub-sampling the stream using row norm sampling and then running the algorithm of Price and Xun (2024), we obtain an algorithm for arbitrary order streams with a gap $R = \Omega(\log^2 d)$.

4.2 Random Order Streams

Lemma 3.5 in Price and Xun (2024) can be tightened when the rows of the stream are uniformly randomly ordered. Specifically, we want to bound the following quantity:

$$\sum_{i=1}^{n} \langle a_i, P\hat{v}_{i-1} \rangle^2$$

where $P = I - v_1 v_1^\mathsf{T}$ denotes the projection away from the top eigenvector, and $\hat{v}_{i-1}$ is a function of $v_1, a_1, \ldots, a_{i-1}$. We have

$$\mathbf{E}[\langle a_i, P\hat{v}_{i-1} \rangle^2] = \mathbf{E}[\mathbf{E}[\langle a_i, P\hat{v}_{i-1} \rangle^2 \mid a_1, \ldots, a_{i-1}]].$$

Given that the first $i-1$ rows are $a_1, \ldots, a_{i-1}$, assuming uniform random order, we have

$$\begin{aligned}\mathbf{E}[\langle a_i, P\hat{v}_{i-1} \rangle^2 \mid a_1, \ldots, a_{i-1}] &= \frac{1}{n-i+1} \hat{v}_{i-1}^\mathsf{T} P(A^\mathsf{T}A - a_1 a_1^\mathsf{T} - \cdots - a_{i-1}a_{i-1}^\mathsf{T})P\hat{v}_{i-1} \\ &\leq \frac{\sigma_2(A)^2}{n-i+1}.\end{aligned}$$

Hence $\mathbf{E}[\langle a_i, P\hat{v}_{i-1} \rangle^2] \leq \sigma_2(A)^2/(n-i+1)$ and $\mathbf{E}[\sum_{i=1}^{n} \langle a_i, P\hat{v}_{i-1} \rangle^2] \leq \sigma_2(A)^2(1+\log n)$. Price and Xun define $\eta \cdot \sigma_2(A)^2$ as σ_2 and in that notation, we obtain $\eta \sum_{i=1}^{n} \langle a_i, P\hat{v}_{i-1} \rangle^2 \leq 10\sigma_2(1 + \log n)$ with probability $\geq 9/10$ by Markov's inequality. In the proof of Lemma 3.6 in Price and Xun (2024), if $\sigma_1/\sigma_2 \geq 20(1 + \log_2 n)$, we obtain $\log \|v_n\|_2 \gtrsim \sigma_1$. Now, $\sigma_1 \geq O(\log d)$ ensures that the Proof of Theorem 1.1 in their work goes through.

Using the row-norm sampling analysis from the previous section, we can assume $n = \text{poly}(d)$ and therefore a gap of $O(\log d)$ between the top two eigenvalues of $A^\mathsf{T}A$ is enough for Oja's algorithm to output a vector with a large correlation with the top eigenvector in random order streams.

5 Hard Instance for Oja's Algorithm

At a high level, the algorithm of Price and Xun (2024) runs Oja's algorithm with different learning rates η and in the event that the norm of the output vector with each of the learning rates η is small, then the row with the largest norm is output. The algorithm is simple and can be implemented using an overall space of $O(d \cdot \text{polylog}(d))$ bits.

The algorithm initializes $z_0 = \boldsymbol{g}$ where $\boldsymbol{g}$ is a random Gaussian vector. The algorithm streams through the rows $a_1, \ldots, a_n$ and performs the following operation

$$z_i \leftarrow z_{i-1} + \eta \cdot \langle z_{i-1}, a_i \rangle a_i.$$

The algorithm computes the smallest learning rate η when $\|z_n\|_2$ is large enough, and then outputs either $z_n/\|z_n\|_2$ or $\bar{a}/\|\bar{a}\|_2$ as an approximation to the eigenvector of the matrix $A^\mathsf{T}A$. Here $\bar{a}$ denotes the row in A with the largest Euclidean norm.

The following theorem shows that at gaps $\leq O(\log d/\log\log d)$, we cannot use Oja's algorithm with a fixed learning rate η to obtain constant correlation with the top eigenvector.

Theorem 5.1. *Given dimension d, a constant $c > 0$, a parameter M, for all gap parameters $R = O_c(\log d/\log\log d)$ there is a stream of vectors $a_1, \ldots, a_n \in \mathbb{R}^d$ with $n = O(R + M)$ such that:*

1. *$\sigma_1(A)^2/\sigma_2(A)^2 \geq R/2$, and*
2. *Oja's algorithm with any learning rate $\eta < M$ fails to output a unit vector $\hat{v}$ that satisfies, with probability $\geq 9/10$,*

$$|\langle \hat{v}, v_1 \rangle| \geq c$$

where v_1 is the top eigenvector of the matrix $A^\mathsf{T}A$.

Moreover, the result holds irrespective of the order in which the vectors $a_1, \ldots, a_n$ are presented to the Oja's algorithm. We will additionally show that even keeping track of the largest norm vector is insufficient to output a vector that has a large correlation with v_1.

Acknowledgements

The authors were supported in part by a Simons Investigator Award and NSF CCF-2335412. D. Woodruff was visiting Google Research while performing this work.

