JMLR: Workshop and Conference Proceedings vol 49:1–18, 2016

Streaming PCA: Matching Matrix Bernstein and Near-Optimal Finite Sample Guarantees for Oja's Algorithm

Prateek Jain PRAJAIN@MICROSOFT.COM
Microsoft Research, Bangalore India

Chi Jin CHIJIN@CS.BERKELEY.EDU
UC Berkeley, Berkeley CA

Sham M. Kakade SHAM@CS.WASHINGTON.EDU
University of Washington, Seattle WA

Praneeth Netrapalli PRANEETH@MICROSOFT.COM
Microsoft Research, Cambridge MA

Aaron Sidford ASID@MICROSOFT.COM
Microsoft Research, Cambridge MA

Abstract

In this paper we provide improved guarantees for streaming principal component analysis (PCA). Given $\mathbf{A}_1, \ldots, \mathbf{A}_n \in \mathbb{R}^{d\times d}$ sampled independently from distributions satisfying $\mathbb{E}[\mathbf{A}_i] = \mathbf{\Sigma}$ for $\mathbf{\Sigma} \succeq \mathbf{0}$, we present an $O(d)$-space linear-time single-pass streaming algorithm for estimating the top eigenvector of $\mathbf{\Sigma}$. The algorithm nearly matches (and in certain cases improves upon) the accuracy obtained by the standard batch method that computes top eigenvector of the empirical covariance $\frac{1}{n}\sum_{i\in[n]}\mathbf{A}_i$ as analyzed by the matrix Bernstein inequality. Moreover, to achieve constant accuracy, our algorithm improves upon the best previous known sample complexities of streaming algorithms by either a multiplicative factor of $O(d)$ or $1/\mathrm{gap}$ where gap is the relative distance between the top two eigenvalues of $\mathbf{\Sigma}$.

We achieve these results through a novel analysis of the classic Oja's algorithm, one of the oldest and perhaps, most popular algorithms for streaming PCA. We show that simply picking a random initial point $\mathbf{w}_0$ and applying the natural update rule $\mathbf{w}_{i+1} = \mathbf{w}_i + \eta_i\mathbf{A}_i\mathbf{w}_i$ suffices for suitable choice of η_i. We believe our result sheds light on how to efficiently perform streaming PCA both in theory and in practice and we hope that our analysis may serve as the basis for analyzing many variants and extensions of streaming PCA.

1. Introduction

Principal component analysis (PCA) is one of the most fundamental problems in machine learning, numerical linear algebra, and data analysis. It is commonly used for data compression, image processing, and visualization (Jolliffe, 2002) etc.

However, when run on large data sets it may be the case that we cannot afford more than single pass over the data (or worse to even store the data in the first place) (Hall et al., 1998; Weng et al., 2003; Ross et al., 2008). To alleviate this issue, a popular line of research over the past several decades has been to consider streaming algorithms for PCA under the assumption that the data has reasonable statistical properties (Krasulina, 1970; Oja, 1982; Balsubramani et al., 2013; Mitliagkas

et al., 2013; Sa et al., 2015). There have been significant breakthroughs in getting near-optimal streaming PCA algorithms under fairly specialized models, e.g. spiked covariance (Sa et al., 2015).

In this paper we consider one of the most fundamental and natural variants of PCA, estimating the top eigenvector of a symmetric matrix, under one of the mildest set of assumptions for which we can prove concentration using the matrix Bernstein inequality (Vershynin, 2010; Tropp, 2012):

Definition 1 (Streaming PCA) *Let $\mathbf{A}_1, \mathbf{A}_2, ..., \mathbf{A}_n \in \mathbb{R}^{d\times d}$ be a sequence of (not necessarily symmetric) matrices sampled independently from distributions that satisfy the following:*

1. $\mathbb{E}[\mathbf{A}_i] = \mathbf{\Sigma}$ *for symmetric positive semidefinite (PSD) matrix* $\mathbf{\Sigma} \in \mathbb{R}^{d\times d}$,
2. $\|\mathbf{A}_i - \mathbf{\Sigma}\|_2 \le \mathcal{M}$ *with probability* 1, *and*
3. $\max\left\{\left\|\mathbb{E}\left[(\mathbf{A}_i - \mathbf{\Sigma})(\mathbf{A}_i - \mathbf{\Sigma})^\top\right]\right\|_2, \left\|\mathbb{E}\left[(\mathbf{A}_i - \mathbf{\Sigma})^\top(\mathbf{A}_i - \mathbf{\Sigma})\right]\right\|_2\right\} \le \mathcal{V}$.

Let $\mathbf{v}_1, ..., \mathbf{v}_d$ denote the eigenvectors of $\mathbf{\Sigma}$ and $\lambda_1 \ge ... \ge \lambda_d$ denote the corresponding eigenvalues. Our goal is to compute an ϵ-approximation to $\mathbf{v}_1$, that is a unit vector $\mathbf{w}$ such that $\sin^2(\mathbf{w}, \mathbf{v}_1) = 1 - (\mathbf{w}^\top \mathbf{v}_1)^2 \le \epsilon$, in a single pass while minimizing space, time, and error (i.e. ϵ).

A special case of Streaming PCA is to estimate the top eigenvector of the covariance matrix of a distribution $\mathcal{D}$ over $\mathbb{R}^d$, i.e. given independent samples $\mathbf{a}_1, ..., \mathbf{a}_n \in \mathbb{R}^d$ estimate the top eigenvector of $\mathbb{E}_{\mathbf{a}\sim\mathcal{D}}[\mathbf{a}\mathbf{a}^\top]$. This encompasses the popular "spiked covariance model" (Johnstone, 2001).

It is well known that to solve the Streaming PCA problem we could simply compute the empirical covariance matrix $\frac{1}{n}\sum_{i\in[n]} \mathbf{A}_i$ and compute the right singular vector of this matrix. Using matrix Bernstein inequality (Vershynin, 2010; Tropp, 2012) and Wedin's theorem (Wedin, 1972) we get the following standard sample complexity bound for the Streaming PCA problem:

Theorem 2 (Eigenvector Concentration using matrix Bernstein and Wedin's theorem) *Under the assumptions of Definition 1, the top right singular vector $\widehat{\mathbf{v}}$ of $\widehat{\mathbf{\Sigma}} = \frac{1}{n}\sum_{i\in[n]} \mathbf{A}_i$ is an ϵ-approximation to the top eigenvector $\mathbf{v}_1$ of $\mathbf{\Sigma}$ with probability $1-\delta$, where*

$$\sin^2(\widehat{\mathbf{v}}, \mathbf{v}_1) \le \epsilon \le \frac{16\mathcal{V}\log\frac{d}{\delta}}{(\lambda_1 - \lambda_2)^2}\cdot\frac{1}{n} + \left(\frac{4\mathcal{M}\log\frac{d}{\delta}}{\lambda_1 - \lambda_2}\right)^2\cdot\frac{1}{n^2}.$$

Theorem 2 is essentially the previous best sample complexity we know for solving the Streaming PCA problem[1]. Unfortunately, there are severe issues with applying the result algorithmically. First, computing the empirical covariance matrix $\widehat{\mathbf{\Sigma}} = \frac{1}{n}\sum_{i\in[n]} \mathbf{A}_i$ naively requires $O(d^2)$ time and space, and second, computing the top eigenvector of the empirical covariance matrix in general may require super linear time (Golub and Van Loan, 2012). While there have been many attempts to produce streaming algorithms that use only $O(d)$ space to solve the streaming PCA problem, as far as we are aware all previous methods either lose a multiplicative factor of either $\frac{\lambda_1}{\lambda_1-\lambda_2}$ or d in the analysis in order to achieve constant accuracy when applied in our setting (Balsubramani et al., 2013; Mitliagkas et al., 2013; Hardt and Price, 2014; Sa et al., 2015; Jin et al., 2015).

In an attempt to overcome this limitation and improve the guarantees for solving the streaming PCA problem we address the fundamental question:

1. In recent work in (Jin et al., 2015) it was shown that the $log(d/\delta)$ factor in the first term could be removed asymptotically for small enough ϵ if only constant success probability is required.

Can we match the sample complexity of matrix Bernstein + Wedin's theorem with an algorithm that uses $O(d)$ space only and takes a single linear-time pass over the input?

We answer the question in the affirmative, showing that we can succeed with constant probability matching the sample complexity of Theorem 2 up to logarithmic terms and small additive factors. Interestingly, we achieve this result by providing a novel analysis of the classical Oja's algorithm, which is perhaps, the most popular algorithm for Streaming PCA (Oja, 1982).

Algorithm 1 Oja's algorithm for computing top eigenvector

Input: $\mathbf{A}_1, \cdots, \mathbf{A}_n$.
  Choose $\mathbf{w_0}$ uniformly at random from the unit sphere
  for $t = 1, \cdots, n$ **do**
    $\mathbf{w_i} \leftarrow \mathbf{w_{i-1}} + \eta_i \mathbf{A}_i \mathbf{w_{i-1}}$
    $\mathbf{w_i} \leftarrow \mathbf{w_i} / \|\mathbf{w_i}\|_2$
  end for
Output: $\mathbf{w_n}$

Oja's algorithm is one of the simplest algorithms one would imagine for the streaming PCA problem (See Algorithm 1). In the case that each $\mathbf{A}_i$ come from the same distribution $\mathcal{D}$ it corresponds to simply performing projected stochastic gradient descent on the objective function of maximizing the Rayleigh Quotient over the distribution $\max_{\|w\|_2=1} \mathbb{E}_{\mathbf{A}\sim\mathcal{D}} \mathbf{w}^\top \mathbf{A}\mathbf{w}$. It is well known that under very mild conditions on the stepsize sequence, Oja's algorithm asymptotically converges to the top eigenvector of the covariance matrix $\mathbf{\Sigma}$ (Oja, 1982). However, obtaining optimal rates of convergence, let alone finite sample guarantees, for Streaming PCA has been quite challenging. The best known results are off from Theorem 2 by a factor of $\mathcal{O}(d)$ (Sa et al., 2015).

