Stochastic Gauss-Newton Algorithms for Online PCA

Siyun Zhou* Xin Liu† Liwei Xu‡

Abstract

In this paper, we propose a stochastic Gauss-Newton (SGN) algorithm to study the online principal component analysis (OPCA) problem, which is formulated by using the symmetric low-rank product (SLRP) model for dominant eigenspace calculation. Compared with existing OPCA solvers, SGN is of improved robustness with respect to the varying input data and algorithm parameters. In addition, turning to an evaluation of data stream based on approximated objective functions, we develop a new adaptive stepsize strategy for SGN (AdaSGN) which requires no priori knowledge of the input data, and numerically illustrate its comparable performance with SGN adopting the manaully-tuned diminishing stepsize. Without assuming the eigengap to be positive, we also establish the global and optimal convergence rate of SGN with the specified stepsize using the diffusion approximation theory.

1 Introduction

Principal component analysis (PCA) [30] is an orthogonal linear transformation technique primarily used for feature extraction and dimension reduction (see [17, 18]), which finds applications in diverse fields (e.g., [31, 42, 45]). Given a random vector $a \in \mathbb{R}^n$ with mean $\mathbb{E}[a] = 0 \in \mathbb{R}^n$ and covariance $\mathbb{E}[aa^\top] = \Sigma \in \mathbb{R}^{n\times n}$, let $\lambda_i \ \ (i = 1, 2, \cdots, n)$ be the i-th largest eigenvalue of Σ and $u_i \ \ (i = 1, 2, \cdots, n)$ be the corresponding unit eigenvector. Suppose the target dimension is p $(p \ll n)$, the PCA aims to recover a p-dimensional subspace spanned by the top p eigenvectors of Σ, i.e., $u_1, u_2, \cdots, u_p$, and these p eigenvectors are called principal components (PCs). In a practical context, the sample average approximation (SAA) picks m samples to construct the empirical covariance $\Sigma_m = \sum_{i=1}^m a^{(i)}a^{(i)\top}/m$ as an estimate of Σ. Denote the sample matrix by $A_m = [a^{(1)}, a^{(2)}, \cdots, a^{(m)}] \in \mathbb{R}^{n\times m}$. Then, the key of traditional PCA lies in the top-p eigenvalue decomposition (EVD) of Σ_m or equivalently the top-p singular value decomposition (SVD) on A_m. For the theoretical analysis, we make the following statistical assumption on A_m:

[A1] (i.i.d.) $a^{(1)}, a^{(2)}, \cdots, a^{(m)}$ *are independently and identically distributed realizations of the given random vector* $a \in \mathbb{R}^n$.

However, in many real-world scenarios, the input data does not arrive simultaneously, instead, it comes in a stream. The dynamic nature of the data demands real-time updates of the estimated PCs. This inspires PCA in an online setting, which proceeds under the following additional assumption:

[A2] (batchsize) *the samples are received sequentially, and no more than* h $(h \ll m)$ *passes over the data flow in within each round of update.*

The estimated PCs need to be updated before the new data enters. This variant of PCA is known as OPCA (OPCA), or streaming PCA (see [5]).

*School of Mathematical Sciences, University of Electronic Science and Technology of China, China, (zhousiyun@std.uestc.edu.cn).

†State Key Laboratory of Scientific and Engineering Computing, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, and University of Chinese Academy of Sciences, China, (liuxin@lsec.cc.ac.cn). The research was supported in part by the National Natural Science Foundation of China (No. 12125108, 11971466, 11991021, 12021001 and 11688101), Key Research Program of Frontier Sciences, Chinese Academy of Sciences (No. ZDBS-LY-7022), and the Youth Innovation Promotion Association, Chinese Academy of Sciences.

‡School of Mathematical Sciences, University of Electronic Science and Technology of China, China, (xul@uestc.edu.cn). The research was supported in part by the National Natural Science Foundation of China (No. 12071060).

There have been many works in solving the OPCA problem, and most of them are developed from the well-known Oja's iteration [29]. The direct extension of Oja's iteration, originally designed for $h = 1,\ \ p = 1$ case, is updated by

$$X^{(k+1)} = \mathbf{orth}\left\{ X^{(k)} + \frac{\alpha^{(k)}}{|\mathcal{I}^{(k)}|} \sum_{i \in \mathcal{I}^{(k)}} a^{(i)} a^{(i)\top} X^{(k)} \right\}, \tag{1.1}$$

where $\alpha^{(k)} > 0$ is the stepsize, and $\mathcal{I}^{(k)}$ is the index set of the k-th data block with the cardinality being smaller than h. One active line of related research is adapting Oja's iteration to meet specific requirements, e.g., embedding iterative soft thresholding for sparse PCs [48], downsampling for time-dependent data [6], studying modified variant for missing data [4], and to name a few.

The research on asymptotic or non-asymptotic analysis of convergence properties for (1.1) has been attracting increasing attention in recent years [1, 2, 12, 14, 25], and much more progress [3, 7, 9, 15, 23, 24, 36, 55] has been made for the simpler $p = 1$ case. Among these works, a three-phase analysis of Oja's iteration based on the diffusion approximation is proposed in [24]. The analysis not only provides theoretical guarantees, but also offers a more intuitive streamline along which we could attain a better understanding on the global and local behaviours of Oja's iteration. However, their results are restricted to the circumstance where $p = 1$ and $\lambda_1 > \lambda_2$. The forementioned theoretical studies indicate that in each step, Oja's update admits a constant variance upper bound resulting from the random samples, which leads to an optimal sublinear rate of $\mathcal{O}(1/k)$. To speed up the convergence, VR-PCA [35, 37] applies the variance reduction (VR) technique [16] to Oja's iteration, and obtains a faster linear rate which is however limited to the full-sampled case. Tang [43] proved the local linear rate of the less-studied Krasulina's method [20], whose update formula is closely related to Oja's iteration, but the result is valid only for data of low-rankness.

As being applied to practical computations, Oja-type algorithms require much more cares during the determination of stepsize. The Robbins-Monro conditions [32] suggest that, the stepsizes of Oja's method have to satisfy $\sum_k \alpha^{(k)} = \infty$ and $\sum_k \left(\alpha^{(k)}\right)^2 < \infty$ in order to guarantee its convergence to the optimum. Due to this reason, the commonly-used stepsize takes the form $\alpha^{(k)} = \gamma/(k+1)$. In this diminishing regime, Oja's method behaves with high sensitivity to γ, and the optimal γ has to be scaled with some quantities, which needs to be known in advance and are usually inaccessible in practical implementations, e.g., the eigengap $(\lambda_p - \lambda_{p+1})$. Some work [28] attempts to design a new stepsize scheme, and however still involves hyperparameter tuning. To be completely parameter-free, AdaOja [13] applies the simplified AdaGrad algorithm [50] designed for SGD to Oja's iteration in a column-wise fashion, taking the update in the form

$$\begin{aligned} b_i^{(k)} &= \sqrt{\left(b_i^{(k-1)}\right)^2 + \left\| G^{(k)}_{(:,\ i)} \right\|_2^2}, \quad i = 1, 2, \cdots, p, \\ X^{(k+1)} &= \mathbf{orth}\left\{ X^{(k)} + G^{(k)} \mathbf{Diag}^{-1}(b_1^{(k)}, \cdots, b_p^{(k)}) \right\}, \end{aligned} \tag{1.2}$$

where $G^{(k)}$ is Oja's direction in the k-th step, $b_i^{(k)}$ and $G^{(k)}_{(:,\ i)}$ denote the i-th element of $b^{(k)}$ and i-th column of $G^{(k)}$, respectively. The AdaOja empirically demonstrates the ability of self-adjustment on the stepsize compared with the state-of-art OPCA algorithms. But the numerical tests in [13] only address the explained variance, which is defined by the percentage of variance recovered from the original dataset, without presenting any results of the error measurement that is based on the canonical subspace angle.

Algorithms for offline PCA have been extensively studied in the past decades. The conventional eigensolvers are built on Krylov subspace (e.g., [34, 38, 39, 40]), and approximate the eigenvectors in an incremental fashion without fulfilling the demands for parallel scalability and high efficiency at moderate accuracy. These limitations could be fixed by developing block algorithms, and such algorithms are mostly derived from the following Rayleigh-Ritz trace minimization model over the Stiefel manifold

$$\min_{X \in \mathbb{R}^{n \times p}} \quad -\mathbf{tr}\left(X^\top \Sigma_m X\right), \qquad \text{subject to} \quad X^\top X = I_p. \tag{1.3}$$

The eigensolvers based on solving (1.3) include SSI (see [11, 33]), LOBPCG [19], LMSVD [26] and ARRABIT [52]. Some other works focus on constructing equivalent models to (1.3) in generating top/bottom eigenspace, e.g., EigPen [51] for the trace-penalty minimization model and SLRPGN [27] for the symmetric low-rank product (SLRP) model. We refer to [10, 49, 54] for recent progress in solving general problems over the Stiefel manifold.

In this paper, taking the advantage of the nonlinear least squares form, we focus on the SLRP model which formulates the PCA problem into

$$\min_{X\in\mathbb{R}^{n\times p}} \quad f(X)=\frac{1}{2}\left\|XX^\top-\Sigma_m\right\|_{\mathrm{F}}^2, \tag{1.4}$$

with the corresponding expectational counterpart

$$\min_{X\in\mathbb{R}^{n\times p}} \quad f_{\mathrm{E}}(X)=\frac{1}{2}\left\|XX^\top-\Sigma\right\|_{\mathrm{F}}^2. \tag{1.5}$$

To solve the SLRP model, Liu et al. [27] proposed a Gauss-Newton (GN) method of minimum weighted-norm named as SLRPGN, which enjoys a simple explicit update rule and a local linear rate with the constant stepsize $\alpha^{(k)}=1\ \ (k=0,1,\cdots)$.

1.1 Contributions

For PCA in the online fashion, we propose a new stochastic optimization algorithm named as the stochastic Gauss-Newton (SGN) method. Numerical experiments on both the simulated and the real data demonstrate that SGN is stably effective with different choices of initial points when the constant stepsizes are used. Furthermore, SGN is empirically robust with respect to (w.r.t.) the change of stepsize parameters as well as a variety of input data when the frequently-used diminishing stepsizes are adopted. In addition, we develop an adaptive stepsize version of SGN named as AdaSGN, which is based on the consistency of successive batches of online data reflected by the approximate objective functions of SLRP. The AdaSGN shows much better numerical performance than the state-of-the-art adaptive OPCA algorithm AdaOja (1.2), and is comparable to SGN using the manually-tuned diminishing stepsizes.

Regarding to the theoretical aspect, based on the diffusion approximation, we establish the weak convergence (or convergence in distribution) of the SGN/properly-rescaled-SGN sequence in the infinitesimal stepsize regime to their corresponding continuous time process limits, which can be modeled by the solution of certain ordinary/stochastic differential equation (ODE/SDE). Using these differential equation approximations, we obtain the global convergence rate of SGN with both the constant and diminishing stepsizes for any $p\geq 1$ under the assumption $\lambda_p>\lambda_n$, which is weaker than the commonly-used positive eigengap assumption $\lambda_p>\lambda_{p+1}$. The error bound for the diminishing stepisze case is optimal in the sense that it exactly matches the minimax lower bound given in [47], and this result is new among the existing related works. However, due to an extra log factor, the bound for the constant stepsize case is only proved to be nearly optimal.

1.2 Notations

For given matrix $A\in\mathbb{R}^{n_1\times n_2}$, $\sigma_{\max}(A)$ and $\sigma_{\min}(A)$ stand for the largest and smallest singular values of A, respectively. The (n_1n_2)-dimensional vector $\mathbf{vec}(A)$ denotes the vectorization of A which piles columns of A on top of one another. The submatrices $A_{(i_1:i_2,j_1:j_2)}$ and $A_{(:,j)}$ are composed of i_1-th to i_2-th rows and j_1-th to j_2-th columns of A and the j-th column of A, respectively. Given an event $\mathcal{A}$, we use $\mathbb{1}_{\{\mathcal{A}\}}$ to denote the indicator function with the value of one if $\mathcal{A}$ occurs, and otherwise the value of zero. We denote the n-dimensional indentity matrix by I_n and the unit vector with the i-th coordinate being one and all others being zero by e_i. We denote $0_{n_1\times n_2}$ as the $n_1\times n_2$ zero matrix. Given two functions $g(\alpha)$ and $h(\alpha)$, the relationships $g\asymp h$ and $g\lesssim h$ imply $\limsup\limits_{\alpha\to 0} g(\alpha)/h(\alpha)=1$ and $\limsup\limits_{\alpha\to 0} g(\alpha)/h(\alpha)\leq 1$, respectively.

1.3 Organization

The rest of this paper is organized as follows. We introduce the derivation of our proposed GN algorithms (SGN and AdaSGN) in Section 2 and prove the convergence properties for SGN in the infinitesimal stepsize regime based on Itô diffusion theory in Section 3. The numerical results in Section 4 demonstrate the feasibility and advantages of the proposed GN algorithms. Finally, a conclusion is made in Section 5.

2 Algorithm

In this section, we first introduce the SLRP model in the online setting, which has been proved to be equivalent to PCA in the sense of eigen-space computation. We then present the derivation of SGN algorithm for solving the OPCA problem, followed by its adaptive-stepsize version.

2.1 Online SLRP

In view of the online setting, we first define a batch-h approximation of Σ_m in the k-th step:

$$\Sigma_h^{(k)} = \frac{1}{h} A_h^{(k+1)} A_h^{(k+1)\top} = \frac{1}{h} \sum_{i=kh+1}^{kh+h} a^{(i)} a^{(i)\top}, \tag{2.1}$$

where the dynamic data block is given by

$$A_h^{(k+1)} = [a^{(kh+1)} \quad a^{(kh+2)} \quad \cdots \quad a^{(kh+h)}]. \tag{2.2}$$

Then (2.1) yields the corresponding batch-h approximation of (1.4) in the k-th step:

$$\min_{X \in \mathbb{R}^{n\times p}} \quad \hat{f}^{(k)}(X) = \frac{1}{2} \left\| XX^\top - \Sigma_h^{(k)} \right\|_{\mathrm{F}}^2 . \tag{2.3}$$

The gradient of $\hat{f}^{(k)}(X)$ is given by

$$\nabla \hat{f}^{(k)}(X) = 2 \left(X(X^\top X) - \Sigma_h^{(k)} X \right).$$

2.2 SGN for OPCA

For simplicity, let $R^{(k)}(X) = (XX^\top - \Sigma_h^{(k)})$ be the residual in the k-th step. The GN direction for the nonlinear least squares problem (2.3), denoted as $S^{(k)}(X)$, could be obtained by solving the normal equation:

$$J(X)^\top J(X)(S) = -J(X)^\top R^{(k)}(X), \tag{2.4}$$

where $J(X) : \mathbb{R}^{n\times p} \to \mathbb{R}^{n\times n}$ is the Jacobian operator of $R^{(k)}(X)$ at X, and $J(X)^\top : \mathbb{R}^{n\times n} \to \mathbb{R}^{n\times p}$ is the adjoint operator of $J(X)$ at X. The rank deficiency of $J(X)$ makes the linear system formed by (2.4) admit an infinite number of solutions, but fortunately, the solutions could be expressed explicitly (see Proposition 3.1 of [27]). In the same way as [27], we choose the one of minimum weighted-norm from the solution set of (2.4) as our SGN direction:

$$S^{(k)}(X) = \Sigma_h^{(k)} X(X^\top X)^{-1} - \frac{1}{2} X - \frac{1}{2} X(X^\top X)^{-1} X^\top \Sigma_h^{(k)} X(X^\top X)^{-1}, \tag{2.5}$$

which could be formulated into a three-step procedure: for $X = X^{(k)}, \;\; A = A_h^{(k+1)}$,

$$P = X(X^\top X)^{-1}, \;\; Q = A^\top P \Big/ \sqrt{h}\,, \;\; S^{(k)}(X) = AQ \Big/ \sqrt{h} - X(I_p + Q^\top Q)/2\,. \tag{2.6}$$

The SGN approach for OPCA is summarized in Algorithm 1.

