Stochastic Scale Invariant Power Iteration for KL-divergence Nonnegative Matrix Factorization

Cheolmin Kim[1], Youngseok Kim[2], and Diego Klabjan[1]

[1]Department of Industrial Engineering and Management Sciences, Northwestern University
[2]Department of Statistics, University of Chicago

Abstract

We introduce a mini-batch stochastic variance-reduced algorithm to solve finite-sum scale invariant problems, which cover several examples in machine learning and statistics such as PCA and estimation of mixture proportions. The proposed algorithm is a stochastic generalization of scale invariant power iteration, which specializes to power iteration when full-batch is used for the PCA problem. We provide a convergence analysis that shows the expectation of the optimality gap decreases at a linear rate under some conditions on the step size, epoch length, batch size and initial iterate. Numerical experiments on the KL-NMF problem using real and synthetic datasets demonstrate that the proposed stochastic approach not only converges faster than state-of-the-art deterministic algorithms but also produces excellent quality robust solutions.

1 Introduction

We consider a class of optimization problems called *scale invariant problems* Kim et al. [2019] of the form

$$\max \quad f(x) = \frac{1}{n}\sum_{i=1}^{n} f_i(x) \quad \text{subject to} \quad \partial\mathcal{B}_d \triangleq \{x \in \mathbb{R}^d \,|\, \|x\|_2 = 1\} \tag{1}$$

where f_i are scale invariant functions of the same type, i.e. f_i are either multiplicatively scale invariant with the multiplicative factor $u(c) = |c|^p$ such that $f_i(cx) = u(c)f_i(x)$ or additively scale invariant satisfying $f_i(cx) = f_i(x) + v(c)$ with the additive factor $v(c) = \log_a |c|$. The scale invariant problem covers interesting problems in machine learning and statistics such as L_p-norm kernel PCA Kim and Klabjan [2019] and estimation of mixture proportions Kim et al. [2018], to name a few. Moreover, as studied in Kim et al. [2019], more examples such as independent component analysis (ICA) [Hyvärinen et al., 2004, Hyvarinen, 1999], Gaussian mixture models (GMM), Kullback–Leibler divergence non-negative matrix factorization (KL-NMF) [Févotte and Idier, 2011, Lee and Seung, 2001, Wang and Zhang, 2013] and the Burer-Monteiro factorization of semi-definite programs [Erdogdu et al., 2018] can be formulated to extended settings of (1).

Assuming that f is twice differentiable on an open set containing $\partial\mathcal{B}_d$, the scale invariant problem (1) can be locally viewed as a leading eigenvector problem; a stationary point x^* is an eigenvector of $\nabla^2 f(x^*)$. Moreover, if the Lagrange multiplier λ^* such that $\lambda^* x^* = \nabla f(x^*)$ is greater than eigenvalues of $\nabla^2 f(x^*)(I - x^*(x^*)^T)$, then a stationary point x^* is a local maximum. Due to this eigenvector property, the scale invariant problem can be efficiently solved by a general form of power iteration called *scale invariant power iteration (SCI-PI)* Kim et al. [2019] specified by

$$x_{k+1} \leftarrow \nabla f(x_k)/\|\nabla f(x_k)\|_2. \tag{2}$$

The convergence behavior of (2) generalizes that of power iteration. If x_0 is initialized close to a local optimum x^*, the optimality gap $1-(x_k^T x^*)^2$ linearly converges at an asymptotic rate of $(\bar{\lambda}/\lambda^*)^2$ where $\bar{\lambda}$ is the largest absolute value of eigenvalues of $\nabla^2 f(x^*)(I - x^*(x^*)^T)$. This convergence rate specializes to $(\lambda_1/\lambda_2)^2$ in the case of the PCA problem Jolliffe [2002] where λ_1 and λ_2 are the first and the second eigenvalues of the covariance matrix $\frac{1}{n}\sum_{i=1}^{n} a_i a_i^T$ constructed by data vectors a_i. Thus, SCI-PI not only has a general form of power iteration but also extends its attractive local linear convergence property.

On the other hand, due to the analogy between power iteration for the leading eigenvector problem and gradient descent for convex optimization, many advanced algorithms have been developed for power iteration such as noisy Hardt and Price [2014], coordinate-wise [Lei et al., 2016], momentum [Xu et al., 2018], online [Boutsidis et al., 2015, Garber et al., 2015] and stochastic [Oja, 1982, Shamir, 2015, 2016, Xu et al., 2018, Kim and Klabjan, 2020] algorithms. In particular, built upon the stochastic variance-reduced gradient technique [Johnson and Zhang, 2013], stochastic variance-reduced power iterations Shamir [2015], Kim and Klabjan [2020] reduce the total runtime to obtain an ϵ-optimal solution to the PCA problem from $\mathcal{O}(dn(\frac{\lambda_1}{\lambda_1-\lambda_2})\log\frac{1}{\epsilon})$ to $\mathcal{O}(d(n+(\frac{\lambda_1}{\lambda_1-\lambda_2})^2)\log\frac{1}{\epsilon})$. This decoupling of the sample size n from the eigen-gap $1-\lambda_2/\lambda_1$ is significant in a large scale setting where n is large. These stochastic PCA algorithms opened up the possibility to solve non-convex constrained machine learning models using stochastic algorithms.