In this paper we show that for proper choice of learning rates η_i Oja's algorithm in fact can improve the best known results for streaming PCA and answer our question in the affirmative. In particular we show the following:

Theorem 3 *Let the assumptions of Definition 1 hold. Suppose the step size sequence for Algorithm 1 is chosen to be $\eta_i = \frac{\log d}{(\lambda_1-\lambda_2)(\beta+i)}$, where*

$$\beta \triangleq 40 \max\left(\frac{\mathcal{M}\log d}{(\lambda_1-\lambda_2)}, \frac{\mathcal{V}\log^2 d}{(\lambda_1-\lambda_2)^2}, \frac{(\lambda_1)^2\log^2 d}{(\lambda_1-\lambda_2)^2}\right).$$

Then the output $\mathbf{w_n}$ of Algorithm 1 is an ϵ-approximation to the top eigenvector $\mathbf{v}_1$ of $\mathbf{\Sigma}$ satisfying

$$\sin^2(\mathbf{w}_n, \mathbf{v}_1) \le \epsilon \ \le \ C\left(\frac{\mathcal{V}\log d}{(\lambda_1-\lambda_2)^2}\cdot\frac{1}{n} + \left(\frac{2\beta}{n}\right)^{2\log d}\right),$$

with probability greater than $3/4$. Here C is an absolute numerical constant.

The error above should be interpreted as being the sum of a higher order $\Theta(\frac{1}{n})$ term and another $\mathcal{O}\left((2\beta/n)^{2\log d}\right)$ term that decays atleast as $o\left(\frac{1}{n^{\log d}}\right)$ (as soon as $n > 2\beta^2$). In particular, this result says that up to an additive lower order term, we can match Theorem 2 with an asymptotic

error of $\mathcal{O}\left(\frac{\mathcal{V}\log d}{(\lambda_1-\lambda_2)^2 n}\right)$ with constant probability. The lower order term has β which is the $\max$ of three parts: $\frac{\mathcal{M}\log d}{(\lambda_1-\lambda_2)}$, $\frac{\mathcal{V}\log^2 d}{(\lambda_1-\lambda_2)^2}$ and $\frac{\lambda_1^2\log^2 d}{(\lambda_1-\lambda_2)^2}$. The first part, depending on $\mathcal{M}$, is exactly the same as what appears in Theorem 2. The second one, depending on $\mathcal{V}$ has an additional $\log d$ factor over the first order term and is irrelevant once, say $n > 10\beta$. The third part, depending on λ_1^2, does not appear in Theorem 2, but arises here completely due to computational reasons: we are allowed only a *single linear-time pass* over the matrices, while Theorem 2 makes no such assumption. For instance, consider the case $\mathcal{V} = 0$ which means $\mathbf{A}_1 = \mathbf{\Sigma}$. Matrix Bernstein tells us that one sample is sufficient to compute $\mathbf{v}_1$. However, we still do not know how to compute it using a single pass over $\mathbf{A}_1$. Note however, that the rate at which we decrease the lower order terms i.e., $o\left(\frac{1}{n^{\log d}}\right)$, is much better than $\mathcal{O}\left(1/n^2\right)$ guaranteed by Theorem 2.

In fact we also improve the asymptotic error rate obtained by Theorem 2. In particular, we prove the following result that Oja's algorithm gets an asymptotic rate of $\mathcal{O}\left(\frac{\mathcal{V}}{(\lambda_1-\lambda_2)^2 n}\right)$ which is better than that of matrix Bernstein by a factor of $\mathcal{O}\left(\log d\right)$.[2]

Theorem 4 *Let the assumptions of Definition 1 hold. Suppose the step size sequence for Algorithm 1 is chosen to be* $\eta_i = \frac{6}{(\lambda_1-\lambda_2)(\beta+i)}$, *where*

$$\beta \triangleq 720\max\left(\frac{\mathcal{M}}{(\lambda_1-\lambda_2)}, \frac{\mathcal{V}+\lambda_1^2}{(\lambda_1-\lambda_2)^2}\right).$$

Suppose $n > \beta^{1.2}d^{0.1}$. *Then the output* $\mathbf{w_n}$ *of Algorithm 1 is an* ϵ*-approximation to the top eigenvector* $\mathbf{v}_1$ *of* $\mathbf{\Sigma}$ *satisfying*

$$\sin^2(\mathbf{w}_n, \mathbf{v}_1) \le \epsilon \ \le\ C\left(\frac{\mathcal{V}}{(\lambda_1-\lambda_2)^2}\cdot\frac{1}{n} + \frac{1}{n^2}\right),$$

with probability greater than $3/4$. *Here* C *is an absolute numerical constant.*

Note that Theorems 3 and 4 guarantee success probability of $3/4$. One way to boost the probability to $1-\delta$, for some $\delta > 0$, is to run $\mathcal{O}\left(\log 1/\delta\right)$ copies of the algorithm, each with $3/4$ success probability and then output the geometric median of the solutions, which can be done in nearly linear time (Cohen et al., 2010). We omit the details here.

Beyond the improved sample complexities we believe our analysis sheds light on the type of step sizes for which Oja's algorithm converges quickly and therefore illuminates how to efficiently perform streaming PCA. Moreover, we believe that our analysis is fairly general and we hope that it may be extended to make progress on analyzing the many variants of PCA that occur in both theory and in practice.

1.1. Comparison with Existing Results

Here we compare our sample complexity bounds with existing analyses of various methods. Recall that we measure the error of estimate $\mathbf{w}$ by $\sin^2(\mathbf{w}, \mathbf{v}_1) = 1 - (\mathbf{w}^\top\mathbf{v}_1)^2$.

We consider three popular methods used for computing $\mathbf{v}_1$. The first one is the batch method which computes largest eigenvector of empirical covariance and uses Wedin's theorem with matrix

2. A similar asymptotic result was recently obtained by (Jin et al., 2015). However, their result requires an initial vector that is constant close to $\mathbf{v}_1$, which itself is a difficult problem.

Algorithm	Error	$\mathcal{O}(d)$ space?
Oja's **(our result, Theorem 8)**	$\mathcal{O}\left(\frac{\mathcal{V}}{(\lambda_1-\lambda_2)^2}\cdot\frac{1}{n}\right)$	Yes
Matrix Bernstein + Wedin's theorem (Theorem 2)	$\mathcal{O}\left(\frac{\mathcal{V}\log d}{(\lambda_1-\lambda_2)^2}\cdot\frac{1}{n}\right)$	No
Alecton (Sa et al., 2015)	$\mathcal{O}\left(\frac{\mathcal{V}d}{(\lambda_1-\lambda_2)^2}\cdot\frac{\log n}{n}\right)$	Yes
Block Power Method (Hardt and Price, 2014)	$\mathcal{O}\left(\frac{\mathcal{V}\lambda_1\log d}{(\lambda_1-\lambda_2)^3}\cdot\frac{\log n}{n}\right)$	Yes

Table 1: Asymptotic error guaranteed by various methods under assumptions of Definition 1 with at least constant probability, and ignoring constant factors. Recall that we define the error to be $\sin^2(\mathbf{w},\mathbf{v}_1) = 1-(\mathbf{w}^\top\mathbf{v}_1)^2$. Our analysis provides the optimal $1/n$ error decay rate as compared to Alecton and Block power method which obtain $\frac{\log n}{n}$. Moreover, our bound is $O(d)$ tighter than that of Alecton (Sa et al., 2015) and $O(\frac{\lambda_1}{\lambda_1-\lambda_2})$ tighter bound than that of Block Power Method (Hardt and Price, 2014). The assumptions made in (Sa et al., 2015) for Alecton are different from ours (which are much more standard) so we optimized their bounds for our setting. See Section 1.1 for a concrete example where our analysis provides these improvements over (Sa et al., 2015; Hardt and Price, 2014).

Bernstein inequality (cf. Theorem 2). The second method is Alecton, which is very similar to Oja's algorithm (Sa et al., 2015). Finally, we also consider a block-power method (BPM) (Hardt and Price, 2014; Mitliagkas et al., 2013) which divides samples into different blocks and applies power iteration to the empirical estimate from each block. See Table 1 for the comparison.

We would like to stress that some of the results we compare to make different assumptions than Definition 1. The bounds we give for them are our best attempt to adapt their bounds in the setting of Definition 1 (which is quite standard). In the next paragraph, we give a simple example, which demonstrates the improvement in our result as compared to existing work.

Let $\mathbf{A}_i = \mathbf{x}_i\mathbf{x}_i^\top$, where $\mathbf{x}_i \in \mathbb{R}^d$ and $\mathbf{x}_i = \mathbf{e}_1$ with probability $1/d$ and $\mathbf{x}_i = \sigma\mathbf{e}_j, 1 < j \le d$ with probability $1/d$ where e_j denotes the j^{th} standard basis vector and $\sigma < 1$. Note that $\mathbf{\Sigma} = \mathbb{E}[\mathbf{A}_i] = \frac{(1-\sigma^2)}{d}\mathbf{e}_1\mathbf{e}_1^T + \frac{1}{d}\sigma^2\mathbf{I}$, $\|\mathbf{A}_i\|_2 \le 1$ for all i, and $\|\mathbb{E}\left[\mathbf{A}_i\mathbf{A}_i^\top\right]\|_2 \le \frac{1}{d}$. Even for constant accuracy $\epsilon = \Omega(1)$, Theorem 3 tells us that $n = \mathcal{O}\left(\frac{d\log^2 d}{(1-\sigma^2)^2}\right)$ is sufficient. On the other hand, Theorem 1 of (Sa et al., 2015) requires $n = \mathcal{O}\left(\frac{d^2\log^2 d}{(1-\sigma^2)^2}\right)$, while Theorem 2.4 of (Hardt and Price, 2014) requires $n = \mathcal{O}\left(\frac{d\log^2 d}{(1-\sigma^2)^3}\right)$. Asymptotically, as n becomes larger, our error scales as $\mathcal{O}\left(\frac{d}{(1-\sigma^2)^2}\cdot\frac{1}{n}\right)$ while that of (Sa et al., 2015) scales as $\mathcal{O}\left(\frac{d^2}{(1-\sigma^2)^2}\cdot\frac{\log n}{n}\right)$ and that of (Hardt and Price, 2014) scales as $\mathcal{O}\left(\frac{d}{(1-\sigma^2)^3}\cdot\frac{\log n}{n}\right)$. Combining matrix Bernstein and Wedin's theorems gives an asymptotic error of $\mathcal{O}\left(\frac{d\log d}{(1-\sigma^2)^2}\cdot\frac{1}{n}\right)$.

1.2. Additional Related Work

Existing results for computing largest eigenvector of a data covariance matrix using streaming samples can be divided into three broad settings: a) stochastic data, b) arbitrary sequence of data, c) regret bounds for arbitrary sequence of data.