References

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)*, pages 487–492. IEEE, 2017. 2

Sepehr Assadi and Janani Sundaresan. (Noisy) gap cycle counting strikes back: Random order streaming lower bounds for connected components and beyond. In *Proceedings of the 55th Annual ACM Symposium on Theory of Computing*, pages 183–195, 2023. 3

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*, pages 284–309. PMLR, 2016. 2

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*, pages 236–249, 2016. 2

Amit Chakrabarti, Graham Cormode, and Andrew McGregor. Robust lower bounds for communication and stream computation. In *Proceedings of the fortieth annual ACM symposium on Theory of computing*, pages 641–650, 2008. 3

Kenneth L Clarkson and David P Woodruff. Low-rank approximation and regression in input sparsity time. *Journal of the ACM (JACM)*, 63(6):1–45, 2017. 2

Mina Ghashami, Edo Liberty, Jeff M Phillips, and David P Woodruff. Frequent directions: Simple and deterministic matrix sketching. *SIAM Journal on Computing*, 45(5):1762–1792, 2016. 2

Ming Gu. Subspace iteration randomization and singular value problems. *SIAM Journal on Scientific Computing*, 37(3):A1139–A1173, 2015. 2, 3

Sudipto Guha and Andrew McGregor. Stream order and order statistics: Quantile estimation in random-order streams. *SIAM Journal on Computing*, 38(5):2044–2059, 2009. 3

Sudipto Guha, Andrew McGregor, and Suresh Venkatasubramanian. Streaming and sublinear approximation of entropy and information distances. *arXiv preprint cs/0508122*, 2005. 3

Anupam Gupta and Sahil Singla. Random-order models. In Tim Roughgarden, editor, *Beyond the Worst-Case Analysis of Algorithms*, chapter 11. Oxford University Press, 2021. doi: 10.1017/9781108637435. URL https://arxiv.org/pdf/2002.12159. 3

Moritz Hardt and Eric Price. The noisy power method: A meta algorithm with applications. *Advances in neural information processing systems*, 27, 2014. 2

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*, pages 2463–2498. PMLR, 2021. 2

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*, pages 1147–1164. PMLR, 2016. 2

Syamantak Kumar and Purnamrita Sarkar. Streaming pca for markovian data. In *Advances in Neural Information Processing Systems*, volume 36, 2023. 2

Malik Magdon-Ismail. Row sampling for matrix algorithms via a non-commutative bernstein bound. *arXiv preprint arXiv:1008.0587*, 2010. 3, 4

Ioannis Mitliagkas, Constantine Caramanis, and Prateek Jain. Memory-limited, streaming PCA. In *Advances in Neural Information Processing Systems*, volume 26, 2013. 2

J Ian Munro and Mike S Paterson. Selection and sorting with limited storage. *Theoretical computer science*, 12(3):315–323, 1980. 3

Cameron Musco and Christopher Musco. Randomized block krylov methods for stronger and faster approximate singular value decomposition. *Advances in neural information processing systems*, 28, 2015. 2

Cameron Musco, Christopher Musco, and Aaron Sidford. Stability of the lanczos method for matrix function approximation. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1605–1624. SIAM, 2018. 2

Erkki Oja. Simplified neuron model as a principal component analyzer. *Journal of mathematical biology*, 15:267–273, 1982. 2

Eric Price and Zhiyang Xun. Spectral guarantees for adversarial streaming PCA. In *FOCS*, 2024. 2, 3, 4, 8, 9, 15

Joel A Tropp. An introduction to matrix concentration inequalities. *Foundations and Trends® in Machine Learning*, 8(1-2):1–230, 2015. 12

Jalaj Upadhyay. Fast and space-optimal low-rank factorization in the streaming model with application in differential privacy. *arXiv preprint arXiv:1604.01429*, 2016. 2

Bo-Ying Wang and Bo-Yan Xi. Some inequalities for singular values of matrix products. *Linear algebra and its applications*, 264:109–115, 1997. 3, 6, 13

A Omitted Proofs

A.1 Proof of Theorem 2.1

Proof. Let $\boldsymbol{X}_i$ denote an indicator random variable which denotes if $\boldsymbol{Q}_{ii}$ is nonzero. Note $\mathbf{E}[\boldsymbol{X}_i] = p_i$ and $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n$ are independent. Define a $d \times d$ random matrix $\boldsymbol{Y}_i = (\boldsymbol{X}_i/p_i - 1)a_i a_i^\mathsf{T}$, where a_i denotes the i-th row of A. We note that

$$A^\mathsf{T}A - A^\mathsf{T}\boldsymbol{Q}^\mathsf{T}\boldsymbol{Q}A = \sum_{i=1}^{n}(\boldsymbol{X}_i/p_i - 1)a_i a_i^\mathsf{T} = \sum_{i=1}^{n}\boldsymbol{Y}_i.$$

We use the Matrix Bernstein inequality (Tropp, 2015) to bound $\|\sum_i \boldsymbol{Y}_i\|_2$. We first uniformly upper bound $\|\boldsymbol{Y}_i\|_2$. If $p_i = 1$, by definition $\|\boldsymbol{Y}_i\|_2 = 0$ with probability 1. Let $p_i \neq 0$. Then, $\|(\boldsymbol{X}_i/p_i - 1)a_i a_i^\mathsf{T}\|_2 \leq \|a_i a_i^\mathsf{T}\|_2/p_i \leq \varepsilon^2\|A\|_2^2/C\log d$ with probability 1.

We now bound $\|\sum_i \mathbf{E}[\boldsymbol{Y}_i^2]\|_2$.

$$\begin{aligned}
\sum_i \mathbf{E}[\boldsymbol{Y}_i^2] &= \sum_i \mathbf{E}[(1/p_i - 1)^2]\|a_i\|_2^2 a_i a_i^\mathsf{T} \\
&= \sum_{i:p_i>0}(1/p_i - 1)\|a_i\|_2^2 a_i a_i^\mathsf{T} \\
&\preceq \sum_{i:p_i>0}\frac{\varepsilon^2\|A\|_2^2}{C\|a_i\|_2^2\log d}\|a_i\|_2^2 a_i a_i^\mathsf{T} \\
&\preceq \frac{\varepsilon^2\|A\|_2^2}{C\log d}A^\mathsf{T}A
\end{aligned}$$

which implies $\|\sum_i \mathbf{E}[\boldsymbol{Y}_i^2]\|_2 \leq \varepsilon^2\|A\|_2^4/(C\log d)$. Now, we obtain

$$\begin{aligned}
\mathbf{Pr}[\|\sum_i \boldsymbol{Y}_i\|_2 \geq \varepsilon\|A\|_2^2] &\leq 2d\cdot\exp\left(-\frac{\varepsilon^2\|A\|_2^4/2}{\varepsilon^2\|A\|_2^4/(C\log d) + \varepsilon^3\|A\|_2^4/(3C\log d)}\right) \\
&\leq 2d\cdot\exp\left(-\frac{C\log d}{2(1+\varepsilon/3)}\right).
\end{aligned}$$

If $C \geq 6(1+\varepsilon/3)$, then $\mathbf{Pr}[\|\sum_i \boldsymbol{Y}_i\|_2 \geq \varepsilon\|A\|_2^2] \leq 1 - 2/d^2$ which implies that with probability $\geq 1 - 2/d^2$, $\|A^\mathsf{T}A - A^\mathsf{T}\boldsymbol{Q}^\mathsf{T}\boldsymbol{Q}A\|_2 \leq \varepsilon\|A\|_2^2$.