Algorithm 1 A SGN method for top-p OPCA

```
Require: Samples $a^{(1)}, a^{(2)}, \cdots$; target dimension $p$; stepsizes $\alpha^{(k)}$; batchsize $h$.
Ensure: Choose rank-$p$ $\hat{X}^{(0)} \in \mathbb{R}^{n\times p}$ randomly and set $X^{(0)} = \mathbf{orth}\{\hat{X}^{(0)}\}$.
for $k = 0, 1, 2, \cdots$ do
    Update the sample batch $A_h^{(k+1)}$ by (2.2).
    Compute $S^{(k)}(X^{(k)})$ by (2.6).
    Iterate $X^{(k+1)} = X^{(k)} + \alpha^{(k)} S^{(k)}(X^{(k)})$.
end for
Output: $\mathbf{orth}\{X^{(k)}\}$.
```

2.3 AdaSGN for OPCA

As being seen from (1.2), the AdaOja uses the stochastic gradients obtained from a single batch to adjust stepsizes, which is a widely used strategy in online algorithms. Taking into account the limitation of incomplete knowledge from a single batch, our adaptive stepsize scheme focuses on the consistency between successive sample batches, which is characterized by the approximated objective function $\hat{f}^{(k)}(X)$ given by (2.3). Since $X^{(k)}$ is derived from the subproblem of minimizing $\hat{f}^{(k-1)}(X)$, it is clear that

$$\hat{f}^{(k-1)}(X^{(k)}) \leq \hat{f}^{(k-1)}(X^{(k-1)}), \tag{2.7}$$

always holds. If the new sample batch $A_h^{(k+1)}$ maintains consistency with history samples in the sense of (2.7), i.e., satisfying $\hat{f}^{(k)}(X^{(k)}) \leq \hat{f}^{(k)}(X^{(k-1)})$, we make the stepsize larger or unchanged; otherwise we decrease the stepsize.

To describe the extent of consistency, we define a parameter-free indicator $r^{(k)}$ as

$$r^{(k)} = \begin{cases} \hat{f}^{(k)}(X^{(k-1)}) \Big/ \hat{f}^{(k)}(X^{(k)}) & \hat{f}^{(k)}(X^{(k)}) > \hat{f}^{(k)}(X^{(k-1)}) \\ 0 & \text{otherwise} \end{cases} \tag{2.8}$$

for $k > 0$ and $r^{(0)} = 1$. Then, we set the stepsize by

$$\alpha^{(k)} = \begin{cases} r^{(k)} \Big/ \sum_{i=0}^{k} r^{(i)} & \hat{f}^{(k)}(X^{(k)}) > \hat{f}^{(k)}(X^{(k-1)}) \\ 1 \Big/ \sum_{i=0}^{k} r^{(i)} & \text{otherwise} \end{cases} \in [0, 1). \tag{2.9}$$

For the poorly-consistent sample case where $\hat{f}^{(k)}(X^{(k)}) > \hat{f}^{(k)}(X^{(k-1)})$, smaller $r^{(k)}$ implies less consistent sample, thus we provide a smaller $\alpha^{(k)}$ according to $a_1/(c + a_1) > a_2/(c + a_2)$ for $a_1 > a_2 > 0$, $c > 0$. For the well-consistent sample case where $\hat{f}^{(k)}(X^{(k)}) \leq \hat{f}^{(k)}(X^{(k-1)})$, our scheme makes AdaSGN yield no monotonic decrease in stepsize but an overall decreasing trend. This strategy has advantages in such circumstance as many extremely bad samples are received in the early stage.

The SGN using the stepsize scheme stated above is generalized in Algorithm 2.

3 Theory of algorithm

In this section, since the sequence $\{X^{(k)}\}$ generated by SGN algorithm constitutes a discrete-time Markov process, we use ODE/SDEs to characterize the SGN sequence with infinitesimal stepsize in the weak sense, and prove its global convergence rate based on these differential equations. We begin with some basic concepts and analytical techniques in Subsection 3.1. In Subsection 3.2, we present the convergence results of SGN in the constant stepsize regime, along with detailed proofs. The last subsection is devoted to investigating the diminishing stepsize case.

Algorithm 2 AdaSGN for OPCA

```
Require: Samples $a^{(1)}, a^{(2)}, \cdots$; target dimension $p$; batchsize $h$.
Ensure: Choose rank-$p$ $\hat{X}^{(0)} \in \mathbb{R}^{n\times p}$ randomly and set $X^{(0)} = \mathbf{orth}\{\hat{X}^{(0)}\}$, $r^{(0)} = 1$.
for $k = 0, 1, 2, \cdots$ do
    Update $A_h^{(k+1)}$ according to (2.2).
    if $k > 0$ then
        Compute $\hat{f}^{(k)}(X^{(k-1)})$, $\hat{f}^{(k)}(X^{(k)})$ by (2.3), and $r^{(k)}$ by (2.8).
    end if
    Update the stepsize $\alpha^{(k)}$ by (2.9).
    Perform Step 3 and Step 4 in Algorithm 1.
end for
Output: $\mathbf{orth}\{X^{(k)}\}$.
```

3.1 Preliminaries

We firstly revisit some basic definitions.

Definition 3.1 (Sub-Gaussian random vector)**.** *A random vector $v \in \mathbb{R}^n$ is said to follow a sub-Gaussian distribution if there exists a constant $\sigma^2 > 0$ (called variance proxy) such that for any $s \in \mathbb{R}$ and unit vector $c \in \mathbb{R}^n$, the following inequality holds*

$$\mathbb{E}\exp\left\{s(c^\top v - c^\top \mathbb{E}v)\right\} \le \exp\left\{\sigma^2 s^2/2\right\}.$$

In order to carry out the analysis, we assume that

[A3] (sub-Gaussian distribution) *$a \in \mathbb{R}^n$ is a sub-Gaussian random vector.*

It is known that a sub-Gaussian vector has finite moments of any order. This property will be used to prove the expectational boundedness and full-rankness of SGN iterates in Lemma B.1 and Lemma B.2.

We assume for distinguishing the top PCs from the remaining ones that

[A4] $\lambda_p > \lambda_n$.

Definition 3.2. *Under the assumption [A4], we define p' as the smallest index satisfying $\lambda_{p'+1} < \lambda_p$. Namely,*

$$p' = \min\{i \mid p \le i \le n-1, \;\; \lambda_{i+1} < \lambda_p\}.$$

We denote $\nu = \lambda_p - \lambda_{p'+1}$.

Note that the assumption [A4] covers a frequently-used requirement $\lambda_p > \lambda_{p+1}$ for the analysis of OPCA algorithms when p' coincides with p. Given $X \in \mathbb{R}^{n\times p}$, let $\bar{X} = X_{(1:p',\;:)} \in \mathbb{R}^{p'\times p}$ and $\underline{X} = X_{((p'+1):n,\;:)} \in \mathbb{R}^{(n-p')\times p}$.

Definition 3.3 (Canonical angles between subspaces)**.** *Let $X_1 \in \mathbb{R}^{n\times p_1}$ and $X_2 \in \mathbb{R}^{n\times p_2}$ ($p_1 \le p_2 \le n$) be two matrices with orthonormal columns. The canonical or principal angles between two subspaces spanned by the columns of X_1 and X_2 are*

$$\theta_i = \arccos \sigma_i \in [0, \pi/2], \quad i = 1, 2, \cdots, p_1,$$

where σ_i are singular values of $X_1^\top X_2$. Denote $\Theta(X_1, X_2) = \mathbf{Diag}(\theta_1, \cdots, \theta_{p_1})$ and $\sin\Theta(X_1, X_2) = \mathbf{Diag}(\sin\theta_1, \cdots, \sin\theta_{p_1})$.

The error of the k-th iterate $X^{(k)}$ is measured by

$$\|\sin\Theta(X^{(k)}, U_{p'})\|_{\mathrm{F}}^2 = p - \|U_{p'}^\top \mathbf{orth}\{X^{(k)}\}\|_{\mathrm{F}}^2, \tag{3.1}$$

where $U_{p'} = [u_1, u_2, \cdots, u_{p'}]$. Without loss of generality, the covariance is assumed to be of a diagonal form $\Sigma = \mathbf{Diag}(\lambda_1, \lambda_2, \cdots, \lambda_n)$ hereinafter, and the corresponding error measurement becomes

$$\|\sin\Theta(X^{(k)}, U_{p'})\|_{\mathrm{F}}^2 = p - \mathbf{tr}\left(\bar{X}^{(k)\top}\bar{X}^{(k)}(X^{(k)\top}X^{(k)})^{-1}\right). \tag{3.2}$$

As mentioned earlier, the SGN sequence $\{X^{(k)}\}$ could be viewed as a discrete-time Markov process. The basis of the theory of SGN is the well-known diffusion approximation technique, of which the fundamental idea is to characterize the given analytically intractable stochastic process by some certain diffusion processes (modeled by the solution to some SDEs) with sufficiently good and useful properties.

Considering the case of $p = 1$ and $\alpha^{(k)} = \alpha$ $(k = 0, 1, \cdots)$, where the SGN update reads

$$X^{(k+1)} = X^{(k)} + \alpha S_{\mathrm{E}}(X^{(k)}) + \sqrt{\alpha}\left[\sqrt{\alpha}\left(S^{(k)}(X^{(k)}) - S_{\mathrm{E}}(X^{(k)})\right)\right], \tag{3.3}$$

with $S_{\mathrm{E}}(X^{(k)}) = \mathbb{E}[S^{(k)}(X^{(k)})]$. By taking $\alpha = \Delta t$ and comparing formula (3.3) with the Euler discretization

$$X^{(k+1)} = X^{(k)} + \Delta t b(X^{(k)}) + \sqrt{\Delta t}\sigma(X^{(k)})\xi^{(k)}, \quad \xi^{(k)} \sim \mathcal{N}(0, I_n),$$

of some SDE in the form

$$dX(t) = b(X(t))dt + \sigma(X(t))dB(t), \tag{3.4}$$

we could then naturally expect (3.4) with certain drift coefficient $b(X)$ and diffusion coefficient $\sigma(X)$ to be a distributional approximation of the discrete SGN sequence.

As a formalization of the above intuitive statement, Corollary 4.2 in Section 7.4 of [8] suggests that the main issues in the convergence analysis (of the constant stepsize case) consist of (i) constructing a continuous-time process

$$X_\alpha(t) = X^{(\lfloor t\alpha^{-1}\rfloor)},$$

from the discrete-time SGN sequence $\{X^{(k)}\}$ using the constant stepsize $\alpha^{(k)} = \alpha \in (0, 1]$ by piecewise constant interpolation; (ii) finding the limiting SDEs (3.4) of vectorized $X_\alpha(t)$ in distribution as $\alpha \to 0$ via calculating the infinitesimal mean b and variance $a = \sigma\sigma^\top$ defined by

$$b(X) = \frac{d}{dt}\mathbb{E}[\mathbf{vec}(X(t))] = \lim_{\alpha\to 0}\frac{1}{\alpha}\mathbb{E}[\mathbf{vec}(\Delta X_\alpha(t))|X_\alpha(t) = X], \tag{3.5}$$

$$a(X) = \frac{d}{dt}\mathrm{Var}[\mathbf{vec}(X(t))] = \lim_{\alpha\to 0}\frac{1}{\alpha}\mathbb{E}[\mathbf{vec}(\Delta X_\alpha(t))\mathbf{vec}(\Delta X_\alpha(t))^\top|X_\alpha(t) = X], \tag{3.6}$$

where $\mathbf{vec}(\Delta X_\alpha(t)) = \mathbf{vec}(X_\alpha(t+\alpha) - X_\alpha(t))$; (iii) using the derived SDE approximation to learn the properties of SGN iterates.

3.2 Main results on constant stepsizes

We first provide the main results of the constant stepsize case in Theorem 3.1 and Theorem 3.2, and delay their proofs until later in this subsection.

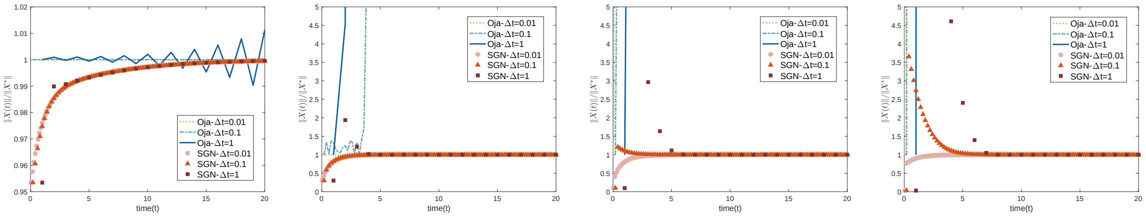
The following theorem states the global convergence of SGN with the constant stepsize.

Theorem 3.1. *Under the assumptions [A1]-[A4], the process $\{X^{(\lfloor t\alpha^{-1}\rfloor)}\}$ generated by Algorithm 1 with the stepsize $\alpha^{(k)} = \alpha$ $(k = 0, 1, \cdots)$ converges in distribution to the solution of*

$$\frac{dX}{dt} = \Sigma X(X^\top X)^{-1} - \frac{1}{2}X - \frac{1}{2}X(X^\top X)^{-1}X^\top\Sigma X(X^\top X)^{-1}, \tag{3.7}$$

as $\alpha \to 0$ with an initial value $X(0) = X^{(0)}$. Furthermore, the solution $X(t)$ to (3.7) satisfies

$$\lim_{t\to\infty}\nabla f_E(X(t)) = 0.$$



(a) $\lambda_1 = 1.1$ (b) $\lambda_1 = 11$ (c) $\lambda_1 = 101$ (d) $\lambda_1 = 1001$

Figure 1: Finite difference discretization of the limiting ODE for Oja's iteration (Theorem 3.1 in [24]) and the one for SGN (3.7) with various stepsize Δt $(n = 500,\ \ p = 1,\ \ \lambda_2 = \cdots = \lambda_{500} = 1)$.

Using the classic forward difference approximations, we provide a simple comparison between the discretization of the limiting ODE for Oja's iteration (Theorem 3.1 in [24]) and the one of (3.7) for SGN in Subsection 3.2, where the numerical instability of Oja's method could be apparently observed as the top eigenvalue λ_1 increases.

Then, we present the convergence rate of SGN in terms of the expectational error.

Theorem 3.2. *Suppose that the assumptions [A1]-[A4] hold. Given the total sample size $m > 0$, and the stepsize*

$$\alpha^{(k)} = \alpha = \frac{\lambda_p}{\nu} K^{-1}\log K, \qquad K = \left\lceil \frac{m}{h} \right\rceil, \qquad k = 0, 1, \cdots, K, \tag{3.8}$$

the expectational error of the final output $X^{(K)}$ of Algorithm 1 is given by

$$\begin{aligned}\mathbb{E}\|\sin\Theta(X^{(K)}, U_{p'})\|_{\mathrm{F}}^2 &\lesssim \frac{C}{h}\frac{\lambda_p \log K}{\nu K}\sum_{l=p'+1}^{n}\sum_{r=1}^{p}\frac{\lambda_l}{2(\lambda_r - \lambda_l)} \\ &\le \frac{C}{2h} p(n-p')\frac{\lambda_p\lambda_{p'+1}}{\nu^2}K^{-1}\log K,\end{aligned}\tag{3.9}$$

where C is some positive constant.