Stochastic data: Here, the data is assumed to be sampled i.i.d. from a fixed distribution. Our analysis of Oja's algorithm as well as those of block power method and Alecton mentioned earlier are in this setting. (Mitliagkas et al., 2013) also obtained a result in the restricted spiked covariance model. (Balsubramani et al., 2013) provides an analysis of a modification of Oja's algorithm but with an extra $O(d^5)$ multiplicative factor compared to ours. (Jin et al., 2015) provides an algorithm based on shift and invert framework that obtains the same asymptotic error as ours. However, their algorithm requires warm start with a vector that is already constant close to the top eigenvector, which itself is a hard problem. For gap free results, the recent paper (Shamir, 2015b) achieves the optimal asymptotic rate although it loses poly (d) factors with random initialization.

Arbitrary data: In this setting, each row of the data matrix is provided in an arbitrary order. Most of the existing methods here first compute a sketch of the matrix and use that to compute an estimate of the top eigenvector (Clarkson and Woodruff, 2009; Liberty, 2013; Nelson and Nguyên, 2013; Cohen et al., 2015; Ghashami et al., 2015; Boutsidis et al., 2015). However, a direct application of such techniques to the stochastic setting leads to sample complexity bounds which are larger by a multiplicative factor of $O(d)$ (ignoring other factors like variance etc). Finally, (Shamir, 2015a; Garber and Hazan, 2015; Jin et al., 2015) also provide methods for eigenvector computation, but they require multiple passes over the data and hence do not apply to the streaming setting.

Regret bounds: Here, at each step the algorithm has to output an estimate $\mathbf{w}$ of $\mathbf{v}_1$ for which we get reward of $\mathbf{w}^T A_i \mathbf{w}$ and the goal is to minimize the regret w.r.t. $\mathbf{v}_1$. The algorithms in this regime are mostly based on online convex optimization and applying them in our setting would again result in a loss of multiplicative $O(d)$. Moreover, typical algorithms in this setting are not memory efficient (Warmuth and Kuzmin, 2006; Garber et al., 2015).

1.3. Notation

We use bold lowercase letters such as $\mathbf{u}, \mathbf{v}, \mathbf{w}$ to denote vectors and bold uppercase letters such as $\mathbf{A}, \mathbf{B}, \mathbf{C}$ to denote matrices. For symmetric matrices $\mathbf{A}$ and $\mathbf{B}$ we use $\mathbf{A} \preceq \mathbf{B}$ to denote the condition that $\mathbf{x}^\top \mathbf{A} x \leq \mathbf{x}^\top \mathbf{B} \mathbf{x}$ for all $\mathbf{x}$ and define $\mathbf{B} \succeq \mathbf{A}$ analogously. We call a symmetric matrix $\mathbf{A}$ positive semidefinite if $\mathbf{A} \preceq \mathbf{0}$. For symmetric matrices $\mathbf{A}, \mathbf{B}$ we define the inner product $\langle \mathbf{A}, \mathbf{B} \rangle \triangleq \operatorname{Tr}\left(\mathbf{A}^\top \mathbf{B}\right)$.

1.4. Paper Organization

The rest of this paper is organized as follows. Section 2 introduces basic mathematical facts we use throughout the paper and also provides a proof of the error bound of the standard batch method (Theorem 2). Section 3 provides an overview of our approach to analyzing Oja's algorithm and provides the main technical result of the paper. We use this technical result in Section 4 to prove our running time for Oja's algorithm and justify our choice of step size. Section 5 presents the proof of our main technical result and we conclude in Section 6 and mention a few interesting future directions.

2. Preliminaries

Throughout this paper we make frequent use of several basic inequalities regarding power series, the exponential, and PSD matrices. We summarize the facts here

Lemma 5 (Basic Inequalities) *The following are true:*

- $1 + x \leq \exp(x)$ *for all* x
- $1 + x \geq \exp\left(x - x^2\right)$ *for all* $x \geq 0$
- $\frac{1}{1+x} \leq \sum_{i=1}^{\infty} \frac{1}{(x+i)^2} \leq \frac{1}{x}$
- $\langle \mathbf{A}, \mathbf{B} \rangle \leq \langle \mathbf{A}, \mathbf{C} \rangle$ *for PSD matrices* $\mathbf{A}, \mathbf{B}, \mathbf{C}$ *with* $\mathbf{B} \preceq \mathbf{C}$
- $\mathrm{Tr}\left(\mathbf{A}^\top \mathbf{B}\right) \leq \frac{1}{2}\mathrm{Tr}\left(\mathbf{A}^\top \mathbf{A} + \mathbf{B}^\top \mathbf{B}\right)$ *for all matrices* $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{m \times n}$.

Proof *The first inequality follows from the Taylor expansion of* $\exp(x)$. *The second comes from* $1 + 0 = \exp(0 - 0^2)$ *and* $\frac{d}{dx}(1 + x) \geq \frac{d}{dx}\exp(x - x^2)$ *for* $x \geq 0$. *The third follows by considering upper and lower Riemann sums of* $\int_{y=1}^{\infty} 1/(x + y)$. *The fourth from the fact that since* $\mathbf{A}$ *is PSD there is a matrix* $\mathbf{D}$ *with* $\mathbf{D}^\top \mathbf{D} = \mathbf{A}$ *and therefore*

$$\langle \mathbf{A}, \mathbf{B} \rangle = \mathrm{Tr}\left(\mathbf{A}^\top \mathbf{B}\right) = \mathrm{Tr}\left(\mathbf{D}\mathbf{B}\mathbf{D}^\top\right) \leq \mathrm{Tr}\left(\mathbf{D}\mathbf{C}\mathbf{D}^\top\right) = \langle \mathbf{A}, \mathbf{C} \rangle \ .$$

The final follows from Cauchy Schwarz and Young's inequality, i.e. $x \cdot y \leq \frac{1}{2}(x^2 + y^2)$ *as*

$$\mathrm{Tr}\left(\mathbf{B}^\top \mathbf{A}\right) = \sum_{i \in [n]} 1_i \mathbf{B}^\top \mathbf{A} 1_i \leq \sum_{i \in [n]} \|\mathbf{A}1_i\|_2 \cdot \|\mathbf{B}1_i\|_2 \leq \frac{1}{2} \sum_{i \in [n]} \left(\|\mathbf{A}1_i\|_2^2 + \|\mathbf{B}1_i\|_2^2\right)$$

■

We next present a matrix Bernstein based proof of the error bound of the batch method.

Proof [Proof of Theorem 2] Using Theorem 1.4 of (Tropp, 2012), we have (w.p. $\geq 1 - \delta$):

$$\left\| \frac{1}{n} \sum_{i=1}^{n} \mathbf{A}_i - \mathbf{\Sigma} \right\|_2 \leq 2 \cdot \max\left\{ \sqrt{\frac{\mathcal{V}}{n} \log \frac{d}{\delta}}, \frac{\mathcal{M}}{n} \log \frac{d}{\delta} \right\}. \tag{1}$$

Let $\widehat{\mathbf{v}}$ be the top eigenvector of $\widehat{\mathbf{\Sigma}} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{A}_i$. Then, using Wedin's theorem (Wedin, 1972), we have:

$$\sin^2\langle \mathbf{v}_1, \widehat{\mathbf{v}} \rangle \leq \frac{\left\| \frac{1}{n}\sum_{i=1}^{n} \mathbf{A}_i - \mathbf{\Sigma} \right\|_2^2}{|\lambda_1 - \lambda_2|^2}. \tag{2}$$

Theorem now follows by combining (1) and (2). ■

3. Approach

Let us now describe the approach to analyze Oja's algorithm. We provide our main theorem regarding the convergence rate of Oja's algorithm and discuss how it is proved. The details of the proof are deferred to Section 5 and the use of the theorem to choose step sizes is in Section 4.

One of the primary difficulties in analyzing Oja's algorithm, or more broadly any algorithm for streaming PCA, is choosing a suitable potential function to analyze the method. If we try to

analyze the progress of Oja's algorithm in every iteration i, by measuring the quality of $\mathbf{w}_i$, we run the risk that during the first few iterations of Oja's algorithm a step may actually yield a $\mathbf{w}_{i+1}$ that is orthogonal to $\mathbf{v}_i$. If this happens, even in the typical best case, where all future samples are $\mathbf{\Sigma}$ itself, we would still fail to converge. In short, if we do not account for the randomness of $\mathbf{w}_0$ in our potential function then it is difficult to show that a rapidly convergent algorithm does not catastrophically fail.

Rather than analyzing the convergence of $\mathbf{w}_i$ directly we instead analyze the convergence of Oja's algorithm as an operator on $\mathbf{w}_0$. Oja's algorithm simply considers the matrix

$$\mathbf{B}_n \stackrel{\triangle}{=} (\mathbf{I} + \eta_n \mathbf{A}_n)(\mathbf{I} + \eta_{n-1}\mathbf{A}_{n-1}) \cdots (\mathbf{I} + \eta_1 \mathbf{A}_1) \tag{3}$$

and outputs the normalized result of applying this matrix, $\mathbf{B}_n$, to the random initial vector, i.e.

$$\mathbf{w}_n = \frac{\mathbf{B}_n \mathbf{w}_0}{\|\mathbf{B}_n \mathbf{w}_0\|_2}\,. \tag{4}$$

Rather than analyze the improvement of $\mathbf{w}_{n+1}$ over $\mathbf{w}_n$ we analyze $\mathbf{B}_{n+1}$'s improvement over $\mathbf{B}_n$.

Another interpretation of (3) and (4) is that Oja's algorithm simply approximates $\mathbf{v}_n$ by performing 1 step of the power method on the matrix $\mathbf{B}_n$. Fortunately, analyzing when 1 step of the power method succeeds is fairly straightforward as we show below:

Lemma 6 (One Step Power Method) *Let $\mathbf{B} \in \mathbb{R}^{d\times d}$, let $\widetilde{\mathbf{v}} \in \mathbb{R}^d$ be a unit vector, and let $\widetilde{\mathbf{V}}_\perp$ be a matrix whose columns form an orthonormal basis of the subspace orthogonal to $\widetilde{\mathbf{v}}$. If $\mathbf{w} \in \mathbb{R}^d$ is chosen uniformly at random from the surface of the unit sphere then with probability at least $1-\delta$*

$$\sin^2\left(\widetilde{\mathbf{v}}, \frac{\mathbf{B}\mathbf{w}}{\|\mathbf{B}\mathbf{w}\|_2}\right) = 1 - \left(\frac{\widetilde{\mathbf{v}}^\top \mathbf{B}\mathbf{w}}{\|\mathbf{B}\mathbf{w}\|_2}\right)^2 \le \frac{C\log(1/\delta)}{\delta^2}\frac{\mathrm{Tr}\left(\widetilde{\mathbf{V}}_\perp^\top \mathbf{B}\mathbf{B}^\top \widetilde{\mathbf{V}}_\perp\right)}{\widetilde{\mathbf{v}}^\top \mathbf{B}\mathbf{B}^\top \widetilde{\mathbf{v}}}$$

where C is an absolute constant.