Now, the number of non-zero entries in the matrix $\boldsymbol{Q}$ is equal to $\sum_i \boldsymbol{X}_i$. We note $\mathbf{E}[\sum_i \boldsymbol{X}_i] \leq C\varepsilon^{-2}\rho\cdot\log d$. By a Chernoff bound, we obtain that $\sum_i \boldsymbol{X}_i = O(\varepsilon^{-2}\rho\cdot\log d)$ with probability $\geq 1 - 1/\operatorname{poly}(d)$. □

A.2 Proof of Lemma 2.3

Proof. Define $M \coloneqq (B_t^\mathsf{T}B_t)\cdots(B_1^\mathsf{T}B_1)$. Our strategy is to show that if v_1 is the top singular vector of the matrix A, then $\|v_1^\mathsf{T}M\|_2$ is comparable to $\|M\|_\mathsf{F}$ given that $\sigma_1(A)/\sigma_2(A) \geq 2$. We can then prove the lemma using simple properties of the Gaussian vector $\boldsymbol{g}$.

For an arbitrary j, let $(B_j^\mathsf{T}B_j)v_1 = \alpha v_1 + \Delta$ where $\Delta \perp v_1$. We note that $v_1^\mathsf{T}(B_j^\mathsf{T}B_j)v_1 = \alpha$. We have $\alpha = v_1^\mathsf{T}B_j^\mathsf{T}B_j v_1 \geq (1-\varepsilon)\sigma_1(A)^2$ using the fact that $\|B_j^\mathsf{T}B_j - A^\mathsf{T}A\|_2 \leq \varepsilon\|A\|_2^2$ and $v_1^\mathsf{T}A^\mathsf{T}Av_1 = \sigma_1(A)^2 = \|A\|_2^2$. If we show that Δ is small, then the vector $(B_j^\mathsf{T}B_j)v_1$ is oriented in a direction very close to that of v_1. Note that

$$\|(B_j^\mathsf{T}B_j)v_1\|_2 \leq \|B_j^\mathsf{T}B_j\|_2 \leq (1+\varepsilon)\sigma_1(A)^2$$

and $\|(B_j^\mathsf{T}B_j)v_1\|_2^2 = \alpha^2 + \|\Delta\|_2^2$ which implies $\|\Delta\|_2^2 \leq ((1+\varepsilon)^2 - (1-\varepsilon)^2)\sigma_1(A)^4 = 4\varepsilon\cdot\sigma_1(A)^4$ and thus $\|\Delta\|_2 \leq \sqrt{4\varepsilon}\sigma_1(A)^2$. Now,

$$\begin{aligned}
&\|M^\mathsf{T}v_1\|_2 \\
&= \|(B_1^\mathsf{T}B_1)\cdots(B_{t-1}^\mathsf{T}B_{t-1})(\langle B_t^\mathsf{T}B_t v_1, v_1\rangle v_1 + \Delta_1)\|_2 \\
&\geq \langle B_t^\mathsf{T}B_t v_1, v_1\rangle\|(B_1^\mathsf{T}B_1)\cdots(B_{t-1}^\mathsf{T}B_{t-1})v_1\|_2 - \|(B_1^\mathsf{T}B_1)\cdots(B_{t-1}^\mathsf{T}B_{t-1})\|_2\|\Delta_1\|_2 \\
&\geq ((1-\varepsilon)\sigma_1(A)^2)\|(B_1^\mathsf{T}B_1)\cdots(B_{t-1}^\mathsf{T}B_{t-1})v_1\|_2 - (\sqrt{4\varepsilon}\sigma_1(A)^2)\|(B_1^\mathsf{T}B_1)\cdots(B_{t-1}^\mathsf{T}B_{t-1})\|_2.
\end{aligned}$$

Expanding similarly, we obtain

$$\|M^\mathsf{T} v_1\|_2 \geq (1-\varepsilon)^t \sigma_1(A)^{2t} - t\sqrt{4\varepsilon}(1+\varepsilon)^{t-1}\sigma_1(A)^{2t}.$$

Assuming $\varepsilon \leq c/t$ for a small constant c, we note that $(1-\varepsilon)^t \geq (1-2t\varepsilon)$ and $(1+\varepsilon)^t \leq (1+2t\varepsilon)$ which implies

$$\|M^\mathsf{T} v_1\|_2 = \|(B_1^\mathsf{T} B_1)\cdots(B_t^\mathsf{T} B_t)v_1\|_2 \geq (1-2t\varepsilon - 4t\sqrt{\varepsilon})\sigma_1(A)^{2t}.$$

We shall now show a bound on $\|M\|_\mathsf{F} = \|(B_t^\mathsf{T} B_t)\cdots(B_1^\mathsf{T} B_1)\|_\mathsf{F}$ which lets us show that the unit vector $\hat{v}$ is highly correlated with v_1. To bound the quantity $\|M\|_\mathsf{F}$, we first note the following facts:

1. $\|B_j^\mathsf{T} B_j\|_2 \leq (1+\varepsilon)\sigma_1(A)^2$, and
2. $\sigma_2(B_j^\mathsf{T} B_j) \leq \sigma_2(A)^2 + \varepsilon\sigma_1(A)^2 \leq (1/4+\varepsilon)\sigma_1(A)^2$ by our gap assumption.