Remark 3.3 (Comparison with previous work)**.** *When taking $p = p' = 1$, our error bound (3.9) is coincident with the bound of [24] for Oja's iteration, where the stepsize is chosen to be $\alpha^{(k)} = \alpha = \nu^{-1}K^{-1}\log K$. This suggests that when λ_1 is large, our SGN allows for a larger stepsize to achieve a fixed order of accuracy.*

Remark 3.4 (Constant variance upper bound)**.** *As the update goes to the optimum, for $X = X^{(k)}$, $\Sigma_c = \Sigma_h^{(k)} - \Sigma$, the variance statistics [44] of one-step Oja's iteration and SGN in the Frobenius norm sense in the k-th update are*

$$v\left(X^{(k+1)}_{Oja}\right) = (\alpha^{(k)})^2\left\|X^\top\mathbb{E}[\Sigma_c^2]X\right\|_F \to (\alpha^{(k)})^2\left\|U_p^\top\mathbb{E}\Sigma_c^2 U_p\right\|_F, \tag{3.10}$$

$$\begin{aligned}v\left(X^{(k+1)}_{SGN}\right) &= (\alpha^{(k)})^2\left\|\mathbb{E}\left[P^\top\Sigma_c\left(I - 3PX^\top/4\right)\Sigma_c P\right]\right\|_F \\ &\le (\alpha^{(k)})^2\left\|P^\top\mathbb{E}[\Sigma_c^2]P\right\|_F \to (\alpha^{(k)})^2\left\|\bar\Lambda^{-\frac{1}{2}}U_p^\top\mathbb{E}[\Sigma_c^2]U_p\bar\Lambda^{-\frac{1}{2}}\right\|_F,\end{aligned}\tag{3.11}$$

where $\bar\Lambda = \mathbf{Diag}(\lambda_1, \cdots, \lambda_p)$, $U_p = [u_1, \cdots, u_p]$ and P is given by (2.6). It could be easily verified from (3.10) and (3.11) that both of SGN and Oja's iteration admit a constant variance upper bound unless the stepsize $\alpha^{(k)}$ vanishes, and therefore neither of them could achieve a linear rate. However, unlike the Oja's iteration, the bound of SGN involves an inverse top-eigenvalue matrix $\bar\Lambda$ as the iterate approaches the optimum, which we could take as an advantage in the adaptation to different datasets.

The followings are two technical lemmas that are necessary for the proofs of Theorem 3.1 and Theorem 3.2.

Lemma 3.5. *Under the assumptions [A1]-[A4], the process $X_\alpha(t) = X^{(\lfloor t\alpha^{-1}\rfloor)}$ generated by Algorithm 1 with the stepsize $\alpha^{(k)} = \alpha$ $(k = 0, 1, \cdots)$ converges in distribution to the solution of (3.7) as $\alpha \to 0$ with $X(0) = X^{(0)}$.*

Remark 3.6. *Since the diffusion coefficient of $X(t)$ is $\sigma(X) = \mathcal{O}(\alpha^{\frac{1}{2}})$, we obtain an ODE (3.7) for the global dynamics of SGN updates omitting the stochasticity. However, the drift term of (3.7) vanishes and its diffusion term dominates instead when the iterate gets close to some stationary point $\tilde{X} \in \mathbb{R}^{n\times p}$ of (1.5). For purposes of capturing local dynamics near $\tilde{X}$, we introduce a new Markov process via rescaling the original SGN updates by a factor of $\alpha^{-\frac{1}{2}}$ and get a new SDE in Lemma 3.7.*

It is known from the first order optimality condition of (1.5)

$$\Sigma X = XX^\top X \tag{3.12}$$

that a rank-p stationary point $\tilde{X}$ spans an invariant subspace of Σ, which could also be spanned by p eigenvectors of Σ. Based on this, we equip each $\tilde{X}$ with an eigen-index set $\mathcal{I}_p = \{i_1, \cdots, i_p\} \subset \{1, \cdots, n\}$, which indicates that $\tilde{X}$ has p singular values $\lambda_i^{\frac{1}{2}}$ $(i \in \mathcal{I}_p)$. Given that the multiplicity of λ_i $(i \in \mathcal{I}_p)$ may not be one, we further define an extended eigen-index set $\hat{\mathcal{I}}_p$ that contains all indexes sharing the same eigenvalue as each one in the original eigen-index set $\mathcal{I}_p$, i.e., satisfying (i) $\mathcal{I}_p \subseteq \hat{\mathcal{I}}_p$; (ii) $\lambda_i = \lambda_j$ for any $i \in \hat{\mathcal{I}}_p$ and some $j \in \mathcal{I}_p$; (iii) $\lambda_i \neq \lambda_j$ for any $i \in \hat{\mathcal{I}}_p$, $j \notin \hat{\mathcal{I}}_p$.

Then, we have the following result.

Lemma 3.7. *Suppose that the assumptions [A1]-[A4] hold and the initial point is around some rank-p stationary point of (1.5), which is associated with an eigen-index set $\mathcal{I}_p = \{i_1, \cdots, i_p\}$ and an orthogonal matrix $V \in \mathbb{R}^{p\times p}$. We define a new process*

$$Y_\alpha(t) = \alpha^{-\frac{1}{2}} X_\alpha(t) V = \alpha^{-\frac{1}{2}} X^{(\lfloor t\alpha^{-1}\rfloor)} V, \tag{3.13}$$

generated by Algorithm 1 with the stepsize $\alpha^{(k)} = \alpha$ $(k = 0, 1, \cdots)$. If $Y_\alpha(0)$ converges in distribution to some constant $Y^0 \in \mathbb{R}^{n\times p}$ as $\alpha \to 0$ and the index l is not in the extended eigen-index set $\hat{\mathcal{I}}_p$, then the (l, r)-th entry of $Y_\alpha(t)$ converges in distribution to the following SDE with $Y_{(l,r)}(0) = Y^0_{(l,r)}$

$$dY_{(l,r)} = \left(\Sigma Y \Lambda_p^{-1} - Y\right)_{(l,r)} dt + h^{-\frac{1}{2}} \lambda_l^{\frac{1}{2}} dB(t), \qquad 1 \le r \le p, \tag{3.14}$$

where $\Lambda_p = \mathbf{Diag}(\lambda_{i_1}, \lambda_{i_2}, \cdots, \lambda_{i_p}) \in \mathbb{R}^{p\times p}$ and $B(t)$ is the Brownian motion.

The proofs of the above two lemmas are provided in the appendix.

We are now ready to prove Theorem 3.1 and Theorem 3.2 as follows.

Proof of Theorem 3.1. We first decompose the right-side of (3.7) into two orthogonal components $P_1(X), P_2(X)$ as

$$\frac{dX}{dt} = P_1(X) + P_2(X), \quad P_1(X) = \Sigma P - XP^\top \Sigma P, \quad P_2(X) = \frac{1}{2} XP^\top \Sigma P - \frac{1}{2} X,$$

where $P = X(X^\top X)^{-1}$. Thus, the gradient of $f_{\mathrm{E}}(X)$ could be expressed by

$$\nabla f_{\mathrm{E}}(X) = 2(XX^\top X - \Sigma X) = -2(P_1 + 2P_2)(X^\top X).$$

We then obtain a new ODE

$$\begin{aligned}\frac{df_{\mathrm{E}}}{dt} = \mathbf{tr}\left((\nabla f_{\mathrm{E}}(X))^\top \frac{dX}{dt}\right) &= -2\mathbf{tr}\left[(X^\top X)(P_1 + 2P_2)^\top (P_1 + P_2)\right] \\ &= -2\left(\|P_1 X^\top\|_{\mathrm{F}}^2 + 2\|P_2 X^\top\|_{\mathrm{F}}^2\right) \le 0,\end{aligned}$$

which demonstrates the non-increasing property of the residual function $f_{\mathrm{E}}(X)$. When both $\|P_1 X^\top\|_{\mathrm{F}}^2 = 0$ and $\|P_2 X^\top\|_{\mathrm{F}}^2 = 0$ hold, we obtain (3.12), which indicates that the asymptotic solution of (3.7) is the stationary point of (1.5). Finally, applying Lemma 3.5 immediately completes the proof of this theorem. □

Remark 3.8. *If $p' = p$ and $\bar{X}(t)$ is invertible, we could further conclude that the solution $X(t)$ to (3.7) converges to the global optimum of (1.5), that is*

$$\lim_{t\to\infty} X(t) = \begin{bmatrix} \bar{\Lambda}^{\frac{1}{2}} & 0_{p\times(n-p)} \end{bmatrix}^\top V^\top, \quad \lim_{t\to\infty} \bar{X}(t) = \bar{\Lambda}^{\frac{1}{2}} V^\top,$$

where $V \in \mathbb{R}^{p\times p}$ is an orthogonal matrix. This indicates that for any $\varepsilon > 0$, there exists a constant $T > 0$ such that for all $t > T$

$$\left\| X(t) - \begin{bmatrix} \bar{\Lambda}^{\frac{1}{2}} & 0_{p\times(n-p)} \end{bmatrix}^\top V^\top \right\|_F < \varepsilon, \quad \|\bar{X}(t) - \bar{\Lambda}^{\frac{1}{2}} V^\top\|_F < \varepsilon. \tag{3.15}$$

Let $r_i = e_i^\top X(\bar{X}^\top \bar{X})^{-1} X^\top e_i$, $(p < i \leq n)$, we can obtain from (3.7) that

$$\begin{aligned} \frac{dr_i}{dt} &= 2e_i^\top \Sigma X(X^\top X)^{-1}(\bar{X}^\top \bar{X})^{-1} X^\top e_i - 2e_i^\top X \bar{X}^{-1} \bar{\Lambda} \bar{X} (X^\top X)^{-1} (\bar{X}^\top \bar{X})^{-1} X^\top e_i \\ &\leq \begin{cases} 2(\lambda_i - \lambda_p)\lambda_1^{-1} r_i, & p = 1, \\ 2(\lambda_i - \lambda_p)\lambda_1^{-1} r_i + \mathcal{O}(\varepsilon), & p > 1, \ t > T. \end{cases} \end{aligned}$$

The inequality above implies that the i-th row of X $(p < i \leq n)$ declines to zero with a speed which is at least exponential when $p = 1$, or when $t > T$.

Recall from (3.2) that the error is measured by $p - \mathbf{tr}\left(\bar{X}^\top \bar{X}^\top (X^\top X)^{-1}\right) \in [0, \ p]$ for $X = X^{(k)}$, we define a zero-error optimal set for the top-p OPCA problem as

$$\mathcal{X}^* = \left\{ X \in \mathbb{R}^{n\times p} \text{ of full rank satisfying } (3.12) \text{ and } \mathbb{E}\mathbf{tr}\left(\bar{X}^\top \bar{X}^\top (X^\top X)^{-1}\right) = p \right\},$$

which includes the global minimum of (1.5). As could be seen from Subsection 3.2, we divide the whole trajectory of SGN iterates into three phases according to the expectational error. Based on this, we could then prove the convergence rate of SGN.

$$\mathbb{E}\mathbf{tr}\left(\bar{X}^\top \bar{X}^\top (X^\top X)^{-1}\right) = 0 \xrightarrow{\textit{Phase 1}} p\delta \xrightarrow{\textit{Phase 2}} p(1-\delta) \xrightarrow{\textit{Phase 3}} \begin{array}{c}\text{certain}\\ \text{accuracy}\end{array}$$

Figure 2: Three-phase analysis $(X = X^{(k)}, \ \delta \in (0, 1/2))$.

Proof of Theorem 3.2. The proof consists of four parts. The first three are for analyzing the behaviours of SGN in different phases as being shown in Subsection 3.2 using the ODE/SDE approximations given in Lemma 3.5 and Lemma 3.7. The last part is dedicated to bringing the first three parts together to obtain the total iteration required for the convergence to a neighbourhood of the optimal set with certain order of accuracy.

[Phase 1]. We investigate into the worst case where the initial point is at or around the lowest saddle point with $\mathcal{I}_p = \{i_1, \cdots, i_p\} \subset \{p'+1, \cdots, n\}$. In this case, Lemma 3.7 suggests that the (l, r)-th element of $\bar{Y}$ satisfies the following SDE

$$dy_{lr}(t) = \left(\frac{\lambda_l}{\lambda_{i_r}} - 1\right) y_{lr} dt + h^{-\frac{1}{2}} \lambda_l^{\frac{1}{2}} dB(t), \qquad 1 \leq l \leq p', \quad 1 \leq r \leq p, \tag{3.16}$$

which is an Ornstein-Uhlenbeck process [46] with formal solution

$$y_{lr}(t) = y_{lr}(0) \exp\left\{ \left(\frac{\lambda_l}{\lambda_{i_r}} - 1\right) t \right\} + h^{-\frac{1}{2}} \lambda_l^{\frac{1}{2}} \int_0^t \exp\left\{ \left(\frac{\lambda_l}{\lambda_{i_r}} - 1\right)(t-s) \right\} dB(s)$$

$$= h^{-\frac{1}{2}}\lambda_l^{\frac{1}{2}}\int_0^t \exp\left\{\left(\frac{\lambda_l}{\lambda_{i_r}}-1\right)(t-s)\right\}dB(s).$$

Calculating the expectational square and placing the timescale t back to $k\alpha$, we obtain

$$\mathbb{E}y_{lr}^2(t)=\frac{\lambda_l}{h}\int_0^t \exp\left\{2\left(\frac{\lambda_l}{\lambda_{i_r}}-1\right)(t-s)\right\}ds=\frac{\lambda_l\lambda_{i_r}}{2h(\lambda_{i_r}-\lambda_l)}\left(1-\exp\left\{\frac{2(\lambda_l-\lambda_{i_r})}{\lambda_{i_r}}k\alpha\right\}\right).$$

Recall that $Y_\alpha(t)=\alpha^{-\frac{1}{2}}X_\alpha(t)V$, we have for $1\leq l\leq p'$

$$\begin{aligned}\mathbb{E}e_l^\top XX^\top e_l=\alpha\mathbb{E}e_l^\top YY^\top e_l&=\sum_{r=1}^p\frac{\alpha\lambda_l\lambda_{i_r}}{2h(\lambda_l-\lambda_{i_r})}\left(\exp\left\{\frac{2(\lambda_l-\lambda_{i_r})}{\lambda_{i_r}}k\alpha\right\}-1\right)\\&\geq\frac{\alpha}{2h}\frac{\lambda_1\lambda_n}{\lambda_1-\lambda_n}\sum_{r=1}^p\left(\exp\left\{\frac{2(\lambda_l-\lambda_{i_r})}{\lambda_{i_r}}k\alpha\right\}-1\right)\\&>\frac{\alpha p}{2h}\frac{\lambda_1\lambda_n}{\lambda_1-\lambda_n}\left(\exp\left\{\frac{2\nu}{\lambda_p}k\alpha\right\}-1\right),\end{aligned}$$

The boundedness of the solution $X(t)$ to ODE (3.7), which is given in Lemma A.1, indicates that $\sigma^2_{\max}(X)<\infty$. Then, we have

$$\mathbb{E}e_l^\top X(X^\top X)^{-1}X^\top e_l>\frac{1}{\sigma^2_{\max}(X)}\frac{\alpha p}{2h}\frac{\lambda_1\lambda_n}{\lambda_1-\lambda_n}\left(\exp\left\{\frac{2\nu}{\lambda_p}k\alpha\right\}-1\right).\tag{3.17}$$

To arrive at the target of this stage, we take the right-hand side of (3.17) to be δ and obtain

$$\frac{\alpha p}{2h}\frac{\lambda_1\lambda_n}{\lambda_1-\lambda_n}\left(\exp\left\{\frac{2\nu}{\lambda_p}k\alpha\right\}-1\right)=\delta\sigma^2_{\max}(X),\tag{3.18}$$

which thus yields the estimated passage time of the first phase:

$$k_{\alpha,1}\asymp\frac{1}{2}\alpha^{-1}\lambda_p\nu^{-1}\log(\alpha^{-1}).\tag{3.19}$$