Proof As $\mathbf{w}$ is distributed uniformly over the sphere, we have: $\mathbf{w} = \mathbf{g}/\|\mathbf{g}\|_2$ where $\mathbf{g} \sim N(0, I)$. Consequently, with probability at least $1-\delta$

$$\begin{aligned}1 - \left(\frac{\widetilde{\mathbf{v}}^\top \mathbf{B}\mathbf{w}}{\|\mathbf{B}\mathbf{w}\|_2}\right)^2 = \frac{\mathbf{g}^\top \mathbf{B}^\top (\mathbf{I} - \widetilde{\mathbf{v}}\widetilde{\mathbf{v}}^\top)\mathbf{B}\mathbf{g}}{\mathbf{g}^\top \mathbf{B}^\top \mathbf{B}\mathbf{g}} &\overset{\zeta_1}{\le} \frac{C_1}{\delta^2}\frac{\mathbf{g}^\top \mathbf{B}^\top (\mathbf{I} - \widetilde{\mathbf{v}}\widetilde{\mathbf{v}}^\top)\mathbf{B}\mathbf{g}}{\widetilde{\mathbf{v}}^\top \mathbf{B}\mathbf{B}^\top \widetilde{\mathbf{v}}} \\ &\overset{\zeta_2}{\le} \frac{C\log(1/\delta)}{\delta^2}\frac{\mathrm{Tr}\left(\mathbf{B}^\top (\mathbf{I} - \widetilde{\mathbf{v}}\widetilde{\mathbf{v}}^\top)\mathbf{B}\right)}{\widetilde{\mathbf{v}}^\top \mathbf{B}\mathbf{B}^\top \widetilde{\mathbf{v}}},\end{aligned}$$

where C_1 and C are absolute constants. ζ_1 follows as $\mathbf{g}^\top \mathbf{B}^\top \mathbf{B}\mathbf{g} \ge (\widetilde{\mathbf{v}}^\top \mathbf{B}\mathbf{g})^2 \ge \frac{\delta^2}{C_1}\widetilde{\mathbf{v}}^\top \mathbf{B}\mathbf{B}^\top \widetilde{\mathbf{v}}$ where the second inequality follows from the fact that $\widetilde{\mathbf{v}}^\top \mathbf{B}\mathbf{g}$ is a Gaussian random variable with variance $\left\|\mathbf{B}^\top \widetilde{\mathbf{v}}\right\|_2^2$ and $Pr(|g| \le \delta) \le C\delta$ for a normal random variable $g \sim N(0,1)$. Similarly, ζ_2 follows from the fact that$\mathbf{g}^\top \mathbf{B}^\top (\mathbf{I} - \widetilde{\mathbf{v}}\widetilde{\mathbf{v}}^\top)\mathbf{B}\mathbf{g}$ is a χ^2 random variable with $\mathrm{Tr}\left(\mathbf{B}^\top (\mathbf{I} - \widetilde{\mathbf{v}}\widetilde{\mathbf{v}}^\top)\mathbf{B}\right)$-degrees of freedom. ■

This lemma makes our goal clear. To show that Oja's algorithm succeeds we simply need to show that with constant probability $\mathbf{v}_1^\top \mathbf{B}_n \mathbf{B}_n^\top \mathbf{v}_1$ is relatively large and $\mathrm{Tr}\left(\mathbf{V}_\perp \mathbf{B}_n \mathbf{B}_n^\top \mathbf{V}_\perp\right)$ is relatively small, where $\mathbf{V}_\perp$ is a matrix whose columns form an orthonormal basis of the subspace

orthogonal to $\mathbf{v}_1$. This immediately alleviates the issues of catastrophic failure that plagued analyzing $\mathbf{w}_n$. So long as we pick η_i sufficiently small, i.e. $\eta_i = O(1/\max\{\mathcal{M}, \lambda_1\})$ then $\mathbf{I} + \eta_i \mathbf{A}_i$ is invertible. In this case $\mathbf{B}_n \mathbf{B}_n^\top$ is invertible and $\mathbf{v}_1^\top \mathbf{B}_n \mathbf{B}_n^\top \mathbf{v}_1 > 0$. In short, so long as we pick η_i sufficiently small the quantity we wish to bound $\mathrm{Tr}\left(\mathbf{V}_\perp \mathbf{B}_n \mathbf{B}_n^\top \mathbf{V}_\perp\right) / \mathbf{v}_1^\top \mathbf{B}_n \mathbf{B}_n^\top \mathbf{v}_1$ is always finite.

To actually bound $\mathbf{v}_1^\top \mathbf{B}_n \mathbf{B}_n^\top \mathbf{v}_1$ and $\mathrm{Tr}\left(\mathbf{V}_\perp^\top \mathbf{B}_n \mathbf{B}_n^\top \mathbf{V}_\perp\right)$ we split the analysis into several parts in Section 5. First, we show that $\mathbb{E}\left[\mathrm{Tr}\left(\mathbf{V}_\perp^\top \mathbf{B}_n \mathbf{B}_n^\top \mathbf{V}_\perp\right)\right]$ is small, which implies by Markov's inequality that $\mathrm{Tr}\left(\mathbf{V}_\perp^\top \mathbf{B}_n \mathbf{B}_n^\top \mathbf{V}_\perp\right)$ is small with constant probability. Then, we show that $\mathbb{E}\mathbf{v}_1^\top \mathbf{B}_n \mathbf{B}_n^\top \mathbf{v}_1$ is large and that $\mathrm{Var}\left[\mathbf{v}_1^\top \mathbf{B}_n \mathbf{B}_n^\top \mathbf{v}_1\right]$ is small. By Chebyshev's inequality this implies that $\mathbf{v}_1^\top \mathbf{B}_n \mathbf{B}_n^\top \mathbf{v}_1$ is large with constant probability. Putting these together we achieve the main technical result regarding the analysis of Oja's method. Once we devise this roadmap, the proof is fairly straightforward.

Theorem 7 (Oja's Algorithm Convergence Rate) *Let $\delta > 0$ and step sizes $\eta_i \leq \frac{1}{4 \cdot \max\{M, \lambda_1\}}$. The output $\mathbf{w_n}$ of Algorithm 1 is an ϵ-approximation to $\mathbf{v}_1$ with probability at least $1 - \delta$ where*

$$\epsilon \leq \frac{1}{Q} \exp\left(5\overline{\mathcal{V}} \sum_{i \in [n]} \eta_i^2\right) \left(d \cdot \exp\left(-2(\lambda_1 - \lambda_2) \sum_{i \in [n]} \eta_i\right) + \mathcal{V} \sum_{i=1}^{n} \eta_i^2 \exp\left(-\sum_{j=i+1}^{n} 2\eta_j(\lambda_1 - \lambda_2)\right)\right)$$

where $Q \triangleq \frac{\delta^3}{C \log(1/\delta)} \left(1 - \frac{1}{\sqrt{\delta}} \sqrt{\exp\left(18\overline{\mathcal{V}} \sum_{i=1}^{n} \eta_i^2\right) - 1}\right)$, $\overline{\mathcal{V}} \triangleq \mathcal{V} + \lambda_1^2$, and C is an absolute constant.

Theorem 7 is proved in Section 5. Theorem 7 serves as the basis for our results regarding Oja's algorithm. In the next section we show how to use this theorem to choose step sizes and achieve the main results of this paper.

4. Our Results

Here we show how to use Theorem 7 presented in the previous section to prove the main result of our paper. The theorem and proof are below and essentially consist of choosing appropriate parameters to efficiently apply Theorem 7. Once we have this theorem, Theorems 3 and 4 follow by choosing $\alpha = \log d$ and $\alpha = 6$ respectively.

Theorem 8 *Fix any $\delta > 0$ and suppose the stepsizes are set to $\eta_t = \frac{\alpha}{(\lambda_1 - \lambda_2)(\beta + t)}$ for $\alpha > \frac{1}{2}$ and*

$$\beta \triangleq 20 \max\left(\frac{\mathcal{M}\alpha}{(\lambda_1 - \lambda_2)}, \frac{\left(\mathcal{V} + (\lambda_1)^2\right)\alpha^2}{(\lambda_1 - \lambda_2)^2 \log\left(1 + \frac{\delta}{100}\right)}\right).$$

Suppose the number of samples $n > \beta$. Then the output $\mathbf{w_n}$ of Algorithm 1 satisfies:

$$1 - (\mathbf{w_n}^\top \mathbf{v}_1)^2 \leq \frac{C \log(1/\delta)}{\delta^3} \left(d\left(\frac{\beta}{n}\right)^{2\alpha} + \frac{\alpha^2 \mathcal{V}}{(2\alpha - 1)(\lambda_1 - \lambda_2)^2} \cdot \frac{1}{n}\right),$$

with probability at least $1 - \delta$. Here C is an absolute numerical constant.