Now, we use the following theorem.

Theorem A.1 ((Wang and Xi, 1997, Theorem 3(ii))). *For any $r > 0$ and any matrices $A_1, \ldots, A_t$,*

$$\sum_i (\sigma_i(A_1 \cdots A_t))^r \leq \sum_i \sigma_i(A_1)^r \cdots \sigma_i(A_t)^r.$$

Applying the above theorem with $r = 2$, we obtain

$$\begin{aligned}\|(B_t^\mathsf{T} B_t)\cdots(B_1^\mathsf{T} B_1)\|_\mathsf{F}^2 &\leq (1+\varepsilon)^{2t}\sigma_1(A)^{4t} + (d-1)(1/4+\varepsilon)^t \sigma_1(A)^{4t} \\ &\leq (1+4t\varepsilon)\sigma_1(A)^{4t} + \frac{d}{3^t}\sigma_1(A)^{4t}.\end{aligned}$$

When $t \geq 3\log(d/\varepsilon)$, we have $\|(B_t^\mathsf{T} B_t)\cdots(B_1^\mathsf{T} B_1)\|_\mathsf{F}^2 \leq (1+4t\varepsilon+\varepsilon)\sigma_1(A)^{4t}$. We now use the following lemma.

Lemma A.2. *Let $\boldsymbol{g}$ be a Gaussian random vector with each of the components being an independent standard Gaussian random variable. Let $\hat{v} = M\boldsymbol{g}/\|M\boldsymbol{g}\|_2$. For any unit vector v, with probability $\geq 4/5$,*

$$|\langle \hat{v}, v\rangle|^2 \geq \frac{1}{1 + C\frac{\|M\|_\mathsf{F}^2 - \|M^\mathsf{T} v\|_2^2}{\|M^\mathsf{T} v\|_2^2}}$$

for a large enough universal constant C.

Proof. Since v is a unit vector, we can write $\|M\boldsymbol{g}\|_2^2 = |v^\mathsf{T} M\boldsymbol{g}|^2 + \|(I - vv^\mathsf{T})M\boldsymbol{g}\|_2^2$. Hence, we have

$$|\langle \hat{v}, v\rangle|^2 = \frac{|v^\mathsf{T} Mg|^2}{\|M\boldsymbol{g}\|_2^2} = \frac{1}{1 + \frac{\|(I-vv^\mathsf{T})M\boldsymbol{g}\|_2^2}{|v^\mathsf{T} M\boldsymbol{g}|^2}}.$$

We now note that $v^\mathsf{T} M\boldsymbol{g} \sim N(0, \|M^\mathsf{T} v\|_2^2)$ and $\mathbf{E}[\|(I-vv^\mathsf{T})M\boldsymbol{g}\|_2^2] = \mathrm{tr}(M^\mathsf{T}(I-vv^\mathsf{T})M) = \|M\|_\mathsf{F}^2 - \|M^\mathsf{T} v\|_2^2$. By a union bound, with probability $\geq 4/5$, we have

$$\frac{\|(I-vv^\mathsf{T})M\boldsymbol{g}\|_2^2}{|v^\mathsf{T} M\boldsymbol{g}|^2} \leq C\frac{\|M\|_\mathsf{F}^2 - \|M^\mathsf{T} v\|_2^2}{\|M^\mathsf{T} v\|_2^2}$$

for a large enough constant C. Therefore, with probability $\geq 4/5$, we get that

$$|\langle \hat{v}, v\rangle|^2 \geq \frac{1}{1 + C\frac{\|M\|_\mathsf{F}^2 - \|M^\mathsf{T} v\|_2^2}{\|M^\mathsf{T} v\|_2^2}}.$$

□

Applying the above lemma for $M = (B_t^\mathsf{T} B_t)\cdots(B_1^\mathsf{T} B_1)$ and $v = v_1$, we obtain

$$|\langle \hat{v}, v_1\rangle|^2 \geq \frac{1}{1 + C't\sqrt{\varepsilon}}$$

with probability $\geq 4/5$. □

A.3 Proof of Theorem 2.5

Proof. Partition the matrix A into A_{light} and A_{heavy}, where A_{heavy} is the submatrix with rows a_i such that $\|a_i\|_2 > \|A\|_{\mathsf{F}}/\sqrt{d \cdot \text{polylog}(d)}$ and A_{light} is the remaining rows. From our assumption, the number of rows in A_{heavy} is at most h. Note that given a uniformly random stream of rows of A, we can obtain a uniformly random stream of rows of A_{light} by just filtering out the rows in A_{heavy}.

Suppose, $\|A_{\text{heavy}} \cdot v_1\|_2 \geq (1-\beta)\|A\|_2$ for a parameter β to be chosen later. Let v_1' be the top singular vector of the matrix A_{heavy}. Note

$$\|A \cdot v_1'\|_2^2 \geq \|A_{\text{heavy}} \cdot v_1'\|_2^2 \geq \|A_{\text{heavy}} \cdot v_1\|_2^2 \geq (1-\beta)^2\|A\|_2^2$$

and therefore we have $\langle v_1', v_1\rangle^2 \geq 1 - 4\beta$, assuming $R \geq 2$. Thus, while processing the stream, we can store all the heavy rows and at the end of the stream compute the top right singular vector of A_{heavy}, in order to obtain a good approximation for v_1.