Note from (3.17) and (3.18) that in this phase, we demand additionally that the square length of each row of $\bar{X}^\top(X^\top X)^{-\frac{1}{2}}$ should move from zero towards δ so that $\bar{X}^{(k_{\alpha,1})}$ is of full rank. Thus, in the next phase, the iterate would not converge to some stationary point out of $\mathcal{X}^*$.
[Phase 2]. Let $R=\mathbf{tr}(X^\top\Sigma X(X^\top X)^{-1})$. By Lemma 3.5, we attain

$$\begin{aligned}\frac{dR}{dt}=&\mathbf{tr}\left[\left(\frac{dX^\top}{dt}\Sigma+X^\top\Sigma\frac{dX}{dt}\right)(X^\top X)^{-1}-X^\top\Sigma X(X^\top X)^{-1}\frac{dX^\top X}{dt}(X^\top X)^{-1}\right]\\=&2\mathbf{tr}\left[(X^\top X)^{-2}X^\top\Sigma^2X-(X^\top X)^{-2}X^\top\Sigma X(X^\top X)^{-1}X^\top\Sigma X\right]\\=&\frac{1}{2}\left\|(I_n-X(X^\top X)^{-1}X^\top)\nabla f_{\mathrm{E}}(X)(X^\top X)^{-1}\right\|_{\mathrm{F}}^2\geq0,\end{aligned}$$

which implies that $R(t)$ will keep increasing until $\nabla f_{\mathrm{E}}(X)=0$. Then, we have

$$\begin{aligned}R(X^*)-R(X)&=\sum_{i=1}^p\lambda_i-\sum_{j=1}^n\lambda_j\mathbf{tr}\left((X^\top X)^{-1}X_{(j,:)}^\top X_{(j,:)}\right)\\&=\sum_{i=1}^p\lambda_i\left(1-X_{(i,:)}(X^\top X)^{-1}X_{(i,:)}^\top\right)-\sum_{j=p+1}^n\lambda_jX_{(j,:)}(X^\top X)^{-1}X_{(j,:)}^\top\\&\geq\lambda_p\sum_{i=1}^p\left(1-X_{(i,:)}(X^\top X)^{-1}X_{(i,:)}^\top\right)-\sum_{j=p+1}^n\lambda_jX_{(j,:)}(X^\top X)^{-1}X_{(j,:)}^\top\end{aligned}$$

$$
\begin{aligned}
&= \lambda_p \left(p - \sum_{i=1}^{p'} X_{(i,:)}(X^\top X)^{-1}X_{(i,:)}^\top \right) - \tau \sum_{j=p'+1}^{n} X_{(j,:)}(X^\top X)^{-1}X_{(j,:)}^\top \\
&= (\lambda_p - \tau)\left(p - \mathbf{tr}(\bar{X}^\top \bar{X}(X^\top X)^{-1}\right) > 0,
\end{aligned}
$$

where $X^* \in \mathcal{X}^*$ and $\lambda_n \le \tau < \lambda_p$ satisfying

$$
\sum_{j=p'+1}^{n} \lambda_j X_{(j,:)}(X^\top X)^{-1}X_{(j,:)}^\top = \tau \sum_{j=p'+1}^{n} X_{(j,:)}(X^\top X)^{-1}X_{(j,:)}^\top.
$$

That is to say, we obtain a lower bound of $\mathbf{tr}\left(\bar{X}^\top \bar{X}^\top (X^\top X)^{-1}\right)$ using $R(X)$. Then, by setting $R(X^*) - R(X) \le (\lambda_p - \tau)p\delta$, $t = k\alpha$, we have

$$
k_{\alpha,2} \asymp \alpha^{-1} R^{-1} \left(\sum_{i=1}^{p} \lambda_i - (\lambda_p - \tau)p\delta \right). \tag{3.20}
$$

[Phase 3]. This stage shows oscillations around some $X^* \in \mathcal{X}^*$, in which case $X_\alpha(t)$ could be written as

$$
X_\alpha(t) = E_{p'} \Lambda_{p'}^{\frac{1}{2}} W V^\top + \mathcal{O}(\alpha^{\frac{1}{2}}), \tag{3.21}
$$

where $E_{p'} = [e_1, e_2, \cdots, e_{p'}] \in \mathbb{R}^{n \times p'}$, $\Lambda_{p'} = \mathbf{Diag}(\lambda_1, \lambda_2, \cdots, \lambda_{p'}) \in \mathbb{R}^{p' \times p'}$,

$$
W = \begin{bmatrix} I_{p-1} & 0_{(p-1)\times 1} \\ 0_{(p'-p+1)\times(p-1)} & w \end{bmatrix} \in \mathbb{R}^{p' \times p}, \quad w^\top w = 1, \quad w \in \mathbb{R}^{p'-p+1}.
$$

In this situation, Lemma 3.7 indicates that the (l,r)-th entry of $\underline{Y}$ takes the form

$$
y_{lr}(t) = y_{lr}(0) \exp\left\{ \frac{(\lambda_l - \lambda_r)}{\lambda_r} t \right\} + h^{-\frac{1}{2}} \lambda_l^{\frac{1}{2}} \int_0^t \exp\left\{ \frac{(\lambda_l - \lambda_r)}{\lambda_r}(t-s) \right\} dB(s).
$$

Then we have the expectational square

$$
\begin{aligned}
\mathbb{E} y_{lr}^2(t) &= y_{lr}^2(0) \exp\left\{ \frac{2(\lambda_l - \lambda_r)}{\lambda_r} t \right\} + h^{-1}\lambda_l \int_0^t \exp\left\{ \frac{2(\lambda_l - \lambda_r)}{\lambda_r}(t-s) \right\} ds \\
&= \exp\left\{ \frac{2(\lambda_l - \lambda_r)}{\lambda_r} t \right\} \left(y_{lr}^2(0) - \frac{\lambda_l \lambda_r}{2h(\lambda_r - \lambda_l)} \right) + \frac{\lambda_l \lambda_r}{2h(\lambda_r - \lambda_l)}.
\end{aligned}
$$

According to the matrix perturbation theory [41], we have

$$
(X^\top X)^{-1} = \left(V \Lambda_{p'}^{\frac{1}{2}} W^\top E_{p'}^\top E_{p'} \Lambda_{p'}^{\frac{1}{2}} W V^\top + \mathcal{O}\left(\alpha^{\frac{1}{2}}\right) \right)^{-1} = V \Lambda_p^{-1} V^\top + \mathcal{O}\left(\alpha^{\frac{1}{2}}\right).
$$

Then, using the relation $Y_\alpha(t) = \alpha^{-\frac{1}{2}} X_\alpha(t) V$, $t = k\alpha$ and the formula (3.21), we get

$$
\begin{aligned}
&\mathbb{E}\mathbf{tr}\left(\underline{X}^\top \underline{X} (X^\top X)^{-1} \right) = \alpha \mathbb{E}\mathbf{tr}\left(V \underline{Y}^\top \underline{Y} V^\top V \Lambda_p^{-1} V^\top \right) + \mathcal{O}\left(\alpha^{\frac{3}{2}}\right) \\
&= \sum_{l=p'+1}^{n} \sum_{r=1}^{p} \exp\left\{ \frac{2(\lambda_l - \lambda_r)}{\lambda_r} k\alpha \right\} \frac{\alpha y_{lr}^2(0)}{\lambda_r} + C \sum_{l=p'+1}^{n} \sum_{r=1}^{p} \frac{\lambda_l}{2h(\lambda_r - \lambda_l)} \alpha + \mathcal{O}\left(\alpha^{\frac{3}{2}}\right) \\
&\lesssim p\delta \cdot \exp\left\{ -\frac{2\nu}{\lambda_p} k\alpha \right\} + C \sum_{l=p'+1}^{n} \sum_{r=1}^{p} \frac{\lambda_l}{2h(\lambda_r - \lambda_l)} \alpha + \mathcal{O}\left(\alpha^{\frac{3}{2}}\right),
\end{aligned}
$$

where C is some positive constant. Then, we set $p\delta \cdot \exp\left\{-2\nu\lambda_p^{-1}k\alpha\right\} = o(\alpha)$, which thus gives the traversing time of the last phase

$$k_{\alpha,3} = \frac{1}{2}\log(p\alpha^{-1}\delta)\lambda_p\nu^{-1}\alpha^{-1} \asymp \frac{1}{2}\log(\alpha^{-1})\lambda_p\nu^{-1}\alpha^{-1}. \tag{3.22}$$

[Combination of three phases]. Based on the Markov property of SGN update, summing up (3.19), (3.20) and (3.22) gives an estimate on the global convergence time of SGN asymptotically:

$$k_\alpha = k_{\alpha,1} + k_{\alpha,2} + k_{\alpha,3} \asymp \lambda_p\nu^{-1}\log(\alpha^{-1})\alpha^{-1}.$$

The expectational estimation error of the approximation $X^{(k_\alpha)}$ is

$$\mathbb{E}\|\sin\Theta(X^{(k_\alpha)}, U_{p'})\|_{\mathrm{F}}^2 \lesssim \frac{C}{h}\sum_{l=p'+1}^{n}\sum_{r=1}^{p}\frac{\lambda_l}{2(\lambda_r-\lambda_l)}\alpha + o(\alpha). \tag{3.23}$$

Since the sample size m is known, the total number of iterations could be expressed as $K = \lceil m/h \rceil$. Let

$$\widetilde{\alpha}(K) = \frac{\lambda_p \log K}{\nu K}, \tag{3.24}$$

we have

$$k_{\widetilde{\alpha}} \asymp \left(\lambda_p\nu^{-1}\widetilde{\alpha}^{-1}\log\widetilde{\alpha}^{-1}\right) \asymp K.$$

Substituting (3.24) into (3.23) yields the bound for SGN. □

3.3 Main results on diminishing stepsizes

The forementioned diffusion approximation theory can also be applied to the diminishing stepsize case

$$\alpha^{(k)} = \frac{\gamma}{c_1(k+c_2)^\beta}, \quad 0<\beta<1, \quad c_1,\ c_2>0, \tag{3.25}$$

with infinitesimal γ. The corresponding differential equation approximations could be obtained by making a slight modification on the construction of the continuous-time extension of SGN iterates $\{X^{(k)}\}$ using the stepsize (3.25), that is,

$$X_\gamma(t) = X^{(\lfloor t\gamma^{\frac{1}{\beta-1}}\rfloor)}.$$

In a way similar to the constant stepsize case, we compute the infinitesimal characteristics (3.5), (3.6), and have the results in Lemma 3.9 and Lemma 3.11. There is no essential difference of the proofs of them from those of Lemma 3.5 and Lemma 3.7, and thus are omitted here.

Lemma 3.9. *Under the assumptions [A1]-[A4], the Markov process $X_\gamma(t) = X^{(\lfloor t\gamma^{\frac{1}{\beta-1}}\rfloor)}$ generated by Algorithm 1 with the stepsize (3.25) and $c_2 = \gamma^{\frac{1}{\beta-1}}$ converges in distribution to the solution of*

$$\frac{dX}{dt} = \frac{1}{c_1(t+1)^\beta}\left(\Sigma X(X^\top X)^{-1} - \frac{1}{2}X - \frac{1}{2}X(X^\top X)^{-1}X^\top\Sigma X(X^\top X)^{-1}\right), \tag{3.26}$$

as $\gamma \to 0$ with $X(0) = X^{(0)}$.

Remark 3.10. *If the parameter c_2 in (3.25) is set to a constant independent of γ, we would have the following limiting differential equation as $\alpha \to 0$*

$$\frac{dX}{dt} = \frac{1}{c_1 t^\beta}\left(\Sigma X(X^\top X)^{-1} - \frac{1}{2}X - \frac{1}{2}X(X^\top X)^{-1}X^\top\Sigma X(X^\top X)^{-1}\right), \tag{3.27}$$

which are almost the same as (3.26) except for the multiplicative factor. In the subsequent discussions, we will make full use of (3.26) because it avoids the singularity of (3.27) at $t = 0$.

Lemma 3.11. *Suppose that the assumptions [A1]-[A4] hold and the initial point is around some rank-p stationary point of (1.5), which is associated with an eigen-index set $\mathcal{I}_p = \{i_1, \cdots, i_p\}$ and an orthogonal matrix $V \in \mathbb{R}^{p\times p}$. We define a new process*

$$Y_\gamma(t) = \gamma^{-\frac{1}{2(1-\beta)}} X_\gamma(t) V, \tag{3.28}$$

generated by Algorithm 1 with the stepsize (3.25) and $c_2 = \gamma^{\frac{1}{\beta-1}}$. If $Y_\gamma(0)$ converges in distribution to some constant $Y^0 \in \mathbb{R}^{n\times p}$ as $\gamma \to 0$ and the index l is not in the extended eigen-index set $\hat{\mathcal{I}}_p$, then the (l,r)-th entry of $Y_\gamma(t)$ converges in distribution to the following SDE with $Y_{(l,r)}(0) = Y^0_{(l,r)}$

$$dY_{(l,r)} = \frac{1}{c_1(t+1)^\beta}\left(\Sigma Y \Lambda_p^{-1} - Y\right)_{(l,r)} dt + \frac{\lambda_l^{\frac{1}{2}}}{c_1 h^{\frac{1}{2}}(t+1)^\beta} dB(t), \qquad 1 \le r \le p, \tag{3.29}$$

where $\Lambda_p = \mathbf{Diag}(\lambda_{i_1}, \lambda_{i_2}, \cdots, \lambda_{i_p}) \in \mathbb{R}^{p\times p}$ and $B(t)$ is the Brownian motion.

The additional multiplicative factor $c_1^{-1}(t+1)^{-\beta}$ makes the limiting differential equations (3.26) and (3.29) time-inhomogeneous, which calls for a more sophisticated analysis than in the constant stepsize case in order to obtain the convergence rate.

The following theorem states the convergence result for the diminishing stepsize case.

Theorem 3.12. *Suppose that the assumptions [A1]-[A4] hold. Given the total sample size $m > 0$, and the stepsize (3.25) with*

$$c_1 = \frac{\nu}{\lambda_p}, \quad c_2 = \gamma^{\frac{1}{\beta-1}}, \quad \beta = 1 - \frac{1}{\log K}, \quad \gamma = \frac{(1-\beta)\log K}{K^{1-\beta}}, \quad K = \left\lceil \frac{m}{h} \right\rceil,$$

where $k = 0, 1, \cdots, K$, the expectational error of the final output $X^{(K)}$ of Algorithm 1 is given by

$$\mathbb{E}\|\sin\Theta(X^{(K)}, U_{p'})\|_{\mathrm{F}}^2 \lesssim \frac{C}{2h} p(n-p') \frac{\lambda_p \lambda_{p'+1}}{\nu^2} K^{-1}, \tag{3.30}$$

where C is some positive constant.

Proof. We adopt the framework same as that in the proof of Theorem 3.2, where the sequence of SGN iterates with the diminishing stepsize (3.25) is described by a three-phase process.