Proof Recall that Theorem 7 gives a bound of

$$\frac{1}{Q}\exp\left(5\overline{\mathcal{V}}\sum_{i\in[n]}\eta_i^2\right)\left(d\cdot\exp\left(-2(\lambda_1-\lambda_2)\sum_{i\in[n]}\eta_i\right)+\mathcal{V}\sum_{i=1}^{n}\eta_i^2\exp\left(-\sum_{j=i+1}^{n}2\eta_j(\lambda_1-\lambda_2)\right)\right) \tag{5}$$

where $Q \triangleq \frac{\delta^2}{C\log(1/\delta)}\left(1-\frac{1}{\sqrt{\delta}}\sqrt{\exp\left(18\overline{\mathcal{V}}\sum_{i=1}^{n}\eta_i^2\right)-1}\right)$. Since $\eta_i = \frac{\alpha}{(\lambda_1-\lambda_2)(\beta+i)}$, we have $\sum_{i\in[n]}\eta_i^2 \le \frac{\alpha^2}{(\lambda_1-\lambda_2)^2\beta}$ and by our assumption that $\frac{\overline{\mathcal{V}}\alpha^2}{(\lambda_1-\lambda_2)^2\beta} \le \frac{1}{18}\log\left(1+\frac{\delta}{100}\right)$, we have:

$$\exp\left(18\overline{\mathcal{V}}\sum_{i\in[n]}\eta_i^2\right)\le\sqrt{2} \quad\Rightarrow\quad Q\ge\frac{\delta^2}{C\log(1/\delta)}. \tag{6}$$

Moreover, since $\sum_{i\in[n]}\eta_i \ge \frac{\alpha}{\lambda_1-\lambda_2}\log\left(1+n/\beta\right)$, we have

$$\exp\left(-2(\lambda_1-\lambda_2)\sum_{i\in[n]}\eta_i\right)\le\left(\frac{\beta}{\beta+n}\right)^{2\alpha}. \tag{7}$$

Note that $\sum_{j=i+1}^{n}\eta_j \le \frac{\alpha}{\lambda_1-\lambda_2}\log\frac{n+\beta+1}{i+\beta+1}$. Moreover, as $\alpha > 1/2$, we have:

$$\begin{aligned}
&\sum_{i=1}^{n}\eta_i^2\exp\left(-2(\lambda_1-\lambda_2)\sum_{j=i+1}^{n}\eta_j\right)\\
&\le\frac{\alpha^2}{(\lambda_1-\lambda_2)^2}\sum_{i=1}^{n}\frac{1}{(\beta+i)^2}\exp\left(2\alpha\log\frac{i+\beta+1}{n+\beta+1}\right),\\
&\le\frac{(\beta+1)^2}{\beta^2}\cdot\frac{\alpha^2}{(\lambda_1-\lambda_2)^2(n+\beta+1)^{2\alpha}}\cdot\sum_{i=1}^{n}(i+\beta+1)^{2\alpha-2},\\
&\le\frac{2\alpha^2}{(2\alpha-1)(\lambda_1-\lambda_2)^2(n+\beta+1)}\quad\left(\text{since }\alpha>1/2\text{ and }\sum_{i=1}^{n}i^{\gamma}\le n^{\gamma+1}/(\gamma+1)\ \forall\ \gamma>-1\right).
\end{aligned} \tag{8}$$

Substituting (6), (7) and (8) into (5) proves the theorem. ■

5. Bounding the Convergence of Oja's Algorithm

In this section, we present a detailed proof of Theorem 7. The proof follows the approach outlined in Section 3 and uses the notation of that section, i.e.

- We let $\mathbf{B}_n \triangleq (\mathbf{I}+\eta_n\mathbf{A}_n)\cdots(\mathbf{I}+\eta_1\mathbf{A}_1)$ with $\mathbf{B}_0 \triangleq \mathbf{I}$
- We let $\overline{\mathcal{V}} \triangleq \mathcal{V}+\lambda_1^2$

- We let $\mathbf{V}_\perp \in \mathbb{R}^{d\times d-1}$ denote a matrix whose columns form an orthonormal basis for the subspace orthogonal to $\mathbf{v}_1$.

We first provide several technical lemmas bounding the expected behavior of $\mathbf{B}_n$ and ultimately use these lemmas to prove Theorem 7. We begin with a straightforward lemma bounding the rate of increase of $\mathbb{E}\left[\mathbf{B}_t\mathbf{B}_t^\top\right]$ in spectral norm.

Lemma 9 *For all $t \geq 0$ and $\eta_i \geq 0$ we have*

$$\left\|\mathbb{E}\left[\mathbf{B}_t\mathbf{B}_t^\top\right]\right\|_2 \leq \exp\left(\sum_{i\in[t]} 2\eta_i\lambda_1 + \eta_i^2\overline{\mathcal{V}}\right) .$$

Proof Let $\alpha_t \triangleq \|\mathbb{E}\left[\mathbf{B}_t\mathbf{B}_t^\top\right]\|_2$, i.e., $\mathbb{E}\left[\mathbf{B}_t\mathbf{B}_t^\top\right] \preceq \alpha_t\mathbf{I}$. For all $t > 0$,

$$\begin{aligned}\mathbb{E}\left[\mathbf{B}_t\mathbf{B}_t^\top\right] &= \mathbb{E}\left[(\mathbf{I}+\eta_t\mathbf{A}_t)\mathbf{B}_{t-1}\mathbf{B}_{t-1}^\top(\mathbf{I}+\eta_t\mathbf{A}_t)^\top\right] \preceq \alpha_{t-1}\mathbb{E}\left[(\mathbf{I}+\eta_t\mathbf{A}_t)(\mathbf{I}+\eta_t\mathbf{A}_t^\top)\right],\\ &= \alpha_{t-1}\mathbb{E}\left[\mathbf{I}+\eta_t\mathbf{A}_t+\eta_t\mathbf{A}_t^\top+\eta_t^2\mathbf{A}_t\mathbf{A}_t^\top\right] \preceq \alpha_{t-1}\left[\mathbf{I}+2\eta_t\mathbf{\Sigma}+\eta_t^2(\mathbf{\Sigma}^2+V\mathbf{I})\right], \quad (9)\end{aligned}$$

where the last inequality follows from $\mathbb{E}\left[A_t\right] = \mathbf{\Sigma}$ and,

$$\mathbb{E}\left[\mathbf{A}_t\mathbf{A}_t^\top\right] = \mathbf{\Sigma}^2 + \mathbb{E}\left[(\mathbf{A}_t-\mathbf{\Sigma})(\mathbf{A}_t-\mathbf{\Sigma})^\top\right] \preceq \mathbf{\Sigma}^2 + V\mathbf{I}\,.$$

Using (9) along with $\|\mathbb{E}\left[\mathbf{B}_t\mathbf{B}_t^\top\right]\|_2 = \alpha_t$, $\mathbf{\Sigma} \preceq \lambda_1\mathbf{I}$, and $\mathbf{\Sigma}^2 \preceq \lambda_1^2\mathbf{I}$, we have for $\forall t > 0$:

$$\alpha_t \leq (1 + 2\eta_t\lambda_1 + \eta_t^2(\lambda_1^2+V))\alpha_{t-1}.$$

The result follows by using induction along with $\alpha_0 = 1$ and $1 + x \leq e^x$. ■

Using Lemma 9 we next bound the expected value of $\mathrm{Tr}\left(\mathbf{V}_\perp^\top\mathbf{B}_n\mathbf{B}_n^\top\mathbf{V}_\perp\right)$. Ultimately this will allow us to bound the value $\mathrm{Tr}\left(\mathbf{V}_\perp^\top\mathbf{B}_n\mathbf{B}_n^\top\mathbf{V}_\perp\right)$ with by Markov's inequality.

Lemma 10 *For all $t \geq 0$ and $\eta_i \leq \frac{1}{\lambda_1}$ the following holds*

$$\mathbb{E}\left[\mathrm{Tr}\left(\mathbf{V}_\perp^\top\mathbf{B}_t\mathbf{B}_t^\top\mathbf{V}_\perp\right)\right] \leq \exp\left(\sum_{j\in[t]} 2\eta_j\lambda_2 + \eta_j^2\overline{\mathcal{V}}\right)\cdot\left(d + \mathcal{V}\sum_{i=1}^{t}\eta_i^2\exp\left(\sum_{j\in[i]} 2\eta_j(\lambda_1-\lambda_2)\right)\right) .$$

Proof Let $\alpha_t \triangleq \mathbb{E}\left[\mathrm{Tr}\left(\mathbf{V}_\perp^\top\mathbf{B}_t\mathbf{B}_t^\top\mathbf{V}_\perp\right)\right]$. We first simplify α_t as follows:

$$\alpha_t = \left\langle\mathbb{E}\left[\mathbf{B}_t\mathbf{B}_t^\top\right], \mathbf{V}_\perp\mathbf{V}_\perp^\top\right\rangle = \left\langle\mathbb{E}\left[\mathbf{B}_{t-1}\mathbf{B}_{t-1}^\top\right], \mathbb{E}\left[(\mathbf{I}+\eta_t\mathbf{A}_t)\,\mathbf{V}_\perp\mathbf{V}_\perp^\top\left(\mathbf{I}+\eta_t\mathbf{A}_t^\top\right)\right]\right\rangle . \quad (10)$$

Recall that $\mathbb{E}\left[\mathbf{A}_t\right] = \mathbf{\Sigma}$. Now, the second term on the right hand side can be bounded as follows:

$$\begin{aligned}
&\mathbb{E}\left[\left(\mathbf{I}+\eta_t\mathbf{A}_t\right)\mathbf{V}_\perp\mathbf{V}_\perp^\top\left(\mathbf{I}+\eta_t\mathbf{A}_t^\top\right)\right],\\
&= \mathbf{V}_\perp\mathbf{V}_\perp^\top + \eta_t\mathbf{\Sigma}\mathbf{V}_\perp\mathbf{V}_\perp^\top + \eta_t\mathbf{V}_\perp\mathbf{V}_\perp^\top\mathbf{\Sigma} + \eta_t^2\mathbb{E}\left[\mathbf{A}_t\mathbf{V}_\perp\mathbf{V}_\perp^\top\mathbf{A}_t^\top\right],\\
&= \mathbf{V}_\perp\mathbf{V}_\perp^\top + \eta_t\mathbf{\Sigma}\mathbf{V}_\perp\mathbf{V}_\perp^\top + \eta_t\mathbf{V}_\perp\mathbf{V}_\perp^\top\mathbf{\Sigma} + \eta_t^2\mathbf{\Sigma}\mathbf{V}_\perp\mathbf{V}_\perp^\top\mathbf{\Sigma} + \eta_t^2\mathbb{E}\left[\left(\mathbf{A}_t-\mathbf{\Sigma}\right)\mathbf{V}_\perp\mathbf{V}_\perp^\top\left(\mathbf{A}_t-\mathbf{\Sigma}\right)^\top\right],\\
&\overset{\zeta_1}{\preceq} \mathbf{V}_\perp\mathbf{V}_\perp^\top + 2\eta_t\lambda_2\mathbf{V}_\perp\mathbf{V}_\perp^\top + \eta_t^2\lambda_2^2\mathbf{V}_\perp\mathbf{V}_\perp^\top + \eta_t^2\mathbb{E}\left[\left(\mathbf{A}_t-\mathbf{\Sigma}\right)\left(\mathbf{A}_t-\mathbf{\Sigma}\right)^\top\right],\\
&\overset{\zeta_2}{\preceq} \left(1+2\eta_t\lambda_2+\eta_t^2\lambda_2^2\right)\mathbf{V}_\perp\mathbf{V}_\perp^\top + \eta_t^2\mathcal{V}\mathbf{I} = \left(1+2\eta_t\lambda_2+\eta_t^2\lambda_2^2+\eta_t^2\mathcal{V}\right)\mathbf{V}_\perp\mathbf{V}_\perp^\top + \eta_t^2\mathcal{V}\cdot\mathbf{v}_1\mathbf{v}_1^\top,
\end{aligned}$$

where ζ_1 follows from the fact that $\mathbf{V}_\perp$ is orthogonal to $\mathbf{v}_1$ and ζ_2 follows from defintion of $\mathcal{V}$.