Suppose $\|A_{\text{heavy}} \cdot v_1\|_2 \leq (1-\beta)\|A\|_2$. This implies $\|A_{\text{light}} \cdot v_1\|_2^2 \geq \|A\|_2^2 - \|A_{\text{heavy}} \cdot v_1\|_2^2 \geq \beta \cdot \|A\|_2^2$. If we set $\beta \geq 2/R$, we have

$$\frac{\sigma_1(A_{\text{light}})^2}{\sigma_2(A_{\text{light}})^2} \geq \frac{\beta\|A\|_2^2}{\sigma_2(A)^2} \geq 2.$$

Let v_1' be the top singular vector of A_{light}. We will describe how to approximate v_1'. Consider applying the row norm sampling procedure with parameter ε to the matrix A_{light}. Given a row $a_i \in A_{\text{light}}$ the corresponding sampling probability p_i is given by

$$p_i = \frac{C\log d \cdot \|a_i\|_2^2}{\varepsilon^2\|A_{\text{light}}\|_2^2} \leq \frac{C\log d \cdot \|A\|_{\mathsf{F}}^2/(d \cdot \text{polylog}(d))}{\varepsilon^2\beta^2\|A\|_2^2} \leq \frac{C}{\varepsilon^2\beta^2\,\text{polylog}(d)}.$$

Assuming that $\varepsilon^2\beta^2 \geq 1/\text{polylog}(d)$, we obtain that $p_i < 1$ for all the rows in the matrix A_{light}. Let $\boldsymbol{B}_{\text{light}}$ be the matrix obtained after applying the row norm sampling procedure to the matrix A_{light}. Note that $\rho(\boldsymbol{B}_{\text{light}}) \approx \rho(\boldsymbol{A}_{\text{light}})$ and the number of rows in $\boldsymbol{B}_{\text{light}}$ is $\Theta(\rho(A_{\text{light}}) \cdot \log d \cdot \varepsilon^{-2})$, and therefore $\Theta(\rho(\boldsymbol{B}_{\text{light}}) \cdot \log d \cdot \varepsilon^{-2})$. Setting $\varepsilon = \alpha^2/\log^{5/2} d$, we obtain that the number of rows in the matrix $\boldsymbol{B}_{\text{light}}$ is $\Theta(\alpha^{-4} \cdot \rho(\boldsymbol{B}_{\text{light}}) \cdot \log^6 d)$ and thus assuming $\varepsilon^2\beta^2 = \alpha^4\beta^2/\log^5 d \geq 1/\text{polylog}(d)$, we can use Theorem 2.4 to obtain a vector $\hat{v}$ satisfying

$$\langle \hat{v}, v_1'\rangle^2 \geq 1 - 3\alpha.$$

We will now show that v_1' has a large correlation with v_1 which then implies $\hat{v}$ has a large correlation with v_1. Since $\|A_{\text{light}}\|_2 \geq \|A\|_2 - \|A_{\text{heavy}}\|_2 \geq \beta\|A\|_2$, $\|A_{\text{light}}\|_2^2 = \|A_{\text{light}} \cdot v_1'\|_2^2 \geq \beta\|A\|_2^2$. Consider the following upper bound on $\|A_{\text{light}} \cdot v_1'\|_2^2$:

$$\begin{aligned}
\|A_{\text{light}}\|_2^2 = \|A_{\text{light}} \cdot v_1'\|_2^2 &= \|A_{\text{light}} \cdot (\langle v_1', v_1\rangle \cdot v_1 + (I - v_1v_1^{\mathsf{T}})v_1')\|_2^2 \\
&= \|\langle v_1, v_1'\rangle A_{\text{light}} \cdot v_1 + A_{\text{light}}(I - v_1v_1^{\mathsf{T}})v_1'\|_2^2 \\
&\leq (1+\theta) \cdot \langle v_1, v_1'\rangle^2 \cdot \|A_{\text{light}} \cdot v_1\|_2^2 + (1+1/\theta) \cdot \|A_{\text{light}}(I - v_1v_1^{\mathsf{T}})v_1'\|_2^2
\end{aligned}$$

for any $\theta > 0$. Using the fact that the rows of the matrix A_{light} are a subset of the rows of the matrix A and that $\|A(I - v_1v_1^{\mathsf{T}})\|_2 = \sigma_2(A) = \sigma_1(A)/\sqrt{R}$, we have

$$\begin{aligned}
\|A_{\text{light}}\|_2^2 &\leq (1+\theta) \cdot \langle v_1, v_1'\rangle^2 \cdot \|A_{\text{light}}\|_2^2 + (1+1/\theta) \cdot \frac{\sigma_1^2}{R} \cdot (1 - \langle v_1, v_1'\rangle^2) \\
&= \langle v_1, v_1'\rangle^2((1+\theta) \cdot \|A_{\text{light}}\|_2^2 - (1+1/\theta)\sigma_1^2/R) + (1+1/\theta) \cdot \sigma_1^2/R
\end{aligned}$$

which implies

$$\begin{aligned}
\langle v_1, v_1'\rangle^2 \geq \frac{\|A_{\text{light}}\|_2^2 - (1+1/\theta) \cdot \sigma_1^2/R}{(1+\theta)\|A_{\text{light}}\|_2^2 - (1+1/\theta)\sigma_1^2/R} &= 1 - \frac{\theta \cdot \|A_{\text{light}}\|_2^2}{(1+\theta)\|A_{\text{light}}\|_2^2 - (1+1/\theta)\sigma_1^2/R} \\
&\geq 1 - \frac{\theta}{1+\theta-(1+1/\theta)/R\beta}
\end{aligned}$$

using the fact that $\|A_{\text{light}}\|_2^2 \geq \beta^2\sigma_1^2$. Now assuming $R\beta \geq 1$ and picking $\theta = 2/(R\beta - 1)$, we obtain

$$\langle v_1, v_1'\rangle^2 \geq 1 - \frac{4R\beta}{(1+R\beta)^2} \geq 1 - \frac{4}{R\beta}.$$

We therefore have

$$\langle \hat{v}, v_1 \rangle^2 \geq 1 - \frac{4}{R\beta} - 4\alpha. \tag{1}$$

Setting $\beta = 1/\sqrt{R}$ and $\alpha = 1/\sqrt{R}$, we satisfy all the requirements assuming that $R \leq \mathrm{polylog}(d)$ and obtain a vector $\hat{v}$ satisfying $\langle \hat{v}, v_1 \rangle^2 \geq 1 - 8/\sqrt{R}$. When $\|A_{\text{heavy}}\|_2 \geq (1-\beta)\|A\|_2$, we already have a vector $v' =$ top eigenvector of A_{heavy} that satisfies $\langle \hat{v}, v_1 \rangle^2 \geq 1 - 4\beta \geq 1 - 4/\sqrt{R}$. Thus, in both the cases, we obtain a vector $\hat{v}$ satisfying $\langle \hat{v}, v_1 \rangle^2 \geq 1 - O(1/\sqrt{R})$.