[Phase 1]. We take the initial condition $\mathbb{E}\mathbf{tr}(\bar{X}^\top \bar{X}(X^\top X)^{-1}) = 0$ and consider the worst case for which the initial point is at or around the lowest saddle point with $\mathcal{I}_p = \{i_1, \cdots, i_p\} \subset \{p'+1, \cdots, n\}$. As Lemma 3.11 suggests in this case, the (l,r)-th element of $\bar{Y}$ is the solution to the following SDE

$$dy_{lr}(t) = \frac{1}{c_1(t+1)^\beta}\left(\frac{\lambda_l}{\lambda_{i_r}} - 1\right) y_{lr} dt + \frac{\lambda_l^{\frac{1}{2}}}{c_1 h^{\frac{1}{2}}(t+1)^\beta} dB(t), \quad 1 \le l \le p', \quad 1 \le r \le p,$$

whose solution has an explicit form

$$y_{lr}(t) = \Phi(t)\left(y_{lr}(0) + \frac{\lambda^{\frac{1}{2}}}{c_1 h^{\frac{1}{2}}} \int_0^t \Phi^{-1}(s)(s+1)^{-\beta} dB(s)\right),$$

where

$$\Phi(t) = \exp\left\{\frac{\lambda_l - \lambda_{i_r}}{c_1 \lambda_{i_r}} \frac{(t+1)^{1-\beta} - 1}{1-\beta}\right\}.$$

The expectational square of $y_{lr}(t)$ then satisfies

$$\mathbb{E}y_{lr}^2(t) = \frac{\lambda_l}{c_1^2 h}\Phi^2(t)\int_0^t \Phi(s)^{-2}(s+1)^{-2\beta} dB(s)$$

$$\geq C_1 \left(\exp\left\{ \frac{2(\lambda_l - \lambda_{i_r})}{c_1 \lambda_{i_r}} \frac{1}{1-\beta} \left((t+1)^{1-\beta} - 1 \right) \right\} - 1 \right), \tag{3.31}$$

where

$$C_1 = \frac{\lambda_l}{c_1^2 h (1-\beta)} \left(\frac{(1-\beta) c_1 \lambda_{i_r}}{2(\lambda_l - \lambda_{i_r})} \right)^{\frac{1-2\beta}{1-\beta}} \frac{2(\lambda_l - \lambda_{i_r})}{\beta c_1 \lambda_{i_r} + 2(\lambda_l - \lambda_{i_r})}.$$

The inequality (3.31) could be attained by taking the variable transformation

$$z = \frac{2(\lambda_l - \lambda_{i_r})}{c_1 \lambda_{i_r} (1-\beta)} (s+1)^{1-\beta},$$

which leads to the difference of two incomplete Gamma functions.

Using the relation $Y_\gamma(t) = \gamma^{-\frac{1}{2(1-\beta)}} X_\gamma(t) V$ and placing the timescale t back to $k\gamma^{\frac{1}{1-\beta}}$, we have for $1 \leq l \leq p'$

$$\begin{aligned} &\mathbb{E} e_l^\top X (X^\top X)^{-1} X^\top e_l \\ >& \sum_{r=1}^{p} \frac{C_1 \gamma^{\frac{1}{1-\beta}}}{\sigma_{\max}^2(X)} \left(\exp\left\{ \frac{2(\lambda_l - \lambda_{i_r})}{c_1 \lambda_{i_r}} \frac{1}{1-\beta} \left((k\gamma^{\frac{1}{1-\beta}} + 1)^{1-\beta} - 1 \right) \right\} - 1 \right). \end{aligned} \tag{3.32}$$

Taking the right side of (3.32) to be δ gives the passage time of the first phase:

$$k_{\gamma,1}^{1-\beta} \asymp \frac{c_1 \lambda_p}{2\nu} \gamma^{-1} \log(\gamma^{-1}). \tag{3.33}$$

[Phase 2]. Following the same steps as in the proof of Theorem 3.2 directly yields the estimated traversing time of the second phase:

$$k_{\gamma,2}^{1-\beta} = \mathcal{O}(\gamma^{-1}). \tag{3.34}$$

[Phase 3]. This phase depicts the behaviour of SGN oscillating around the optimal set $\mathcal{X}^*$, and in the neighbourhood of $\mathcal{X}^*$, $X_\gamma(t)$ could be formulated like (3.21) as

$$X_\gamma(t) = E_{p'} \Lambda_{p'}^{\frac{1}{2}} W V^\top + \mathcal{O}(\gamma^{\frac{1}{2(1-\beta)}}). \tag{3.35}$$

In this case, Lemma 3.11 indicates that the (l,r)-th entry of $\underline{Y}$ takes the form

$$y_{lr}(t) = \Phi(t) \left(y_{lr}(0) + \frac{\lambda_l^{\frac{1}{2}}}{c_1 h^{\frac{1}{2}}} \int_0^t \Phi^{-1}(s)(s+1)^{-\beta} dB(s) \right),$$

where

$$\Phi(t) = \exp\left\{ \frac{\lambda_l - \lambda_r}{c_1 \lambda_r} \frac{(t+1)^{1-\beta} - 1}{1-\beta} \right\}.$$

As before, we have the expectational square

$$\mathbb{E} y_{lr}^2(t) = \Phi^2(t) \left(y_{lr}^2(0) + \frac{\lambda_l}{c_1^2 h} \int_0^t \Phi^{-2}(s)(s+1)^{-2\beta} ds \right).$$

According to $Y_\gamma(t) = \gamma^{-\frac{1}{2(1-\beta)}} X_\gamma(t) V$ and the expression (3.35), we have

$$\begin{aligned} &\mathbb{E}\mathbf{tr}\left(\underline{X}^\top \underline{X} (X^\top X)^{-1} \right) \\ =&\gamma^{\frac{1}{1-\beta}} \mathbb{E}\mathbf{tr}\left(V \underline{Y}^\top \underline{Y} V^\top V \Lambda_p^{-1} V^\top \right) + \mathcal{O}\left(\gamma^{\frac{3}{2(1-\beta)}} \right) \end{aligned}$$

$$
\begin{aligned}
&\asymp \gamma^{\frac{1}{1-\beta}} \sum_{l=p'+1}^{n} \sum_{r=1}^{p} \exp\left\{ \frac{2(\lambda_l - \lambda_r)}{c_1 \lambda_r} \frac{(t+1)^{1-\beta} - 1}{1-\beta} \right\} \frac{y_{lr}^2(0)}{\lambda_r} \\
&+ \sum_{l=p'+1}^{n} \sum_{r=1}^{p} \frac{\gamma^{\frac{1}{1-\beta}} \lambda_l}{c_1^2 h \lambda_r} \exp\left\{ \frac{2(\lambda_l - \lambda_r)}{c_1 \lambda_r} \frac{(t+1)^{1-\beta}}{1-\beta} \right\} \int_0^t \exp\left\{ \frac{2(\lambda_r - \lambda_l)}{c_1 \lambda_r} \frac{(s+1)^{1-\beta}}{1-\beta} \right\} (s+1)^{-2\beta} ds \\
&\le \; p\delta \exp\left\{ \frac{2\nu}{c_1 \lambda_p} \frac{(t+1)^{1-\beta} - 1}{1-\beta} \right\} \\
&+ \sum_{l=p'+1}^{n} \sum_{r=1}^{p} \frac{\gamma^{\frac{1}{1-\beta}} \lambda_l}{c_1^2 h \lambda_r} \exp\left\{ \frac{2(\lambda_l - \lambda_r)}{c_1 \lambda_r} \frac{(t+1)^{1-\beta}}{1-\beta} \right\} \int_0^t \exp\left\{ \frac{2(\lambda_r - \lambda_l)}{c_1 \lambda_r} \frac{(s+1)^{1-\beta}}{1-\beta} \right\} (s+1)^{-\beta} ds,
\end{aligned}
$$

Then, using $t = k\gamma^{\frac{1}{1-\beta}}$ and setting

$$
p\delta \exp\left\{ \frac{2\nu}{c_1 \lambda_p} \frac{(t+1)^{1-\beta} - 1}{1-\beta} \right\} = o(\gamma^{\frac{1}{1-\beta}}),
$$

give the traversing time of the last phase:

$$
k_{\gamma,3}^{1-\beta} = \frac{c_1 \lambda_p}{2\nu} \gamma^{-1} \log(\gamma^{-1}). \tag{3.36}
$$

[Combination of three phases]. Adding up (3.19), (3.20) and (3.22), we obtain an estimate on the global convergence time of SGN asymptotically as

$$
k_{\gamma}^{1-\beta} = k_{\gamma,1}^{1-\beta} + k_{\gamma,2}^{1-\beta} + k_{\gamma,3}^{1-\beta} \asymp \frac{c_1 \lambda_p}{\nu} \gamma^{-1} \log(\gamma^{-1}).
$$

The expectational estimation error of the approximation $X^{(k_\gamma)}$ is

$$
\begin{aligned}
&\mathbb{E}\| \sin\Theta(X^{(k_\gamma)}, U_{p'}) \|_{\mathrm{F}}^2 \\
\lesssim\; & \frac{\gamma^{\frac{1}{1-\beta}}}{c_1 h} \sum_{l=p'+1}^{n} \sum_{r=1}^{p} \frac{\lambda_l}{2(\lambda_r - \lambda_l)} \left(1 - \exp\left\{ \frac{2(\lambda_r - \lambda_l)}{c_1 (1-\beta) \lambda_r} \left(1 - (t+1)^{1-\beta} \right) \right\} \right) \\
\le\; & \frac{\lambda_{p'+1} p (n-p')}{c_1 \nu h} \gamma^{\frac{1}{1-\beta}}.
\end{aligned} \tag{3.37}
$$

Given the sample size m, we let

$$
\widetilde{\gamma}(K) = \frac{(1-\beta)\log K}{K^{1-\beta}}, \quad c_1 = \frac{\nu}{\lambda_p}, \tag{3.38}
$$

where $K = \lceil m/h \rceil$ is the total number of iterations, and we have

$$
k_{\widetilde{\gamma}}^{1-\beta} \asymp \left(c_1 \lambda_p \nu^{-1} \widetilde{\gamma}^{-1} \log \widetilde{\gamma}^{-1} \right) \asymp K^{1-\beta}.
$$

Substituting (3.38) into (3.37) and further setting $\beta = 1 - 1/\log K$ give the error bound (3.30) for SGN with the diminishing stepsize (3.25). □

Remark 3.13. *Comparing the bound (3.9) for the constant stepsize case with the one (3.30) for the diminishing stepsize case, we see that the latter bound removes the $\log K$ factor, and thus yields the optimality of SGN method.*

4 Numerical experiments

This section presents the numerical performance of our proposed stochastic GN algorithms for solving OPCA on both synthetic Gaussian data and frequently-used real data in machine learning. All experiments are implemented on a MacBook Pro 3.1GHz laptop with 8GB of RAM using Matlab 2018a.

4.1 Testing examples

We test each competing algorithm on both simulated data of nearly low-rank structure and three of the most popular real datasets in machine learning. Given the integers $p \ll n$ and $p \leq p' \ll n$, the simulated data are generated from an n-dimensional Gaussian distribution $\mathcal{N}_n(0, \Sigma)$ with

$$\Sigma = QDQ^\top + \rho^2 I_n, \tag{4.1}$$

where $Q \in \mathbb{R}^{n \times p'}$ is a randomly generated orthogonal matrix, $D \in \mathbb{R}^{p' \times p'}$ is a diagonal matrix with $D_{ii} = \mu_i \in [\underline{\mu},\ \bar{\mu}]$. We categorize the data into three groups as shown in Subsection 4.1 for specific purpose.

Top Eigenvalues	**Parameters**	**Remark**
`Gau-gap-1` $(p' = p)$		
$\mu_i \sim \mathcal{U}(\underline{\mu}, \bar{\mu})$ sorted in a descending order	$\underline{\mu} = 0.01; \bar{\mu} = 1, 10, 100;$ $p = 1, \cdots, 30$	various gaps & uniformity
`Gau-gap-2` $(p' = p)$		
$\mu_1 = \cdots = \mu_{p_1} = \bar{\mu};$ $\mu_{p_1+1} = \cdots = \mu_p = \underline{\mu}$	$\underline{\mu} = 1;\ \ \bar{\mu} = 100;$ $p = 30;\ \ p_1 = 0, 5, 15, 25$	fixed gap & nonuniformity
`Gau-ngap` $(p' > p)$		
$\mu_j \sim \mathcal{U}(\underline{\mu}, \bar{\mu}),\ \ 1 \leq j \leq p$ sorted in a descending order; $\mu_{p+1} = \cdots = \mu_{p'} = \mu_p$	$\underline{\mu} = 0.01;\ \ \bar{\mu} = 100;$ $p = 30;\ \ p' = 32, 45, 60$	no gap

Table 1: Descriptions of simulated data.
The other parameters are fixed to $n = 500,\ \ m = 10000,\ \ \rho = 0.1$.

The real data examples are described in Subsection 4.1.

Name	**m**	**n**	**Remark**
`MNIST` [22]	10000	784	test set, hand-written digits in gray-scale
`Fashion-MNIST` [53]	10000	784	test set, fashion product in gray-scale
`CIFAR-10` [21]	10000	3072	test set, color images of natural objects

Table 2: Descriptions of real data.

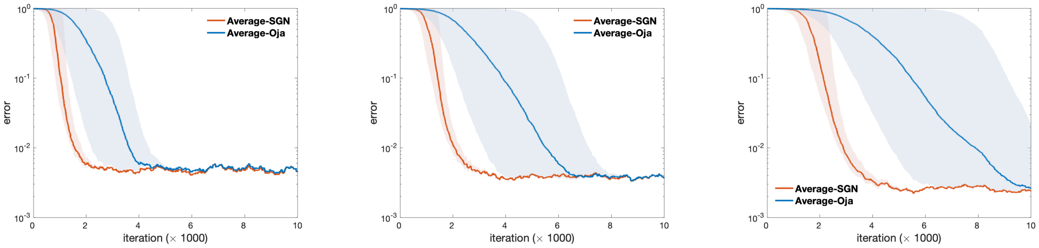
4.2 Default settings of algorithms

The algorithms for comparison are Oja's iteration (1.1) with constant stepsize $\alpha^{(k)} = \alpha$, diminishing stepsize $\alpha^{(k)} = \gamma/(k+1)$, its adaptive variant AdaOja (1.2) and the proposed SGN using constant and diminishing stepsizes along with the adaptive-stepsize version AdaSGN. For the test in Subsection 4.3, the stepsize parameter α is set according to (3.8). For the test in Subsection 4.4, the stepsize parameter γ goes through the set $\{2^{-5}, 2^{-4}, \cdots, 2^4, 2^5\}$ and for the test in Subsection 4.5, γ is optimally selected from this set, used as a benchmark result for the parameter-free AdaSGN. Both of the single-pass $h = 1$ and mini-batch $h = 10, 100$ models are considered in the tests. The accuracy of algorithms is measured by $\|\sin\Theta(X^{(k)}, U_{p'})\|_{\mathrm{F}}^2/p \in [0, 1]$. Due to the inaccessible population covariance Σ in the real-data case, we use the full-sampled empirical one Σ_m to compute $U_{p'}$ as a compromise. All results shown in this paper are the average of 100 runs from random initial points.

4.3 Comparison on constant stepsizes

The experiments in this subsection are to (i) show the three-phase behaviour of SGN as introduced in Section 3 and verify the statement in Remark 3.3; (ii) prove the stability of SGN w.r.t. the random initialization; (iii) validate that SGN produces smaller volatility in the third phase than Oja's iteration when λ_1 is large as analyzed in Remark 3.4, when adopting the constant stepsize.

To provide a fair comparison with the three-phase result of Oja's iteration in [24], we set $h = 1$ and $p = p' = 1$ in this test. Under the case of fixed α but 100 random initializations and the case of fixed initial point $X^{(0)} = e_n$ but different α and μ_1, the averaged results of SGN and Oja's iteration with variance shaded are presented in Figure 3 and Figure 4, respectively, where the "Z"-shaped trajectories are consistent with the three-phase analysis in the theory. As could be observed in Figure 3, the initialization has obvious impact on the stability of Oja's itertiaon, and however has little effect on SGN. Fixing the initial value to the saddle point $X^{(0)} = e_n$, Figure 4 implies that as the top eigenvalue gets larger, the Oja's iteration fluctuates more significantly and yields results of much lower accuaray than SGN, and thus requires smaller stepsizes for the purposes of obtaining both the smaller oscillation and the higher accuracy, which agrees with the statements in Remark 3.3 and Remark 3.4.