Plugging the above into (10), we get for all $t \geq 1$,

$$\begin{aligned}
\alpha_t &\leq \left(1+2\eta_t\lambda_2+\eta_t^2(\lambda_2^2+\mathcal{V})\right)\left\langle\mathbb{E}\left[\mathbf{B}_{t-1}\mathbf{B}_{t-1}^\top\right],\mathbf{V}_\perp\mathbf{V}_\perp^\top\right\rangle + \eta_t^2\mathcal{V}\left\langle\mathbb{E}\left[\mathbf{B}_{t-1}\mathbf{B}_{t-1}^\top\right],\mathbf{v}_1\mathbf{v}_1^\top\right\rangle,\\
&\leq \left(1+2\eta_t\lambda_2+\eta_t^2\overline{\mathcal{V}}\right)\alpha_{t-1} + \eta_t^2\mathcal{V}\left\|\mathbb{E}\left[\mathbf{B}_{t-1}\mathbf{B}_{t-1}^\top\right]\right\|_2,\\
&\leq \exp\left(2\eta_t\lambda_2+\eta_t^2\overline{\mathcal{V}}\right)\alpha_{t-1} + \eta_t^2\mathcal{V}\exp\left(\sum_{i\in[t-1]}\eta_i\lambda_1+\eta_i^2\overline{\mathcal{V}}\right),
\end{aligned}$$

where the last inequality follows from $1+x\leq e^x$ and using Lemma 9.

Recursing the above inequality, we obtain

$$\begin{aligned}
\alpha_t &\leq \sum_{i\in[t]}\eta_i^2\mathcal{V}\exp\left(\sum_{j=i+1}^{t}2\eta_j\lambda_2+\eta_j^2\overline{\mathcal{V}}\right)\exp\left(\sum_{j\in[i]}2\eta_j\lambda_1+\eta_j^2\overline{\mathcal{V}}\right) + \exp\left(\sum_{j\in[t]}2\eta_j\lambda_2+\eta_j^2\overline{\mathcal{V}}\right)\alpha_0,\\
&\leq \exp\left(\sum_{j\in[t]}2\eta_j\lambda_2+\eta_j^2\overline{\mathcal{V}}\right)\left(\alpha_0+\mathcal{V}\sum_{i=1}^{t}\eta_i^2\exp\left(\sum_{j\in[i]}2\eta_j(\lambda_1-\lambda_2)+\eta_j^2\overline{\mathcal{V}}\right)\right)
\end{aligned}$$

Since $\mathbf{B}_0 = \mathbf{I}$ we see that $\alpha_0 = d-1 \leq d$. Using that $\eta_i \leq \frac{1}{\lambda_1} \leq \frac{1}{\lambda_2}$ completes the proof. ■

Next we provide the lemmas that will allow us to lower bound $\mathbf{v}_1^\top\mathbf{B}_t\mathbf{B}_t^\top\mathbf{v}_1$. In Lemma 11 we lower bound $\mathbb{E}\left[\mathbf{v}_1^\top\mathbf{B}_t\mathbf{B}_t^\top\mathbf{v}_1\right]$ and in Lemma 12 we upper bound $\mathrm{Var}\left[\mathbf{v}_1^\top\mathbf{B}_t\mathbf{B}_t^\top\mathbf{v}_1\right]$. Ultimately, the lower bound follows using Chebyshev's inequality.

Lemma 11 *For all $t \geq 0$ and $\eta_i \geq 0$ we have*

$$\mathbb{E}\left[\mathbf{v}_1^\top\mathbf{B}_t\mathbf{B}_t^\top\mathbf{v}_1\right] \geq \exp\left(\sum_{i\in[t]}2\eta_i\lambda_1 - 4\eta_i^2\lambda_1^2\right)$$

If we further assume that $\eta_i \leq \frac{1}{4\cdot\max\{\lambda_1,M\}}$ then $\mathbb{E}\left[\mathbf{v}_1^\top\mathbf{B}_t\mathbf{B}_t^\top\mathbf{v}_1\right] \geq \exp(\lambda_1\sum_{i\in[t]}\eta_i)$.

Proof Let $\beta_t \triangleq \mathbb{E}\left[\mathbf{v}_1^\top \mathbf{B}_t \mathbf{B}_t^\top \mathbf{v}_1\right]$. Since $\mathbf{B}_t = (\mathbf{I} + \eta_t \mathbf{A}_t)\,\mathbf{B}_{t-1}$, we can bound β_t as

$$\begin{aligned}
\beta_t &= \left\langle \mathbb{E}\left[\mathbf{B}_{t-1}\mathbf{B}_{t-1}^\top\right], \mathbb{E}\left[(\mathbf{I}+\eta_t\mathbf{A}_t)\,\mathbf{v}_1\mathbf{v}_1^\top\left(\mathbf{I}+\eta_t\mathbf{A}_t^\top\right)\right]\right\rangle \\
&= \left\langle \mathbb{E}\left[\mathbf{B}_{t-1}\mathbf{B}_{t-1}^\top\right], \mathbf{v}_1\mathbf{v}_1^\top + \eta_t\mathbf{\Sigma}\mathbf{v}_1\mathbf{v}_1^\top + \eta_t\mathbf{v}_1\mathbf{v}_1^\top\mathbf{\Sigma} + \eta_t^2\mathbb{E}\left[\mathbf{A}_t\mathbf{v}_1\mathbf{v}_1^\top\mathbf{u}^{*\top}\mathbf{A}_t^\top\right]\right\rangle \\
&\geq \left\langle \mathbb{E}\left[\mathbf{B}_{t-1}\mathbf{B}_{t-1}^\top\right], \mathbf{v}_1\mathbf{v}_1^\top + \lambda_1\eta_t\mathbf{v}_1\mathbf{v}_1^\top + \lambda_1\eta_t\mathbf{v}_1\mathbf{v}_1^\top\right\rangle\ .
\end{aligned}$$

Consequently $\beta_t \geq (1+2\eta_t\lambda_1)\beta_{t-1}$. Furthermore, $\mathbf{B}_0 = \mathbf{I}$ and hence $\beta_0 = \|\mathbf{v}_1\|_2^2 = 1$. Proceeding by induction and using that $1 + x \geq \exp(x - x^2)$ for all $x \geq 0$ finishes the proof. ■

Lemma 12 *For $t \geq 0$ suppose that $\eta_i \leq \frac{1}{4\cdot\max\{\lambda_1, M\}}$ for all $i \in [t]$ then.*

$$\mathbb{E}\left[\left(\mathbf{v}_1^\top\mathbf{B}_t\mathbf{B}_t^\top\mathbf{v}_1\right)^2\right] \leq \exp\left(\sum_{i\in[t]} 4\eta_i\lambda_1 + 10\eta_i^2\overline{\mathcal{V}}\right)$$

Proof Let $\mathbf{W}_{t,s} \triangleq (\mathbf{I}+\eta_t\mathbf{A}_t)\cdots(\mathbf{I}+\eta_{t-s+1}\mathbf{A}_{t-s+1})$ and $\gamma_s \triangleq \mathbb{E}\left[\left(\mathbf{v}_1^\top\mathbf{W}_{t,s}\mathbf{W}_{t,s}^\top\mathbf{v}_1\right)^2\right]$. Note that $\mathbf{W}_{t,t} = \mathbf{B}_t$ and $\gamma_t = \mathbb{E}\left[\mathbf{v}_1^\top\mathbf{B}_t\mathbf{B}_t^\top\mathbf{v}_1\right]$. Now,

$$\begin{aligned}
\gamma_t &= \mathrm{Tr}\left(\mathbb{E}\left[\mathbf{W}_{t,t}^\top\mathbf{v}_1\mathbf{v}_1^\top\mathbf{W}_{t,t}\mathbf{W}_{t,t}^\top\mathbf{v}_1\mathbf{v}_1^\top\mathbf{W}_{t,t}\right]\right) \\
&= \mathrm{Tr}\left(\mathbb{E}\left[(\mathbf{I}+\eta_1\mathbf{A}_1^\top)\mathbf{W}_{t,t-1}^\top\mathbf{v}_1\mathbf{v}_1^\top\mathbf{W}_{t,t-1}(\mathbf{I}+\eta_1\mathbf{A}_1)(\mathbf{I}+\eta_1\mathbf{A}_1^\top)\mathbf{W}_{t,t-1}^\top\mathbf{v}_1\mathbf{v}_1^\top\mathbf{W}_{t,t-1}(\mathbf{I}+\eta_1\mathbf{A}_1)\right]\right) \\
&= \mathrm{Tr}\left(\mathbb{E}\left[(\mathbf{I}+\eta_1\mathbf{A}_1^\top)\mathbf{G}_{t-1}(\mathbf{I}+\eta_1\mathbf{A}_1)(\mathbf{I}+\eta_1\mathbf{A}_1^\top)\mathbf{G}_{t-1}(\mathbf{I}+\eta_1\mathbf{A}_1)\right]\right), \qquad (11)
\end{aligned}$$

where $\mathbf{G}_{t-1} \triangleq \mathbf{W}_{t,t-1}^\top\mathbf{v}_1\mathbf{v}_1^\top\mathbf{W}_{t,t-1}$. In order to bound the above quantity, we first bound the above expression for an arbitrary $\mathbf{G}_{t-1} \equiv \mathbf{G}$. We then take an expectation over only $\mathbf{A}_1$ and then finally take an expectation over $\mathbf{G}_{t-1}$. That is, for an arbitrary fixed symmetric matrix $\mathbf{G}$, we have:

$$\begin{aligned}
&\mathrm{Tr}\left(\mathbb{E}\left[\left(\mathbf{I}+\eta_1\mathbf{A}_1^\top\right)\mathbf{G}\left(\mathbf{I}+\eta_1\mathbf{A}_1\right)\left(\mathbf{I}+\eta_1\mathbf{A}_1^\top\right)\mathbf{G}\left(\mathbf{I}+\eta_1\mathbf{A}_1\right)\right]\right) \\
&= \mathrm{Tr}\left(\mathbb{E}\left[\left(\mathbf{G}+\eta_1\mathbf{A}_1^\top\mathbf{G}+\eta_1\mathbf{G}\mathbf{A}_1+\eta_1^2\mathbf{A}_1^\top\mathbf{G}\mathbf{A}_1\right)^2\right]\right) \\
&= \mathrm{Tr}\left(\mathbf{G}^2 + \eta_1\mathbb{E}\left[\mathbf{A}_1^\top\right]\mathbf{G}^2 + \eta_1\mathbf{G}^2\mathbb{E}\left[\mathbf{A}_1\right] + \eta_1\mathbf{G}\left(\mathbb{E}\left[\mathbf{A}_1\right] + \mathbb{E}\left[\mathbf{A}_1^\top\right]\right)\mathbf{G}\right. \\
&\quad + \eta_1^2\mathbb{E}\left[\mathbf{A}_1^\top\mathbf{G}\mathbf{A}_1\mathbf{G}\right] + \eta_1^2\mathbb{E}\left[\mathbf{A}_1^\top\mathbf{G}\mathbf{A}_1^\top\mathbf{G}\right] + \eta_1^2\mathbb{E}\left[\mathbf{G}\mathbf{A}_1\mathbf{G}\mathbf{A}_1\right] + \eta_1^2\mathbb{E}\left[\mathbf{G}\mathbf{A}_1^\top\mathbf{G}\mathbf{A}_1\right] \\
&\quad + \eta_1^2\mathbf{G}\mathbb{E}\left[\mathbf{A}_1\mathbf{A}_1^\top\right]\mathbf{G} + \eta_1^2\mathbb{E}\left[\mathbf{A}_1^\top\mathbf{G}^2\mathbf{A}_1\right] + \eta_1^3\mathbb{E}\left[\mathbf{A}_1^\top\mathbf{G}\left(\mathbf{A}_1+\mathbf{A}_1^\top\right)\mathbf{G}\mathbf{A}_1\right] \\
&\quad \left. + \eta_1^3\mathbb{E}\left[\mathbf{A}_1^\top\mathbf{G}\mathbf{A}_1\mathbf{A}_1^\top\mathbf{G}\right] + \eta_1^3\mathbb{E}\left[\mathbf{G}\mathbf{A}_1\mathbf{A}_1^\top\mathbf{G}\mathbf{A}_1\right] + \eta_1^4\mathbb{E}\left[\mathbf{A}_1^\top\mathbf{G}\mathbf{A}_1\mathbf{A}_1^\top\mathbf{G}\mathbf{A}_1\right]\right) \\
&= \mathrm{Tr}\left(\mathbf{G}^2\right) + 4\eta_1\mathrm{Tr}\left(\mathbf{\Sigma}\mathbf{G}^2\right) + 2\eta_1^2\mathrm{Tr}\left(\mathbb{E}\left[\mathbf{A}_1\mathbf{A}_1^\top\right]\mathbf{G}^2\right) + \eta_1^2\mathrm{Tr}\left(\mathbb{E}\left[\mathbf{A}_1^\top\mathbf{G}\mathbf{A}_1\mathbf{G}\right]\right) \\
&\quad + \eta_1^2\mathrm{Tr}\left(\mathbb{E}\left[\mathbf{A}_1^\top\mathbf{G}\mathbf{A}_1^\top\mathbf{G}\right]\right) + \eta_1^2\mathrm{Tr}\left(\mathbb{E}\left[\mathbf{G}\mathbf{A}_1\mathbf{G}\mathbf{A}_1\right]\right) + \eta_1^2\mathrm{Tr}\left(\mathbb{E}\left[\mathbf{G}\mathbf{A}_1^\top\mathbf{G}\mathbf{A}_1\right]\right) \\
&\quad + 2\eta_1^3\mathrm{Tr}\left(\mathbb{E}\left[\mathbf{A}_1^\top\mathbf{G}\left(\mathbf{A}_1+\mathbf{A}_1^\top\right)\mathbf{G}\mathbf{A}_1\right]\right) + \eta_1^4\mathrm{Tr}\left(\mathbb{E}\left[\mathbf{A}_1^\top\mathbf{G}\mathbf{A}_1\mathbf{A}_1^\top\mathbf{G}\mathbf{A}_1\right]\right) \qquad (12)
\end{aligned}$$

We now bound the various terms above as follows. Each of the second order terms can be bounded using Lemma 5 as follows:

$$\mathbb{E}\left[\mathrm{Tr}\left(\mathbf{A}_1^\top \mathbf{G}\mathbf{A}_1\mathbf{G}\right)\right] \le \frac{1}{2}\mathbb{E}\left[\left\|\mathbf{A}_1^\top \mathbf{G}\right\|_F^2 + \|\mathbf{A}_1\mathbf{G}\|_F^2\right] \\ = \frac{1}{2}\left(\mathrm{Tr}\left(\mathbf{G}\mathbb{E}\left[\mathbf{A}_1\mathbf{A}_1^\top\right]\mathbf{G} + \mathbf{G}\mathbb{E}\left[\mathbf{A}_1^\top\mathbf{A}_1\right]\mathbf{G}\right)\right) \le (\mathcal{V}+\lambda_1^2)\mathrm{Tr}\left(\mathbf{G}^2\right). \tag{13}$$

The third order terms can be bounded as follows:

$$\mathbb{E}\left[\mathrm{Tr}\left(\mathbf{A}_1^\top \mathbf{G}\mathbf{A}_1\mathbf{G}\mathbf{A}_1\right)\right] \le \mathbb{E}\left[\|\mathbf{A}_1\|_2 \mathrm{Tr}\left(\mathbf{A}_1^\top \mathbf{G}\mathbf{G}\mathbf{A}_1\right)\right] \\ \le (\mathcal{M}+\lambda_1)\,\mathrm{Tr}\left(\mathbf{G}\mathbb{E}\left[\mathbf{A}_1\mathbf{A}_1^\top\right]\mathbf{G}\right) \le (\mathcal{M}+\lambda_1)\,\overline{\mathcal{V}}\cdot\mathrm{Tr}\left(\mathbf{G}^2\right). \tag{14}$$

where we used the assumption that $\|\mathbf{A}_1\|_2 \le \|\mathbf{A}_1 - \mathbf{\Sigma}\|_2 + \|\mathbf{\Sigma}\|_2 \le \mathcal{M}+\lambda_1$ with probability 1. Finally the fourth order term can be bounded as

$$\mathrm{Tr}\left(\mathbb{E}\left[\mathbf{A}_1^\top \mathbf{G}\mathbf{A}_1\mathbf{A}_1^\top \mathbf{G}\mathbf{A}_1\right]\right) \le (\mathcal{M}+\lambda_1)^2\,\mathrm{Tr}\left(\mathbf{G}^2\mathbb{E}\left[\mathbf{A}_1\mathbf{A}_1^\top\right]\right) \le (\mathcal{M}+\lambda_1)^2\,\overline{\mathcal{V}}\cdot\mathrm{Tr}\left(\mathbf{G}^2\right). \tag{15}$$

Plugging (13), (14) and (15) into (12) tells us that

$$\begin{aligned}
&\mathrm{Tr}\left(\mathbb{E}\left[\left(\mathbf{I}+\eta_1\mathbf{A}_1^\top\right)\mathbf{G}\left(\mathbf{I}+\eta_1\mathbf{A}_1\right)\left(\mathbf{I}+\eta_1\mathbf{A}_1^\top\right)\mathbf{G}\left(\mathbf{I}+\eta_t\mathbf{A}_1\right)\right]\right) \\
&\le \mathrm{Tr}\left(\mathbf{G}^2\right) + 4\eta_1\lambda_1\mathrm{Tr}\left(\mathbf{G}^2\right) + 5\eta_1^2\overline{\mathcal{V}}\cdot\mathrm{Tr}\left(\mathbf{G}^2\right) \\
&\quad + 4\eta_1^3(\mathcal{M}+\lambda_1)\,\overline{\mathcal{V}}\cdot\mathrm{Tr}\left(\mathbf{G}^2\right) + \eta_1^4(\mathcal{M}+\lambda_1)^2\,\overline{\mathcal{V}}\cdot\mathrm{Tr}\left(\mathbf{G}^2\right) \\
&= \left(1 + 4\eta_1\lambda_1 + 5\eta_1^2\overline{\mathcal{V}} + 4\eta_1^3(\mathcal{M}+\lambda_1)\,\overline{\mathcal{V}} + \eta_1^4(\mathcal{M}+\lambda_1)^2\,\overline{\mathcal{V}}\right)\mathrm{Tr}\left(\mathbf{G}^2\right) \\
&\le \exp\left(4\eta_1\lambda_1 + 10\eta_1^2\overline{\mathcal{V}}\right)\mathrm{Tr}\left(\mathbf{G}^2\right)
\end{aligned}$$

where in the last line we used that $\eta_i \le \frac{1}{4\max\{\mathcal{M},\lambda_1\}}$ and that $1+x \le \exp(x)$

Using the value $\mathbf{G} = \mathbf{G}_{t-1} = \mathbf{W}_{t,t-1}^\top \mathbf{v}_1\mathbf{v}_1^\top \mathbf{W}_{t,t-1}$ and plugging the above into (11), we have

$$\begin{aligned}
\gamma_t &= \mathrm{Tr}\left(\mathbb{E}\left[\left(\mathbf{I}+\eta_1\mathbf{A}_1^\top\right)\mathbf{G}_{t-1}\left(\mathbf{I}+\eta_1\mathbf{A}_1\right)\left(\mathbf{I}+\eta_1\mathbf{A}_1^\top\right)\mathbf{G}_{t-1}\left(\mathbf{I}+\eta_1\mathbf{A}_1\right)\right]\right) \\
&\le \exp\left(4\eta_1\lambda_1 + 10\eta_1^2\overline{\mathcal{V}}\right)\mathbb{E}\left[\mathrm{Tr}\left({\mathbf{G}_{t-1}}^2\right)\right] = \exp\left(4\eta_1\lambda_1 + 10\eta_1^2\overline{\mathcal{V}}\right)\gamma_{t-1},
\end{aligned}$$

where we used the fact that $\gamma_{t-1} = \mathbb{E}\left[\mathrm{Tr}\left({\mathbf{G}_{t-1}}^2\right)\right]$. Since $\gamma_0 = 1$, induction proves the lemma. ■

We now have everything to prove Theorem 7.