The procedure described requires knowing the approximate values of $\|A\|_{\mathsf{F}}$, $\|A_{\text{light}}\|_2$. Since, we assume that all the non-zero entries of the matrix have an absolute value at least $1/\operatorname{poly}(d)$ and at most $\operatorname{poly}(d)$, the values $\|A\|_{\mathsf{F}}, \|A_{\text{light}}\|_2$ lie in the interval $[1/\operatorname{poly}(d), \operatorname{poly}(nd)]$. Hence, using $O(\log nd)$ guesses each for $\|A\|_{\mathsf{F}}$ and $\|A_{\text{light}}\|_2$ and using a Gaussian sketch of A similar to that in Algorithm 1, we can obtain a vector satisfying the guarantees in the theorem. □

A.4 Proof of Theorem 3.1

Proof. For each $i \in [h]$, let $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_h$ be drawn independently and uniformly at random from $\{+1, -1\}^d$. Let $\boldsymbol{i} \sim [h]$ be drawn uniformly at random, and for an integer k to be chosen later, let $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_k \in \mathbb{R}^d$ be vectors that share the first $(1-\gamma)d$ coordinates with the vector $\boldsymbol{x}_{\boldsymbol{i}}$. Each of the last $\gamma \cdot d$ elements of each of $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_k$ are sampled uniformly at random from the set $\{+1, -1\}$. Define $\boldsymbol{z}_1, \ldots, \boldsymbol{z}_{h+k}$ such that for $j \leq h$, $\boldsymbol{z}_j = \boldsymbol{x}_j$ and for $j > h$, let $\boldsymbol{z}_j = \boldsymbol{y}_{j-h}$.

Now consider the stream $\boldsymbol{z}_1, \ldots, \boldsymbol{z}_{h+k}$. Price and Xun argue that when $k \geq 4R$, the gap of this stream is at least R with large probability over the randomness used in the construction of the stream. Let $\boldsymbol{\pi} : [h+k] \to [h+k]$ be a uniformly random permutation independent of $\boldsymbol{i}$. Consider the following event $\mathcal{E}$:

$$\boldsymbol{\pi}(\boldsymbol{i}) \leq h/2 \text{ and } \boldsymbol{\pi}(h+1), \ldots, \boldsymbol{\pi}(h+k) > h/2.$$

We have that the probability of the event $\mathcal{E}$ is

$$\frac{h/2+k}{h+k} \cdot \frac{h/2+k-1}{h+k-1} \cdots \frac{h/2+1}{h+1} \cdot \frac{h/2}{h} \geq (1/2)^{k+1}.$$

Let $S_{\boldsymbol{i}}$ be the set of permutations π that satisfy the above event. Therefore we have $\mathbf{Pr}_{\boldsymbol{\pi}}[\boldsymbol{\pi} \in S_{\boldsymbol{i}}] \geq (1/2)^{k+1}$. If the probability of failure, δ, of the algorithm $\mathcal{A}$ satisfies $\delta \leq (1/2)^{k+4}$, we have that

$$\mathop{\mathbf{Pr}}_{\boldsymbol{\pi},\ \text{internal randomness}}[\mathcal{A} \text{ succeeds on } \boldsymbol{z}_{\boldsymbol{\pi}(1)}, \ldots, \boldsymbol{z}_{\boldsymbol{\pi}(h+k)} \mid \boldsymbol{\pi} \in S_{\boldsymbol{i}}] \geq \frac{3}{4}.$$

Let $\boldsymbol{s}_{\text{mid}}$ be the state of the algorithm after $h/2$ steps and $\boldsymbol{s}_{\text{fin}}$ be the final state of the algorithm. The randomness in $\boldsymbol{s}_{\text{fin}}$ is from the following sources: (i) randomness of the vectors $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_h$, (ii) the index $\boldsymbol{i} \in [h]$, (iii) the vectors $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_k$, (iv) the permutation $\boldsymbol{\pi}$, and (v) the internal randomness of the algorithm. From here on, condition on the event $\mathcal{E}$, i.e., that the permutation $\boldsymbol{\pi} \in S_{\boldsymbol{i}}$. We will not explicitly mention that all entropy and information terms in the proof are conditioned on $\mathcal{E}$. Since $\boldsymbol{\pi}(\boldsymbol{i}) \leq h/2$, we have

$$\boldsymbol{s}_{\text{fin}} \text{ is conditionally independent of } x_{\boldsymbol{i}}[(1-\gamma)\cdot d + 1 : d] \text{ given } \boldsymbol{s}_{\text{mid}}.$$

Using the data processing inequality, we obtain that

$$I(\boldsymbol{s}_{\text{mid}}; \boldsymbol{x}_{\boldsymbol{i}}[(1-\gamma)\cdot d + 1 : d]) \geq I(\boldsymbol{s}_{\text{fin}}; \boldsymbol{x}_{\boldsymbol{i}}[(1-\gamma)\cdot d + 1 : d]).$$

When $h \leq cd/R^2$, $k = 4R$, $\gamma = 1/4$ and $\varepsilon \leq c/k^2$ for a small constant, we have as in the proof of Theorem 1.5 in Price and Xun (2024) that,

$$I(\boldsymbol{s}_{\text{fin}}; \boldsymbol{x}_{\boldsymbol{i}}[(1-\gamma)\cdot d + 1 : d]) \geq \Omega(d/R)$$

which now implies

$$I(\boldsymbol{s}_{\text{mid}}; \boldsymbol{x}_{\boldsymbol{i}}[(1-\gamma)\cdot d + 1 : d]) \geq \Omega(d/R).$$

Note that conditioned on the event $\mathcal{E}$, the distribution of $\boldsymbol{i}$ is uniform over $\{\boldsymbol{\pi}^{-1}(1), \ldots, \boldsymbol{\pi}^{-1}(h/2)\}$. We now prove the following lemma:

Lemma A.3. *Let $\boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_\ell$ be independent random variables. Let $\boldsymbol{i} \sim [\ell]$ be a uniform random variable independent of $\boldsymbol{X}$. We have*

$$I(\boldsymbol{X}\,;\,\boldsymbol{Y}_1) + \cdots + I(\boldsymbol{X}\,;\,\boldsymbol{Y}_\ell) \geq \ell \cdot (I(\boldsymbol{X}; \boldsymbol{Y}_{\boldsymbol{i}}) - \log_2 \ell).$$

Proof. By definition, we have

$$I(\boldsymbol{X}\,;\boldsymbol{Y}_{\boldsymbol{i}}) = H(\boldsymbol{Y}_{\boldsymbol{i}}) - H(\boldsymbol{Y}_{\boldsymbol{i}} \mid \boldsymbol{X}).$$

Now, we note that $H(\boldsymbol{Y}_{\boldsymbol{i}}) \leq H(\boldsymbol{Y}_{\boldsymbol{i}}, \boldsymbol{i}) = H(\boldsymbol{i}) + H(\boldsymbol{Y}_{\boldsymbol{i}} \mid \boldsymbol{i}) = \log_2 \ell + \frac{H(\boldsymbol{Y}_1)+\cdots+H(\boldsymbol{Y}_\ell)}{\ell}$. We now lower bound $H(\boldsymbol{Y}_{\boldsymbol{i}} \mid \boldsymbol{X})$. Since conditioning always decreases entropy, we obtain

$$H(\boldsymbol{Y}_{\boldsymbol{i}} \mid \boldsymbol{X}) \geq H(\boldsymbol{Y}_{\boldsymbol{i}} \mid \boldsymbol{i}, \boldsymbol{X}).$$

As $\boldsymbol{X}$ is independent of $\boldsymbol{i}$, we have

$$H(\boldsymbol{Y}_{\boldsymbol{i}} \mid \boldsymbol{X}) \geq H(\boldsymbol{Y}_{\boldsymbol{i}} \mid \boldsymbol{i}, \boldsymbol{X}) = \frac{H(\boldsymbol{Y}_1 \mid \boldsymbol{X}) + \cdots + H(\boldsymbol{Y}_\ell \mid \boldsymbol{X})}{\ell}$$

which then implies

$$\begin{aligned} I(\boldsymbol{X}\,;\,\boldsymbol{Y}_{\boldsymbol{i}}) &\leq H(\boldsymbol{i}) + \frac{H(\boldsymbol{Y}_1) + \cdots + H(\boldsymbol{Y}_\ell)}{\ell} - \frac{H(\boldsymbol{Y}_1 \mid \boldsymbol{X}) + \cdots + H(\boldsymbol{Y}_\ell \mid \boldsymbol{X})}{\ell} \\ &\leq H(\boldsymbol{i}) + \frac{I(\boldsymbol{X}\,;\,\boldsymbol{Y}_1) + \cdots + I(\boldsymbol{X}\,;\,\boldsymbol{Y}_\ell)}{\ell}. \end{aligned}$$

Since $H(\boldsymbol{i}) = \log_2 \ell$, we have the proof. □

Using this lemma,

$$\begin{aligned} &I(\boldsymbol{s}_{\text{mid}}; \boldsymbol{x}_{\boldsymbol{\pi}^{-1}(1)}[(1-\gamma)\cdot d + 1 : d]) + \cdots + I(\boldsymbol{s}_{\text{mid}}; \boldsymbol{x}_{\boldsymbol{\pi}^{-1}(h/2)}[(1-\gamma)\cdot d + 1 : d]) \\ &= (h/2) \cdot I(\boldsymbol{s}_{\text{mid}}; \boldsymbol{x}_{\boldsymbol{i}}[(1-\gamma)\cdot d + 1 : d] - \log_2(h/2)) \\ &\geq \Omega(hd/R) - h \log_2 h. \end{aligned}$$

Lemma A.4. *If $\boldsymbol{X}, \boldsymbol{Y}$ are independent, then $I(\boldsymbol{Z}\,;\,(\boldsymbol{X}, \boldsymbol{Y})) \geq I(\boldsymbol{Z}\,;\,\boldsymbol{X}) + I(\boldsymbol{Z}\,;\,\boldsymbol{Y})$.*

Proof.

$$\begin{aligned} I(\boldsymbol{Z}\,;\,(\boldsymbol{X}, \boldsymbol{Y})) &= H((\boldsymbol{X}, \boldsymbol{Y})) - H((\boldsymbol{X}, \boldsymbol{Y}) \mid \boldsymbol{Z}) \\ &= H(\boldsymbol{X}) + H(\boldsymbol{Y}) - H((\boldsymbol{X}, \boldsymbol{Y}) \mid \boldsymbol{Z}). \end{aligned}$$

Now, we note that for any three random variables $\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}$, we have $H((\boldsymbol{X}, \boldsymbol{Y}) \mid \boldsymbol{Z}) \leq H(\boldsymbol{X} \mid \boldsymbol{Z}) + H(\boldsymbol{Y} \mid \boldsymbol{Z})$ which proves the lemma. □

Using the independence of $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_h$ conditioned on the event $\mathcal{E}$, we obtain

$$I(\boldsymbol{s}_{\text{mid}}; (\boldsymbol{x}_{\boldsymbol{\pi}^{-1}(1)}[(1-\gamma)\cdot d + 1 : d], \ldots, \boldsymbol{x}_{\boldsymbol{\pi}^{-1}(h/2)}[(1-\gamma)\cdot d + 1 : d])) \geq \Omega(hd/R) - h \log_2 h$$

which then implies

$$H(\boldsymbol{s}_{\text{mid}}) \geq \Omega(hd/R)$$

using the fact that $R^2 \cdot h = O(d)$. Finally, we have $\max |\boldsymbol{s}_{\text{mid}}| \geq \Omega(hd/R)$. Here $|\boldsymbol{s}_{\text{mid}}|$ is the number of bits used in the representation of the state $\boldsymbol{s}_{\text{mid}}$. □