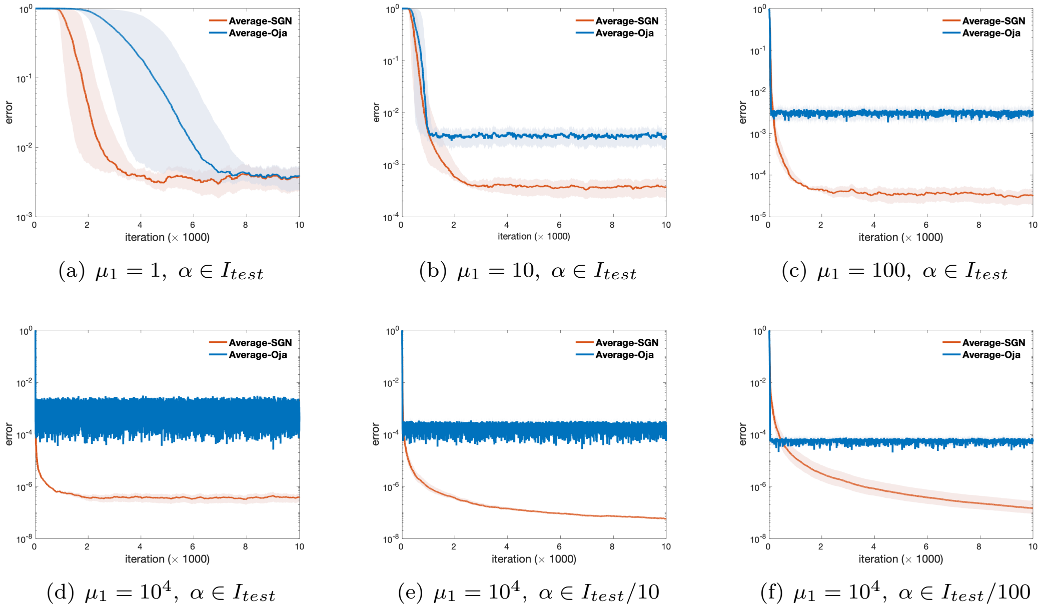
(a) $\alpha = 4 \times 10^{-3}$ (b) $\alpha = 2 \times 10^{-3}$ (c) $\alpha = 1 \times 10^{-3}$

Figure 3: The estimation error of Oja's iteration and SGN using constant stepsizes $\alpha^{(k)} = \alpha$ on simulated data with $\mu_1 = 1$ from 100 randomly generated initial points.

4.4 Comparison on diminishing stepsizes

In this subsection, we are going to demonstrate the robustness of SGN over Oja's iteration when adopting the diminishing stepsize $\alpha^{(k)} = \gamma/(k + 1)$ in terms of the performance as γ varies and the optimally-tuned γ from $\{2^{-5}, 2^{-4}, \cdots, 2^4, 2^5\}$.

In Figures 5 to 7, the gradation in color indicates the change of parameter γ and the deeper-colored dotted line represents the larger value of parameter γ used. The performance of the proposed SGN is shown in the warm tone while the one of Oja's iteration is shown in the cold tone. Here, we focus on the overall performance of two algorithms w.r.t. the variation in γ. In addition, the optimal performer is highlighted in sold line with corresponding γ being bolded in the legend. Figure 5 exhibits the results of processing `Gau-gap-1` with $\bar{\mu} = 10$. When the number of PCs (i.e., p) increases, on one side, SGN exhibits a better performance than Oja's iteration in overall performance. On the other side, the optimal γ for Oja's iteration goes from 2^{-2} to 2^5 in the single-pass model while the one for SGN adimts a slight change from 1 to 2 in the single-pass model and stays in the value of one in the mini-batch model. Similar results could be observed in Figure 7 for `CIFAR-10` dataset. Moreover, in `Gau-gap-2`, we fix the gap and p, and change the distribution of top p eigenvalues. As being seen in Figure 6, Oja's iteration becomes dramatically worse as p_1 increases, while SGN could manage it within the given parameter set. To save space, the results of other datasets listed in Subsection 4.1 are not provided here, which are in a good agreement with the above observations.



(a) $\mu_1 = 1,\ \alpha \in I_{test}$

(b) $\mu_1 = 10,\ \alpha \in I_{test}$

(c) $\mu_1 = 100,\ \alpha \in I_{test}$

(d) $\mu_1 = 10^4,\ \alpha \in I_{test}$

(e) $\mu_1 = 10^4,\ \alpha \in I_{test}/10$

(f) $\mu_1 = 10^4,\ \alpha \in I_{test}/100$

Figure 4: The estimation error of Oja's iteration and SGN using $\alpha^{(k)} = \alpha$ with 100 evenly spaced $\alpha \in I_{test} = [9 \times 10^{-4}, 2 \times 10^{-3}], I_{test}/10, I_{test}/100$ starting at one saddle point.

Further, we present the best parameter $\gamma \in \{2^{-5}, 2^{-4}, \cdots, 2^4, 2^5\}$ in handling various datasets in Subsection 4.4 and Subsection 4.4. The color of the area $\{\gamma \leq 1\}$ is filled with gray. It is apparent that the value of the best γ for SGN varies smoothly in both different datasets and number of PCs, and however the one for Oja's iteration changes greatly. And interestingly, a larger p always calls for a larger γ.

We have also performed tests on the diminishing stepsize (3.25) with different choices of β, which produce the results similar to the case of $\beta = 1$ as shown in Figure 5-**??**. For brevity,, these results are not presented in this paper.

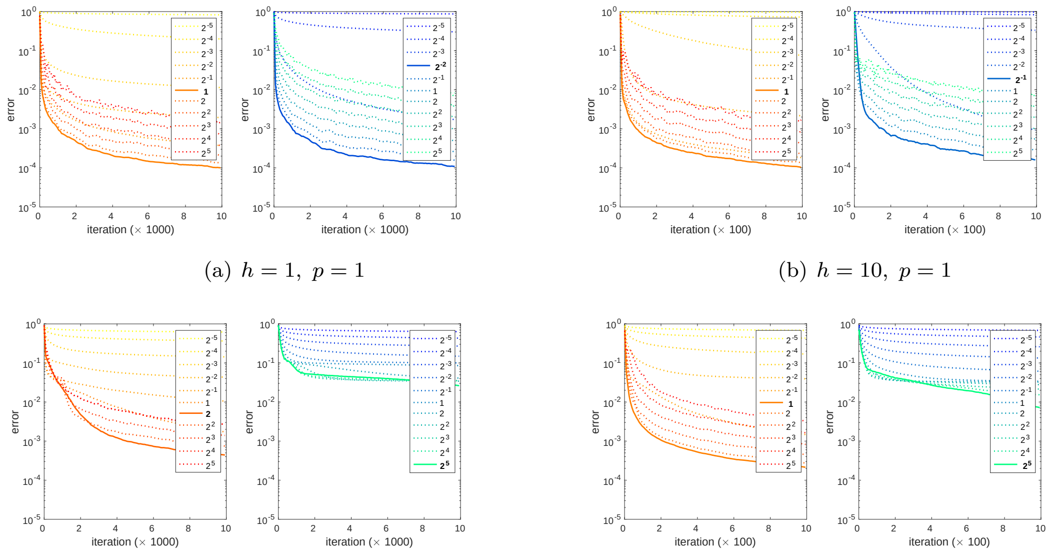
4.5 Comparison on adaptive stepsizes

Finally, we try to demonstrate the effectiveness of AdaSGN compared to the diminishing-stepsize Oja's iteration and SGN with the optimally tuned γ from $\{2^{-5}, 2^{-4}, \cdots, 2^4, 2^5\}$ and AdaOja (1.2).

In Figure 10 and Figure 11, AdaSGN shows comparable performance with manually tuned SGN while AdaOja produces unstable results in different settings (for example, good performance in Figure 10(c) but poor performance in Figure 10(f)). In addition, both SGN and AdaSGN exhibit excellent stability against the batchsize change but Oja's iteration and AdaOja suggest otherwise. And an appropriate batchsize could make a certain improvement (for example, Figure 10(d) and Figure 10(e)).

5 Concluding remarks

In this paper, we develop an SGN method for solving the OPCA problem, which exhibits the empirical robustness with respect to the varying of the stepsize parameter and input data. By adopting the diffusion approximation, we provide the global convergence of SGN under sub-Gaussian samples without traditional gap assumption. Then, to avoid stepsize tuning which often utilizes prior information on the input data, we focus on the sample consistency and design an adaptive scheme for SGN called AdaSGN, with a basic idea of assigning smaller stepsizes for samples of lower consistency. Numerical results indicate that AdaSGN shows a better performance than the adaptive version of Oja method and is comparable with SGN using diminishing stepsizes of manual selection.



(a) $h = 1,\ p = 1$

(b) $h = 10,\ p = 1$

(c) $h = 1,\ p = 30$

(d) $h = 10,\ p = 30$

Figure 5: The estimation errors of Oja's iteration (right in each subfigure) and SGN (left in each subfigure) with diminishing stepsizes when varying γ from 2^{-5} to 2^5 on `Gau-gap-1` with $\bar{\mu} = 10$.

Several aspects of this work deserve further studies. First, our theoretical results are established on the stepsize scheme (3.25) with infinitesimal γ and $\beta \in [0, 1)$. In spite of the desirable empirical performance of SGN with $\beta \in [0, 1]$, it is worthwhile to extend the analysis techniques to the case of $\beta = 1$ so that the gap between the theory and practice could be filled. Then, how our proposed adaptive strategy is analyzed theoretically and whether it will work with other OPCA algorithms remain to be answered. Additionally, developing variants of SGN for different settings (e.g., sparsity on PCs, capability of missing data) is of particular interest as well.

References

[1] Z. Allen-Zhu and Y. Li, *First efficient convergence for streaming k-pca: a global, gap-free, and near-optimal rate*, in 2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS), IEEE, 2017, pp. 487–492.

[2] M.-F. Balcan, S. S. Du, Y. Wang, and A. W. Yu, *An improved gap-dependency analysis of the noisy power method*, in Conference on Learning Theory, PMLR, 2016, pp. 284–309.

[3] A. Balsubramani, S. Dasgupta, and Y. Freund, *The fast convergence of incremental pca*, in Advances in Neural Information Processing Systems, vol. 26, 2013, pp. 3174–3182.

[4] L. Balzano, Y. Chi, and Y. M. Lu, *Streaming pca and subspace tracking: The missing data case*, Proceedings of the IEEE, 106 (2018), pp. 1293–1310.

[5] H. Cardot and D. Degras, *Online principal component analysis in high dimension: Which algorithm to choose?*, International Statistical Review, 86 (2018), pp. 29–50.

[6] M. Chen, L. F. Yang, M. Wang, and T. Zhao, *Dimensionality reduction for stationary time series via stochastic nonconvex optimization*, in Proceedings of the 32nd International Conference on Neural Information Processing Systems, 2018, pp. 3500–3510.

[7] C. De Sa, C. Re, and K. Olukotun, *Global convergence of stochastic gradient descent for some non-convex matrix problems*, in International Conference on Machine Learning, PMLR, 2015, pp. 2332–2341.

[8] S. N. Ethier and T. G. Kurtz, *Markov processes: characterization and convergence*, Wiley, 1986.

[9] Y. Feng, L. Li, and J.-G. Liu, *Semigroups of stochastic gradient descent and online principal component analysis: properties and diffusion approximations*, Communications in Mathematical Sciences, 16 (2018), pp. 777–789.

[10] B. Gao, X. Liu, and Y.-x. Yuan, *Parallelizable algorithms for optimization problems with orthogonality constraints*, SIAM Journal on Scientific Computing, 41 (2019), pp. A1949–A1983.

[11] G. H. Golub and C. F. Van Loan, *Matrix computations*, vol. 3, JHU press, 2013.

[12] M. Hardt and E. Price, *The noisy power method: a meta algorithm with applications*, in Proceedings of the 27th International Conference on Neural Information Processing Systems-Volume 2, 2014, pp. 2861–2869.

[13] A. Henriksen and R. Ward, *Adaoja: Adaptive learning rates for streaming pca*, arXiv preprint arXiv:1905.12115, (2019).

[14] D. Huang, J. Niles-Weed, and R. Ward, *Streaming k-pca: Efficient guarantees for oja's algorithm, beyond rank-one updates*, in Conference on Learning Theory, PMLR, 2021, pp. 2463–2498.

[15] P. Jain, C. Jin, S. M. Kakade, P. Netrapalli, and A. Sidford, *Streaming pca: Matching matrix bernstein and near-optimal finite sample guarantees for oja's algorithm*, in Conference on Learning Theory, PMLR, 2016, pp. 1147–1164.

[16] R. Johnson and T. Zhang, *Accelerating stochastic gradient descent using predictive variance reduction*, in Proceedings of the 26th International Conference on Neural Information Processing Systems-Volume 1, 2013, pp. 315–323.

[17] I. T. Jolliffe, *Principal component analysis*, Springer-Verlag, 1986.

[18] I. T. Jolliffe and J. Cadima, *Principal component analysis: A review and recent developments*, Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 374 (2016), p. 20150202.

[19] A. V. Knyazev, *Toward the optimal preconditioned eigensolver: Locally optimal block preconditioned conjugate gradient method*, SIAM Journal on Scientific Computing, 23 (2001), pp. 517–541.

[20] T. Krasulina, *The method of stochastic approximation for the determination of the least eigenvalue of a symmetrical matrix*, USSR Computational Mathematics and Mathematical Physics, 9 (1969), pp. 189–195.

[21] A. Krizhevsky and G. Hinton, *Learning multiple layers of features from tiny images*, Technical Report, Department of Computer Science, University of Toronto, (2009).

[22] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, *Gradient-based learning applied to document recognition*, Proceedings of the IEEE, 86 (1998), pp. 2278–2324.

[23] C. J. Li, M. Wang, L. Han, and Z. Tong, *Near-optimal stochastic approximation for online principal component estimation*, Mathematical Programming, 167 (2016), pp. 75–97.

[24] C. J. Li, M. Wang, H. Liu, and T. Zhang, *Diffusion approximations for online principal component estimation and global convergence*, in Advances in Neural Information Processing Systems, vol. 30, 2017.

[25] C.-L. Li, H.-T. Lin, and C.-J. Lu, *Rivalry of two families of algorithms for memory-restricted streaming pca*, in Artificial Intelligence and Statistics, PMLR, 2016, pp. 473–481.

[26] X. Liu, Z. Wen, and Y. Zhang, *Limited memory block krylov subspace optimization for computing dominant singular value decompositions*, SIAM Journal on Scientific Computing, 35 (2013), pp. A1641–A1668.

[27] ———, *An efficient gauss–newton algorithm for symmetric low-rank product matrix approximations*, SIAM Journal on Optimization, 25 (2015), pp. 1571–1608.

[28] J. C. Lv, Z. Yi, and K. K. Tan, *Global convergence of oja's pca learning algorithm with a non-zero-approaching adaptive learning rate*, Theoretical Computer Science, 367 (2006), pp. 286–307.

[29] E. Oja, *Simplified neuron model as a principal component analyzer*, Journal of Mathematical Biology, 15 (1982), pp. 267–273.

[30] K. Pearson, *Liii. on lines and planes of closest fit to systems of points in space*, The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science, 2 (1901), pp. 559–572.

[31] S. Raychaudhuri, J. M. Stuart, and R. B. Altman, *Principal components analysis to summarize microarray experiments: Application to sporulation time series*, in Biocomputing 2000, World Scientific, 1999, pp. 455–466.

[32] H. Robbins and S. Monro, *A stochastic approximation method*, Annals of Mathematical Statistics, 22 (1951), pp. 400–407.