Proof [Proof of Theorem 7] As discussed in Section 3 the main idea of this proof to use that Algorithm 1 is essentially one step of power method for the matrix $\mathbf{B}_n$ and use Lemma 6 to bound the error. To this end, we lower and upper bound $\mathbf{v}_1^\top \mathbf{B}_n\mathbf{B}_n^\top \mathbf{v}_1$ and $\mathrm{Tr}\left(\mathbf{V}_\perp^\top \mathbf{B}_n\mathbf{B}_n^\top \mathbf{V}_\perp\right)$, respectively.

First, using Chebyshev's inequality, we have:

$$\mathbb{P}\left[\left|\mathbf{v}_1^\top \mathbf{B}_n\mathbf{B}_n^\top \mathbf{v}_1 - \mathbb{E}\left[\mathbf{v}_1^\top \mathbf{B}_n\mathbf{B}_n^\top \mathbf{v}_1\right]\right| > \frac{1}{\sqrt{\delta}}\sqrt{\mathrm{Var}\left[\mathbf{v}_1^\top \mathbf{B}_n\mathbf{B}_n^\top \mathbf{v}_1\right]}\right] < \delta.$$

So with probability greater than $1-\delta$, the following holds:

$$\begin{aligned}
&\mathbf{v}_1^\top \mathbf{B}_n \mathbf{B}_n^\top \mathbf{v}_1 > \mathbb{E}\left[\mathbf{v}_1^\top \mathbf{B}_n \mathbf{B}_n^\top \mathbf{v}_1\right] - \frac{1}{\sqrt{\delta}}\sqrt{\mathrm{Var}\left[\mathbf{v}_1^\top \mathbf{B}_n \mathbf{B}_n^\top \mathbf{v}_1\right]} \\
&= \mathbb{E}\left[\mathbf{v}_1^\top \mathbf{B}_n \mathbf{B}_n^\top \mathbf{v}_1\right] - \frac{1}{\sqrt{\delta}}\sqrt{\mathbb{E}\left[\left(\mathbf{v}_1^\top \mathbf{B}_n \mathbf{B}_n^\top \mathbf{v}_1\right)^2\right] - \mathbb{E}\left[\mathbf{v}_1^\top \mathbf{B}_n \mathbf{B}_n^\top \mathbf{v}_1\right]^2} \\
&\overset{\zeta_1}{\geq} \exp\left(2\lambda_1 \sum_{i=1}^n \eta_i - 4\lambda_1^2 \sum_{i=1}^n \eta_i^2\right) \times \left(1 - \frac{1}{\sqrt{\delta}}\sqrt{\exp\left(18\sum_{i=1}^n \eta_i^2 \overline{\mathcal{V}}\right) - 1}\right)
\end{aligned} \tag{16}$$

where ζ_1 follows from Lemma 11 and 12.

Furthermore, using Lemma 10 and Markov's inequality, we have with probability at least $1-\delta$,

$$\mathrm{Tr}\left(\mathbf{V}_\perp^\top \mathbf{B}_t \mathbf{B}_t^\top \mathbf{V}_\perp\right) \leq \frac{\exp\left(\sum_{i\in[n]} 2\eta_i\lambda_2 + \eta_i^2\overline{\mathcal{V}}\right)}{\delta} \cdot \left(d + \mathcal{V}\sum_{i=1}^n \eta_i^2 \exp\left(\sum_{j\in[i]} 2\eta_j(\lambda_1-\lambda_2)\right)\right). \tag{17}$$

Consequently with probability at least $1-2\delta$ both (16) and (17) hold and therefore the result follows by Lemma 6 and choosing a δ that is smaller by a constant. ■

6. Conclusion and Future Work

In this paper we presented finite sample complexity and asymptotic convergence rates for the classic Oja's algorithm for top-1 component streaming PCA that match well known matrix concentration and perturbation results for computing the top eigenvector. In fact, asymptotically our bound improves upon standard matrix Bernstein bounds by a factor of $\mathcal{O}(\log d)$. Our results are tighter than existing streaming PCA results by a factor of either $\mathcal{O}(d)$ or $\mathcal{O}(1/\mathrm{gap})$.

Our analysis relied on a novel view of the algorithm and is technically fairly simple. We hope that our analysis opens a way to make progress on the many variants of PCA that occur in both theory and practice. In particular, we believe the following directions should be of wide interest:

- **Multiple components**: Currently, our result holds only for estimating the top eigenvector of $\mathbf{\Sigma}$. Extension of our technique to compute top-k eigenvectors is an important future direction.
- **Rayleigh quotient**: Another standard metric to measure optimality of $\mathbf{w_n}$ is Rayleigh quotient: $\mathbf{w_n}^\top \mathbf{\Sigma} \mathbf{w_n}$. Converting our bounds on $\sin^2(\mathbf{w_n}, \mathbf{v}_1)$ to Rayleigh quotient loses a multiplicative factor of $\mathcal{O}(1/\mathrm{gap})$ compared to the optimal rate. A direct analysis that does not lose this factor is an interesting open problem. Results on Rayleigh quotient may also help in obtaining sample complexity guarantees that are independent of eigenvalue gap.
- **High Probability**: In this work, we focused on obtaining tight bounds on the error. However, the dependence of our results on success probability is quite suboptimal. One way to fix this is to run many copies of the algorithm, each with say $3/4$ success probability and then output

the geometric median of the solutions, which can be done in nearly linear time (Cohen et al., 2010). However, we conjecture that a tighter analysis using our techniques might directly lead to improved dependency on success probability and possibly help solve some of the other problems we mention above.

References

Akshay Balsubramani, Sanjoy Dasgupta, and Yoav Freund. The fast convergence of incremental pca. In *Advances in Neural Information Processing Systems*, pages 3174–3182, 2013.

Christos Boutsidis, Dan Garber, Zohar Karnin, and Edo Liberty. Online principal components analysis. In *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 887–901. SIAM, 2015.

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*, pages 205–214. ACM, 2009.

Michael Cohen, Yin Tat Lee, Gary Miller, Jakub Pachocki, and Aaron Sidford. Geometric median in nearly linear time. *To Appear in 48th Annual Symposium on the Theory of Computing (STOC) 2016*, 2010.

Michael B Cohen, Jelani Nelson, and David P Woodruff. Optimal approximate matrix product in terms of stable rank. *arXiv preprint arXiv:1507.02268*, 2015.

Dan Garber and Elad Hazan. Fast and simple pca via convex optimization. *arXiv preprint arXiv:1509.05647*, 2015.

Dan Garber, Elad Hazan, and Tengyu Ma. Online learning of eigenvectors. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, pages 560–568, 2015.

Mina Ghashami, Edo Liberty, Jeff M Phillips, and David P Woodruff. Frequent directions: Simple and deterministic matrix sketching. *arXiv preprint arXiv:1501.01711*, 2015.

Gene H Golub and Charles F Van Loan. *Matrix computations*, volume 3. JHU Press, 2012.

Peter M Hall, A David Marshall, and Ralph R Martin. Incremental eigenanalysis for classification. In *BMVC*, volume 98, pages 286–295. Citeseer, 1998.

Moritz Hardt and Eric Price. The noisy power method: A meta algorithm with applications. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 2861–2869, 2014.

Chi Jin, Sham M Kakade, Cameron Musco, Praneeth Netrapalli, and Aaron Sidford. Robust shift-and-invert preconditioning: Faster and more sample efficient algorithms for eigenvector computation. *arXiv preprint arXiv:1510.08896*, 2015.

Iain M Johnstone. On the distribution of the largest eigenvalue in principal components analysis. *Annals of statistics*, pages 295–327, 2001.

Ian Jolliffe. *Principal component analysis*. Wiley Online Library, 2002.

TP Krasulina. Method of stochastic approximation in the determination of the largest eigenvalue of the mathematical expectation of random matrices. *Automatation and Remote Control*, pages 50–56, 1970.

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

Ioannis Mitliagkas, Constantine Caramanis, and Prateek Jain. Memory limited, streaming pca. In *Advances in Neural Information Processing Systems*, pages 2886–2894, 2013.

John Nelson and Huy L Nguyên. Osnap: Faster numerical linear algebra algorithms via sparser subspace embeddings. In *Foundations of Computer Science (FOCS), 2013 IEEE 54th Annual Symposium on*, pages 117–126. IEEE, 2013.

Erkki Oja. Simplified neuron model as a principal component analyzer. *Journal of mathematical biology*, 15(3):267–273, 1982.

David A Ross, Jongwoo Lim, Ruei-Sung Lin, and Ming-Hsuan Yang. Incremental learning for robust visual tracking. *International Journal of Computer Vision*, 77(1-3):125–141, 2008.

Christopher De Sa, Christopher Re, and Kunle Olukotun. Global convergence of stochastic gradient descent for some non-convex matrix problems. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, pages 2332–2341, 2015.

Ohad Shamir. A stochastic PCA and SVD algorithm with an exponential convergence rate. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, pages 144–152, 2015a.

Ohad Shamir. Convergence of stochastic gradient descent for pca. *arXiv preprint arXiv:1509.09002*, 2015b.

Joel A Tropp. User-friendly tail bounds for sums of random matrices. *Foundations of Computational Mathematics*, 12(4):389–434, 2012.

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

Manfred K. Warmuth and Dima Kuzmin. Randomized PCA algorithms with regret bounds that are logarithmic in the dimension. In *Advances in Neural Information Processing Systems 19, Proceedings of the Twentieth Annual Conference on Neural Information Processing Systems, Vancouver, British Columbia, Canada, December 4-7, 2006*, pages 1481–1488, 2006.

Per-Åke Wedin. Perturbation bounds in connection with singular value decomposition. *BIT Numerical Mathematics*, 12(1):99–111, 1972.

Juyang Weng, Yilu Zhang, and Wey-Shiuan Hwang. Candid covariance-free incremental principal component analysis. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 25(8): 1034–1040, 2003.