A.5 Proof of Theorem 5.1

Proof. Our instance consists of the following vectors:

1. R copies of the vector $(1/\sqrt{R})e_1$,
2. 1 copy of the vector $(1/\sqrt{R-\varepsilon})e_2$, and
3. α copies of the vector $(1/\sqrt{\alpha \cdot R})e_3$.

where $\alpha = 2M$. Let A be a matrix with rows given by the stream of vectors defined above. We note that the matrix A has rank 3 and the non-zero eigenvalues of the matrix $A^\mathsf{T}A$ are $1, 1/(R-\varepsilon), 1/R$ and therefore the *gap* $\lambda_1(A^\mathsf{T}A)/\lambda_2(A^\mathsf{T}A) = R - \varepsilon$. The top eigenvector of the matrix $A^\mathsf{T}A$ is e_1 and the row with the largest norm is $(1/\sqrt{R-\varepsilon})e_2$. Thus, the row with the largest norm is not useful to obtain correlation with the true top eigenvector e_1.

Consider an execution of Oja's algorithm with a learning rate η on the above stream of vectors. The final vector z_n can be written as

$$z_n = \left(I + \frac{\eta}{R} e_1 e_1^\mathsf{T}\right)^R \left(I + \frac{\eta}{R\alpha} e_3 e_3^\mathsf{T}\right)^\alpha \left(I + \frac{1}{R-\varepsilon} e_2 e_2^\mathsf{T}\right) v_0.$$

For $j \in [d]$, let z_{ij} denote the j-th coordinate of the vector z_i so that we have

$$\begin{aligned} z_{n1} &= \left(1 + \frac{\eta}{R}\right)^R \cdot z_{01}, \\ z_{n2} &= \left(1 + \frac{\eta}{R-\varepsilon}\right) \cdot z_{02}, \quad \text{and} \\ z_{n3} &= \left(1 + \frac{\eta}{R\alpha}\right)^\alpha \cdot z_{03}. \end{aligned}$$

We note that $z_{nj} = z_{0j}$ for all $j > 3$. Since $\alpha = 2M$, we have $\eta/R\alpha \leq 1/2$ and therefore $(1 + \eta/R\alpha) \geq \exp(\eta/2R\alpha)$ and $(1 + \eta/R\alpha)^\alpha \geq \exp(\eta/2R)$.

Recall that we want to show that $|\langle z_n, e_1\rangle| < c\|z_n\|_2$ with a large probability. Suppose otherwise and that with probability $\geq 1/10$, we have $|\langle z_n, e_1\rangle| > c\|z_n\|_2 > c\|(0,\ 0,\ 0,\ z_{04},\ \ldots,\ z_{0d})\|_2$.

Since, z_0 is initialized to be a random Gaussian, we have $\|(0, 0, 0, z_{04}, \ldots, z_{0d})\|_2 \geq \sqrt{d}/2$ with probability $1 - \exp(-d)$. Thus, we have with probability $\geq 1/11$ that,

$$|z_{n1}| \geq c\sqrt{d}/2$$

which implies the learning rate must satisfy

$$(1 + \eta/R)^R \geq c'\sqrt{d}/2$$

since $|z_{01}| \leq 10$ with probability $\geq 99/100$. Hence $\eta \geq R((c'd^{1/2})^{1/R} - 1)$. Now consider $|\langle z_n, e_3\rangle|/|\langle z_n, e_1\rangle|$. We have

$$\frac{|\langle z_n, e_3\rangle|}{|\langle z_n, e_1\rangle|} = \frac{\exp(\eta/R)}{(1 + \eta/R)^R} \cdot \frac{|z_{03}|}{|z_{01}|}.$$

With probability $\geq 95/100$, we have $1/C \leq |z_{03}|/|z_{01}| \leq C$ for a large enough constant C. We now consider the expression

$$\frac{\exp(\eta/R)}{(1 + \eta/R)^R}.$$

The expression is minimized at $\eta = R^2 - R$ and is increasing in the range $\eta \in [R^2 - R, \infty)$. When, $R = O(\log d/\log\log d)$, we have that $R^2 - R \leq R((c'd^{1/2})^{1/R} - 1)$ and therefore for all $\eta \geq R((c'd^{1/2})^{1/R} - 1)$, we have

$$\frac{\exp(\eta/R)}{(1 + \eta/R)^R} \geq \frac{\exp((c'd^{1/2})^{1/R})}{e \cdot c'd^{1/2}}.$$

When $R = O(\log d/\log\log d)$, we have

$$\frac{\exp(\eta/R)}{(1 + \eta/R)^R} \geq \text{poly}(d)$$

which then implies $|\langle z_n, e_3\rangle| \geq |\langle z_n, e_1\rangle| \cdot \text{poly}(d)/C$ with probability $\geq 95/100$ which contradicts our assumption that $|\langle z_n, e_1\rangle| \geq c\|z_n\|_2$. □

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Our paper is purely theoretical studying space-efficient algorithms for approximating the top eigenvector. We prove all the claims made in the abstract and introduction.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We do not have a specific limitations section but we do qualify all the statements noting the assumptions that need to be made to prove that our algorithms work.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: We include all the proofs in the main body and the appendix.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [NA]

   Justification: No experimental results are given in this paper.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [NA]

Justification: [NA]

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [NA]

Justification: [NA]

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [NA]

Justification: [NA]

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)

- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [NA]

   Justification: [NA]

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: Results in this paper are purely theoretical. While the algorithms proposed in this paper may be used with potentially negative consequences, the authors are unaware of such uses.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [No]

    Justification: Our work studies algorithms for the top eigenvector estimation problem. Our work is purely theoretical. While our algorithms may be used to impact society in a negative way, we are unaware of such usecases.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: [NA]

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [NA]

    Justification: [NA]

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.

- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: [NA]

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: [NA]

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: [NA]

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.