[33] H. Rutishauser, *Simultaneous iteration method for symmetric matrices*, Numerische Mathematik, 16 (1970), pp. 205–223.

[34] Y. Saad, *On the rates of convergence of the lanczos and the block-lanczos methods*, SIAM Journal on Numerical Analysis, 17 (1980), pp. 687–706.

[35] O. Shamir, *A stochastic pca and svd algorithm with an exponential convergence rate*, in International Conference on Machine Learning, PMLR, 2015, pp. 144–152.

[36] ———, *Convergence of stochastic gradient descent for pca*, in International Conference on Machine Learning, PMLR, 2016, pp. 257–265.

[37] ———, *Fast stochastic algorithms for svd and pca: Convergence properties and convexity*, in International Conference on Machine Learning, PMLR, 2016, pp. 248–256.

[38] G. L. Sleijpen and H. A. Van der Vorst, *A jacobi–davidson iteration method for linear eigenvalue problems*, SIAM Review, 42 (2000), pp. 267–293.

[39] D. C. Sorensen, *Implicitly restarted arnoldi/lanczos methods for large scale eigenvalue calculations*, in Parallel Numerical Algorithms, Springer, 1997, pp. 119–165.

[40] A. Stathopoulos and C. F. Fischer, *A davidson program for finding a few selected extreme eigenpairs of a large, sparse, real, symmetric matrix*, Computer Physics Communications, 79 (1994), pp. 268–290.

[41] G. Stewart and J. Sun, *Matrix perturbation theory*, Academic Press, 1990.

[42] A. Subasi and M. I. Gursoy, *Eeg signal classification using pca, ica, lda and support vector machines*, Expert Systems with Applications, 37 (2010), pp. 8659–8666.

[43] C. Tang, *Exponentially convergent stochastic k-pca without variance reduction*, in Advances in Neural Information Processing Systems, 2019, pp. 12393–12404.

[44] J. A. Tropp, *An introduction to matrix concentration inequalities*, Foundations and Trends in Machine Learning, 8 (2015), pp. 1–230.

[45] M. Turk and A. Pentland, *Eigenfaces for recognition*, Journal of Cognitive Neuroscience, 3 (1991), pp. 71–86.

[46] G. E. Uhlenbeck and L. S. Ornstein, *On the theory of the brownian motion*, Physical Review, 36 (1930), p. 823.

[47] V. Q. Vu and J. Lei, *Minimax sparse principal subspace estimation in high dimensions*, Annals of Statistics, 41 (2013), pp. 2905–2947.

[48] C. Wang and Y. M. Lu, *Online learning for sparse pca in high dimensions: Exact dynamics and phase transitions*, in 2016 IEEE Information Theory Workshop (ITW), IEEE, 2016, pp. 186–190.

[49] L. Wang, B. Gao, and X. Liu, *Multipliers correction methods for optimization problems over the stiefel manifold*, CSIAM Transactions on Applied Mathematics, 2 (2021), pp. 508–531.

[50] R. Ward, X. Wu, and L. Bottou, *Adagrad stepsizes: Sharp convergence over nonconvex landscapes*, in International Conference on Machine Learning, PMLR, 2019, pp. 6677–6686.

[51] Z. Wen, C. Yang, X. Liu, and Y. Zhang, *Trace-penalty minimization for large-scale eigenspace computation*, Journal of Scientific Computing, 66 (2016), pp. 1175–1203.

[52] Z. Wen and Y. Zhang, *Accelerating convergence by augmented rayleigh–ritz projections for large-scale eigenpair computation*, SIAM Journal on Matrix Analysis and Applications, 38 (2017), pp. 273–296.

[53] H. Xiao, K. Rasul, and R. Vollgraf, *Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms*, arXiv preprint arXiv:1708.07747, (2017).

[54] N. Xiao, X. Liu, and Y.-x. Yuan, *A class of smooth exact penalty function methods for optimization problems with orthogonality constraints*, Optimization Methods and Software, (2020), pp. 1–37.

[55] S. Zhou and Y. Bai, *Convergence analysis of oja's iteration for solving online pca with nonzero-mean samples*, Science China Mathematics, 64 (2021), pp. 849–868.

A Boundedness of solution to ODE (3.7)

Lemma A.1. *The solution $X(t)$ to ODE (3.7) is uniformly bounded.*

Proof. Denote $L = \mathbf{tr}(X^\top X)$ and $M = \mathbf{tr}(X^\top \Sigma X(X^\top X)^{-1})$.

$$\begin{aligned}\frac{dL}{dt} &= 2\mathbf{tr}\left(X^\top \frac{dX}{dt}\right)\\ &= \mathbf{tr}\left(2X^\top \Sigma X(X^\top X)^{-1} - X^\top X - X^\top \Sigma X(X^\top X)^{-1}\right) = M - L.\end{aligned} \tag{A.1}$$

The explicit solution of (A.1) is given by

$$L(t) = \exp\{-t\}\left(\int M\exp\{t\}dt + C(L^0)\right),$$

where $C(L^0)$ is some constant depending only on the initial value $L(0) = L^0 = \mathbf{tr}(X^{0\top}X^0)$. Since

$$\lambda_n I_p \preccurlyeq (X^\top X)^{-\frac{1}{2}} X^\top \Sigma X (X^\top X)^{-\frac{1}{2}} \preccurlyeq \lambda_1 I_p,$$

we have

$$p\lambda_n \le M \le p\lambda_1,$$

which then yields the bound

$$p\lambda_n + C(L^0)\exp\{-t\} \le L \le p\lambda_1 + C(L^0)\exp\{-t\}.$$

□

B Proof of Lemma 3.5

We begin with the conditional expected boundedness of the increment of SGN iterates using sufficiently small stepsize.

Lemma B.1. *Suppose the assumptions [A1]-[A3] hold and stepsize $\alpha^{(k)}$ is chosen by*

$$\alpha^{(k)} \leq \min\left\{\frac{\sigma_{\min}^2(X^{(k)})\varphi_2}{2\varphi_4},\ 1\right\}, \tag{B.1}$$

with $\varphi_2 = \mathbb{E}\|\Sigma_h^{(k)}\|, \varphi_4 = \mathbb{E}\|\Sigma_h^{(k)}\|^2$. Then $\{X^{(k)}\}$ generated by Algorithm 1 satisfies

$$\frac{1}{\alpha^{(k)}}\mathbb{E}\left[\left\|(X^{(k+1)} - X^{(k)})\right\|_{\mathrm{F}}^2 \middle| X^{(k)} = X\right] < \infty.$$

Proof. To simplify the notation, we denote

$$X^+ = X^{(k+1)}, \quad X = X^{(k)}, \quad \alpha = \alpha^{(k)}, \quad \Sigma_h = \Sigma_h^{(k)}, \quad S = S^{(k)}. \tag{B.2}$$

From the update rule of SGN, it is straightforward to get

$$\begin{aligned}\mathbf{tr}\left(X^{+\top}X^+\right) = \left(\frac{1}{4}\alpha^2 - \alpha + 1\right)\mathbf{tr}(X^\top X) + \left(-\frac{1}{2}\alpha^2 + \alpha\right)\mathrm{tr}\left(X^\top\Sigma_h X(X^\top X)^{-1}\right)\\ + \alpha^2\mathbf{tr}\left(X^\top\Sigma_h^2 X(X^\top X)^{-2}\right) - \frac{3}{4}\alpha^2\mathbf{tr}\left((X^\top\Sigma_h X)^2(X^\top X)^{-3}\right).\end{aligned} \tag{B.3}$$

Here and below in this proof, we assume that $X^{(k)} = X$ is known. Suppose that

$$\mathbf{tr}(X^\top X) \leq 2p\mathbb{E}\|\Sigma_h\|, \tag{B.4}$$

then we have

$$\begin{aligned}\mathbb{E}\mathbf{tr}(X^{+\top}X^+) &\leq p\left(\frac{1}{2}\alpha^2 - 2\alpha + 2 - \frac{1}{2}\alpha^2 + \alpha\right)\mathbb{E}\|\Sigma_h\| + \alpha^2\mathbb{E}\|\Sigma_h\|^2\mathbf{tr}(X^\top X)^{-1}\\ &\leq^{(i)} p\left(-\alpha + 2\right)\mathbb{E}\|\Sigma_h\| + \frac{p}{2}\alpha\mathbb{E}\|\Sigma_h\| = p\left(2 - \frac{1}{2}\alpha\right)\varphi_2 < \infty,\end{aligned}$$

where the inequality (i) follows from (B.1). Given that $\mathbf{tr}(X^{(0)\top}X^{(0)}) = p$, the hypothesis (B.4) may not be available. Hence, we turn to the case of $\mathbf{tr}(X^\top X) \geq 2p\mathbb{E}\|\Sigma_h\|$, where we have

$$\begin{aligned}\mathbb{E}\mathbf{tr}\left(X^{+\top}X^+\right) &\leq \left(\frac{1}{4}\alpha^2 - \alpha + 1 - \frac{1}{4}\alpha^2 + \frac{1}{2}\alpha\right)\mathbf{tr}(X^\top X) + \alpha^2\mathbb{E}\|\Sigma_h\|^2\mathbf{tr}(X^\top X)^{-1}\\ &\leq \left(\frac{1}{4}\alpha^2 - \alpha + 1 - \frac{1}{4}\alpha^2 + \frac{1}{2}\alpha + \frac{1}{4}\alpha\right)\mathbf{tr}(X^\top X) \leq \mathbf{tr}(X^\top X).\end{aligned}$$

The above inequality shows that $\mathbb{E}\mathbf{tr}\left(X^{+\top}X^+\right)$ will keep going down, until (B.4) is satisfied. Even though (B.4) might never hold, the SGN iterates could always be bounded in expectation by the initial value due to the non-increasing property.

We then turn to the expectational increment

$$\begin{aligned}\mathbb{E}\mathbf{tr}\left(S^\top S\right) &\leq \mathbb{E}\mathbf{tr}\left((X^\top X)^{-1}X^\top\Sigma_h^2 X(X^\top X)^{-1}\right) + \frac{1}{4}\mathbf{tr}\left(X^\top X\right)\\ &\leq \mathbb{E}\|\Sigma_h\|^2\mathbf{tr}(X^\top X)^{-1} + \frac{1}{4}\mathbf{tr}\left(X^\top X\right) < \infty,\end{aligned}$$

which thus completes the proof. □

We then prove the expectational full-rankness of SGN with infinitesimal stepsize.

Lemma B.2. *Suppose that the assumptions [A1]-[A3] hold. Taking $\alpha^{(k)} = \alpha \to 0$ $(k = 0, 1 \cdots)$, the sequence $\{X^{(k)}\}$ generated by Algorithm 1 satisfy*

$$\mathbb{E}\left[\sigma_{min}(X^{(k+1)}) \;\middle|\; \sigma_{min}(X^{(k)}) = \sigma\right] \geq \sigma > 0,$$

where σ is some constant.

Proof. For simplicity, we use the same notations as (B.2). Recall that $P = X(X^\top X)^{-1}$. It is easy to get

$$X^{+\top}\left(\lambda_{\max}(PP^\top) - PP^\top\right)X^+ \succeq 0.$$

Further, we have $\sigma^2_{\min}(X^+)\sigma^2_{\max}(P) \geq \sigma^2_{\min}(P^\top X^+)$ and

$$\begin{aligned}
\sigma_{\min}(P^\top X^+) &= 1 + \frac{\alpha}{2}\lambda_{\min}\left((X^\top X)^{-1}X^\top(\Sigma_h X - X(X^\top X))(X^\top X)^{-1}\right) \\
&= 1 - \frac{\alpha}{4}\lambda_{\max}\left(P^\top \nabla \hat{f}^{(k)}(X)(X^\top X)^{-1}\right) \\
&\geq 1 - \frac{\alpha}{4}\frac{\|\nabla \hat{f}^{(k)}(X)\|}{\sigma^3_{\min}(X)}.
\end{aligned}$$

Invoke the assumption [A3] and Lemma B.1, $\mathbb{E}\|\nabla \hat{f}^{(k)}(X)\|$ is thus finite. Since $\sigma_{\max}(P) = \sigma^{-1}_{\min}(X)$, the following holds as $\alpha \to 0$

$$\sigma^2_{\min}(X^+) \geq \sigma^2_{\min}(X)\sigma^2_{\min}(P^\top X^+) \geq \sigma^2_{\min}(X) \geq \sigma > 0.$$

□

Then, we are in a position to prove Lemma 3.5.

Proof of Lemma 3.5. The infinitesimal mean of $\mathbf{vec}\,(X(t))$ could be directly obtained from (3.5) as

$$\begin{aligned}
&\lim_{\alpha\to 0}\frac{1}{\alpha}\mathbb{E}[\mathbf{vec}\,(\Delta X_\alpha)\,|X_\alpha = X] \\
= \;&\mathbf{vec}\left(\Sigma X(X^\top X)^{-1} - \frac{1}{2}X - \frac{1}{2}X(X^\top X)^{-1}X^\top \Sigma X(X^\top X)^{-1}\right),
\end{aligned}$$

and the infinitesimal variance from (3.6) as

$$\lim_{\alpha\to 0}\frac{1}{\alpha}\mathbb{E}\left[\mathbf{vec}\,(\Delta X_\alpha(t))\,\mathbf{vec}\,(\Delta X_\alpha(t))^\top\,|X_\alpha(t) = X\right] \leq 0 = 0 (= \mathcal{O}(\alpha)),$$

where the inequality follows from Lemma B.1. Applying Corollary 4.2 in Section 7.4 of [8] then completes the proof. □

C Proof of Lemma 3.7

Proof. As $X_\alpha(t)$ lies around some stationary point $\tilde{X}$ associated with the eigen-index set $\mathcal{I}_p = \{i_1, \cdots, i_p\}$ and the extended one $\hat{\mathcal{I}}_p$, we impose two additional assumptions on $\tilde{X}$ for simplicity:

(i) $i_1 > i_2 > \cdots > i_{p-1} > i_p$;

(ii) $\lambda_{i_1}, \lambda_{i_2}, \cdots, \lambda_{i_{p-1}}$ are single eigenvalues and λ_{i_p} has multiplicity $(q - p + 1) \geq 1$, where $q = |\hat{\mathcal{I}}_p|$. Then, $X_\alpha(t)$ could be expressed in the form

$$X_\alpha(t) = E_q \Lambda_q^{\frac{1}{2}} W V^\top + \mathcal{O}(\alpha^{\frac{1}{2}}),$$

where $E_q = [e_{i_1}, e_{i_2}, \cdots, e_{i_q}] \in \mathbb{R}^{n \times q}$, $\Lambda_q = \mathbf{Diag}(\lambda_{i_1}, \lambda_{i_2}, \cdots, \lambda_{i_q}) \in \mathbb{R}^{q \times q}$, V is a p-dimensional orthogonal matrix, and the weight matrix $W \in \mathbb{R}^{q \times p}$ is defined by

$$W = \left[\begin{array}{cc} I_{p-1} & 0_{(p-1)\times 1} \\ 0_{(q-p+1)\times(p-1)} & w \end{array} \right], \quad w^\top w = 1, \quad w \in \mathbb{R}^{q-p+1}.$$

Let $\Delta Y_\alpha(t) = Y_\alpha(t+\alpha) - Y_\alpha(t) = \alpha^{-\frac{1}{2}} \left(X_\alpha(t+\alpha) - X_\alpha(t) \right) V$. In the same vein as the proof of Lemma 3.5, we compute the (np)-dimensional infinitesimal mean as

$$\lim_{\alpha \to 0} \frac{1}{\alpha} \mathbb{E} \left[\mathbf{vec}(\Delta Y_\alpha(t)) | Y_\alpha(t) = Y \right] = \mathbf{vec} \left(\Sigma Y \Lambda_p^{-1} - Y \right), \tag{C.1}$$

and the (np)-dimensional infinitesimal variance matrix as

$$\lim_{\alpha \to 0} \frac{1}{\alpha} \mathbb{E} \left[\mathbf{vec}(\Delta Y_\alpha) \mathbf{vec}(\Delta Y_\alpha)^\top | Y_\alpha(t) = Y \right] = \mathbb{E} \left[\begin{array}{ccc} S_{(:,1)} S_{(:,1)}^\top & \cdots & S_{(:,1)} S_{(:,p)}^\top \\ \cdots & \cdots & \cdots \\ S_{(:,p)} S_{(:,1)}^\top & \cdots & S_{(:,p)} S_{(:,p)}^\top \end{array} \right],$$

where

$$\begin{aligned} S_{(:,j)} = \Sigma_h E_q \Lambda_q^{\frac{1}{2}} W \Lambda_{p(:,j)}^{-\frac{1}{2}} &- \frac{1}{2} E_q \Lambda_q^{\frac{1}{2}} W_{(:,j)} \\ &- \frac{1}{2} E_q \Lambda_q^{\frac{1}{2}} W \Lambda_p^{-1} W^\top \Lambda_q^{\frac{1}{2}} E_q^\top \Sigma_h E_q \Lambda_q^{\frac{1}{2}} W \Lambda_{p(:,j)}^{-1}, \qquad 1 \le j \le p, \end{aligned}$$

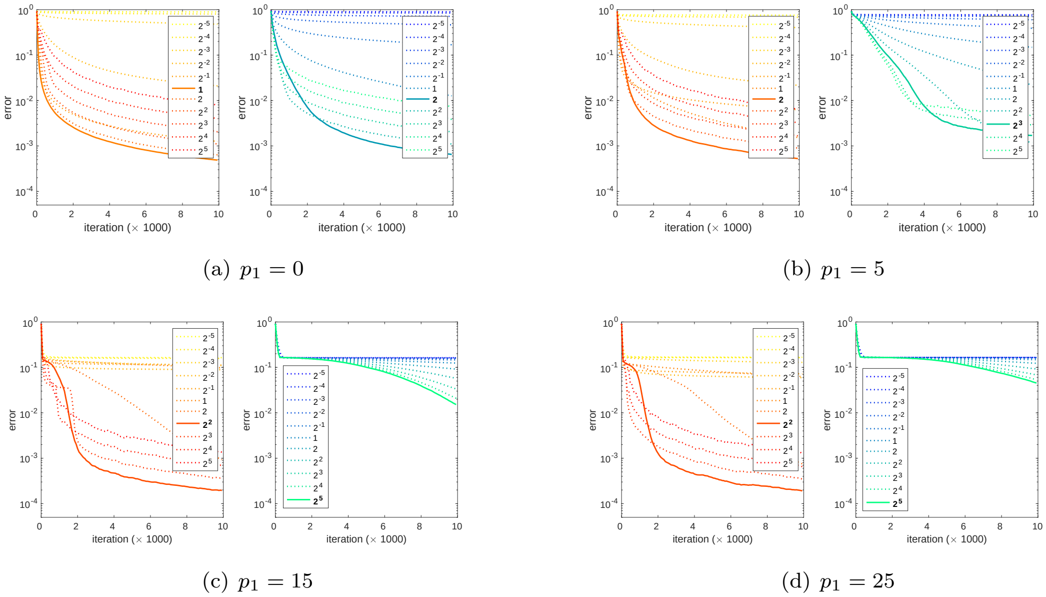
where $\Sigma_h = \sum_{i=1}^h a_i a_i^\top / h$ and $a_i \in \mathbb{R}^n$ $(i = 1, \cdots, h)$ are i.i.d. copies of the random vector a. For any $1 \le l,\ r \le p$, we have

$$\begin{aligned} \mathbb{E} \left[S_{(:,l)} S_{(:,r)}^\top \right] = &\left(-\frac{1}{2} - \frac{1}{2} + \frac{1}{4} + \frac{1}{4} + \frac{1}{4} \right) T_1^{lr} + \left(1 - \frac{1}{2} - \frac{1}{2} + \frac{1}{4} \right) \left(1 - \frac{1}{h} \right) T_1^{lr} \\ &+ \frac{1}{h} T_2^{lr} - \frac{1}{2h} \left(\tilde{W} T_2^{lr} + T_2^{lr} \tilde{W} \right) + \frac{1}{4h} \tilde{W} T_2^{lr} \tilde{W}, \end{aligned} \tag{C.2}$$

where

$$\begin{aligned} T_1^{lr} &= (\lambda_{i_l} \lambda_{i_r})^{\frac{1}{2}} (E_q W)_{(:,l)} \left((E_q W)_{(:,r)} \right)^\top, \\ T_2^{lr} &= \begin{cases} \Sigma + 2T_1^{lr} + \sum_{j=1}^\tau w_j^2 (\mathbb{E}[b_{i_p}^4]/\lambda_{i_p} - 3\lambda_{i_p}) E_{i_p+j-1, i_p+j-1}, & l = r = p, \\ \Sigma + (\mathbb{E}[b_{i_l}^4]/\lambda_{i_l}^2 - 1) T_1^{lr}, & l = r \neq p, \\ (\lambda_{i_l} \lambda_{i_r})^{\frac{1}{2}} \left(T_1^{lr} + T_1^{rl} \right), & otherwise, \end{cases} \\ \tilde{W} &= \left[\begin{array}{cc} W W^\top & 0_{q \times (n-q)} \\ 0_{(n-q) \times q} & 0_{(n-q) \times (n-q)} \end{array} \right] \in \mathbb{R}^{n \times n}, \end{aligned}$$

and $E_{l,r}$ denotes the n-dimensional matrix with its (l, r)-th element being one and other entries being zero. Using (C.1), (C.2) and applying Corollary 4.2 in Section 7.4 of [8] finally yield the SDE approximation (3.14). It could be easily verified that even if the additional assumptions (i), (ii) are eliminated, (3.14) is still valid. □



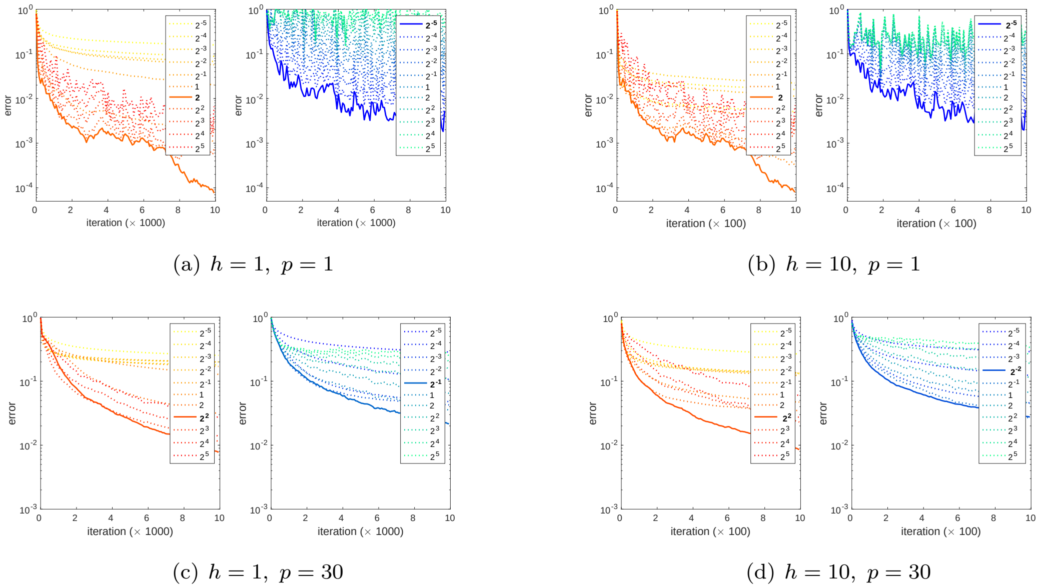
(a) $p_1 = 0$

(b) $p_1 = 5$

(c) $p_1 = 15$

(d) $p_1 = 25$

Figure 6: The estimation errors of Oja's iteration (right in each subfigure) and SGN (left in each subfigure) with diminishing stepsizes when varying γ from 2^{-5} to 2^5 on `Gau-gap-2` with $h = 1$.



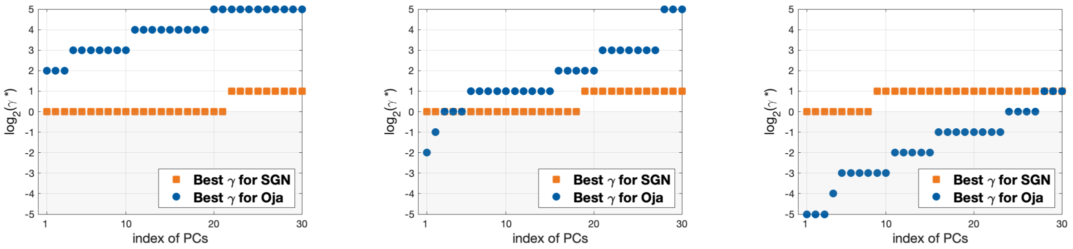
(a) $h = 1,\ p = 1$

(b) $h = 10,\ p = 1$

(c) $h = 1,\ p = 30$

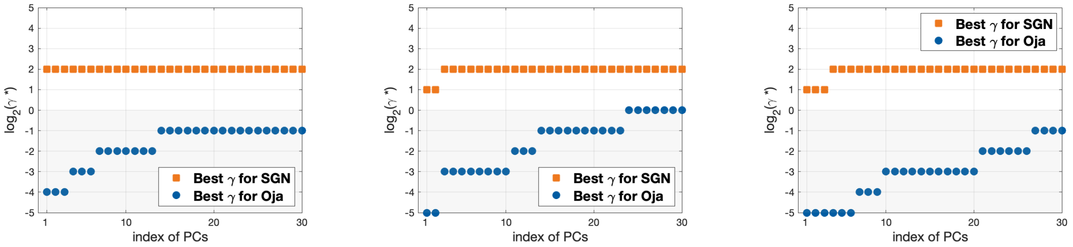
(d) $h = 10,\ p = 30$

Figure 7: The estimation errors of Oja's iteration (right in each subfigure) and SGN (left in each subfigure) with diminishing stepsizes when varying γ from 2^{-5} to 2^5 on `CIFAR-10`.



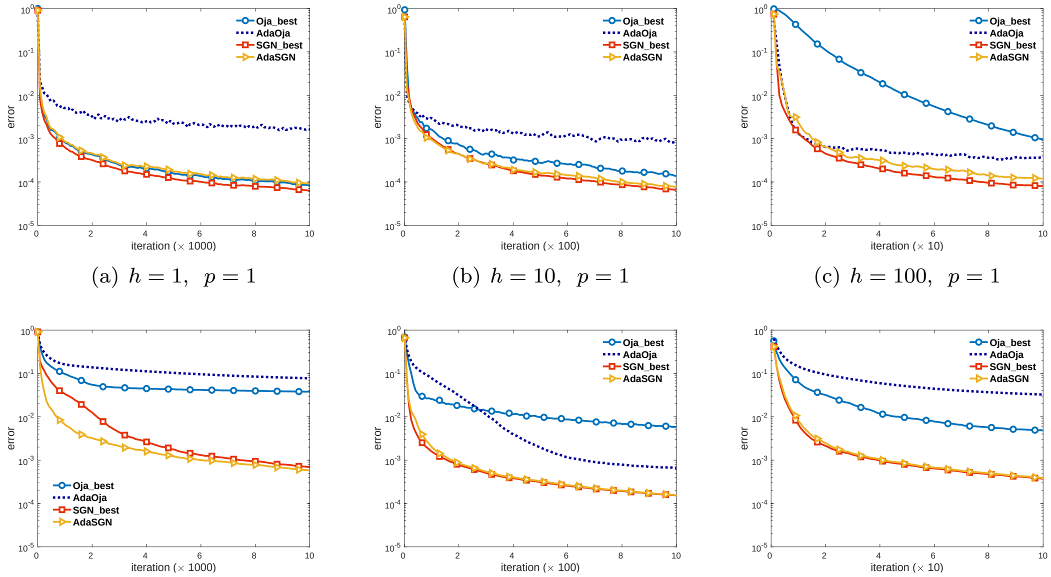
(a) $\bar{\mu} = 1$ (b) $\bar{\mu} = 10$ (c) $\bar{\mu} = 100$

Figure 8: The best $\gamma \in \{2^{-5}, \cdots, 2^5\}$ for Oja's iteration and SGN on `Gau-gap-1` with $h = 1$.



(a) `MNIST` (b) `Fashion-MNIST` (c) `CIFAR-10`

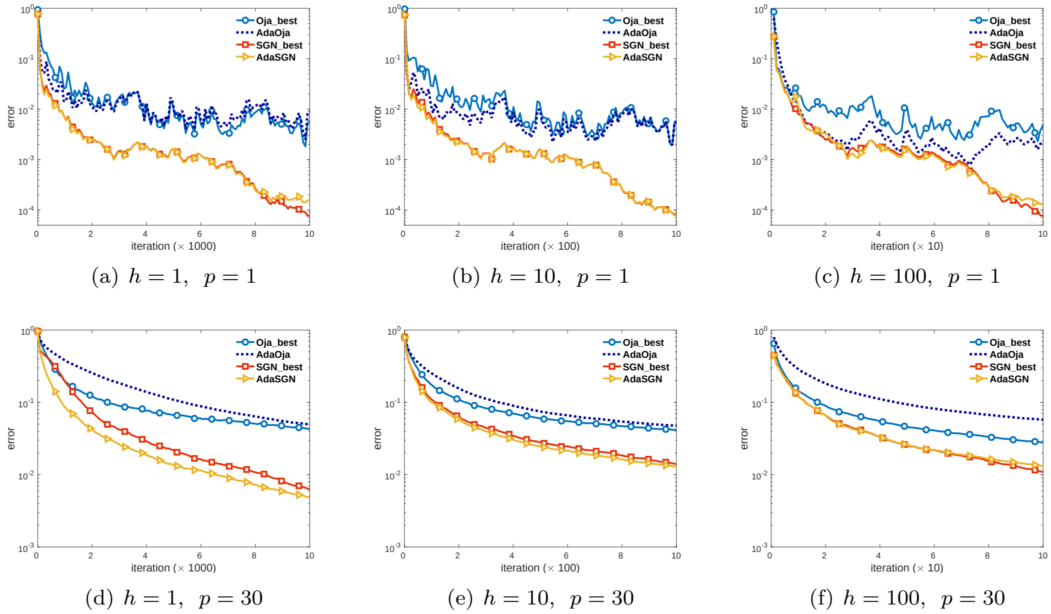
Figure 9: The best $\gamma \in \{2^{-5}, \cdots, 2^5\}$ for Oja's iteration and SGN on real datasets with $h = 1$.



(a) $h = 1, \ \ p = 1$ (b) $h = 10, \ \ p = 1$ (c) $h = 100, \ \ p = 1$

(d) $h = 1, \ \ p = 30$ (e) $h = 10, \ \ p = 30$ (f) $h = 100, \ \ p = 30$

Figure 10: The estimation error of Oja's iteration and SGN with best-tuned diminishing stepsizes, AdaOja and AdaSGN on `Gau-gap-1` with $\bar{\mu} = 10$.



(a) $h = 1, \;\; p = 1$

(b) $h = 10, \;\; p = 1$

(c) $h = 100, \;\; p = 1$

(d) $h = 1, \;\; p = 30$

(e) $h = 10, \;\; p = 30$

(f) $h = 100, \;\; p = 30$

Figure 11: The estimation error of Oja's iteration and SGN with best-tuned diminishing stepsizes, AdaOja and AdaSGN on `CIFAR`-10.