In this work, we develop a mini-batch stochastic algorithm to solve finite-sum scale invariant problems (1) and provide a convergence analysis for it. Our algorithm S-SCI-PI is a stochastic generalization of SCI-PI Kim et al. [2019]. The update formula of S-SCI-PI has a similar form to that of VR Power Kim and Klabjan [2020] but the scaling factor for a full-gradient is not simply the dot product of an inner iterate and an outer iterate but a homogeneous function of it. In the convergence analysis of S-SCI-PI, we derive a bound on the expectation of the optimality gap. This analysis is not trivial since two types of errors are involved. The first one is attributed to the difference of the Hessians between the iterate and the optimal solution. To control this error, we derive a condition on the step size, batch size and initial iterate, which ensures that the error is not increasing in the course of the algorithm. On the other hand, the second error occurs from the stochastic sampling of gradient. Using recursion, we develop a compact decomposition of the optimality gap in expectation. We show that the expected optimality gap converges at a linear rate under some conditions on the step size, epoch length, batch size and initial iterate.

Moreover, we introduce an application of S-SCI-PI to the KL-NMF problem. As shown in Kim et al. [2019], the KL-NMF subproblem is a scale invariant problem. Using the separable structure of the KL-NMF subproblems, we can reformulate the KL-NMF problem as a two-block scale invariant problem. We alternatively apply S-SCI-PI to optimize two non-negative matrices. Experiments on synthetic and real datasets reveal that the proposed stochastic approach not only converges faster than state-of-the-art deterministic algorithms but also produces robust solutions under random initialization.

Our work has the following contributions.

1. We propose the stochastic algorithm S-SCI-PI to solve finite-sum scale invariant problems. The algorithm adapts the stochastic variance-reduced gradient technique by adjusting the scaling factor of full-gradients depending on the order of scale invariance.

2. We provide a convergence analysis for S-SCI-PI. Deriving compact representations of error terms, we prove linear convergence of S-SCI-PI where we show that the expected optimality gap decreases at a linear rate under some conditions on the step size, epoch length, batch size and initial iterate.

3. We introduce an application to the KL-NMF problem where we present the stochastic algorithm for KL-NMF. Computational experiments show that our stochastic algorithm converges faster than state-of-the-art deterministic algorithms for the KL-NMF problem.

The paper is organized as follows. We present the algorithm in Section 2 and provide the convergence analysis in Section 3. We introduce the KL-NMF problem and its reformulation to a two-block scale invariant problem in Section 4. We discuss some implementation issues in Section 4.2. The experimental results on real and synthetic datasets are followed in Section 5.

2 Algorithm

Before presenting the algorithm, we first introduce some notations. For the scale invariant objective function f in (1), we let p be the degree of scale invariance. If f is multiplicatively scale invariant, p is the order of the multiplicative factor $u(c)=|c|^p$. On the other hand, for additively scale invariant function, let $p=0$. We denote the k-th coordinate of the gradient ∇f as $\nabla_k f(x)$. For a mini-batch sample $S\subset[n]\triangleq\{1,2,\cdots,n\}$, we define a stochastic function as $f_S=\sum_{l\in S}f_l/|S|$.

Similar to the stochastic variance-reduced gradient (SVRG) method Johnson and Zhang [2013], our algorithm has a two-loop structure. At the start of each inner-loop, we compute the full

gradient $\tilde{g}_T$ at the outer iterate $\tilde{x}_T$, and utilize this gradient information to construct a stochastic variance-reduced gradient g_t at the inner iterate x_t. In order to derive a stochastic variance-reduced gradient at x_t using the full gradient at $\tilde{x}_s$, we decompose x_t as

$$x_t = \frac{x_t^T \tilde{x}_T}{\|\tilde{x}_T\|^2}\tilde{x}_T + x_t - \frac{x_t^T \tilde{x}_T}{\|\tilde{x}_T\|^2}\tilde{x}_T.$$

In the above equation, the first component is the projection of x_t onto $\tilde{x}_T$ while the second part represents the orthogonal component of x_t with respect to $\tilde{x}_T$. Since ∇f is scale invariant with degree $p-1$ [Kim et al., 2019, Proposition 3], using $\tilde{g}_T$, we can compute the exact gradient at the first component as

$$\nabla f\left(\frac{x_t^T \tilde{x}_T}{\|\tilde{x}_T\|^2}\tilde{x}_T\right) = \frac{|x_t^T \tilde{x}_T|^{p-1}}{\|\tilde{x}_T\|^{2(p-1)}}\nabla f(\tilde{x}_T) = \alpha_t \tilde{g}_T, \quad \alpha_t = \frac{|x_t^T \tilde{x}_T|^{p-1}}{\|\tilde{x}_T\|^{2(p-1)}}. \tag{3}$$

To approximate the difference of gradients at x_t and $(x_t^T \tilde{x}_T)\tilde{x}_T/\|\tilde{x}_T\|^2$, we use a stochastic sample $S_t \subset [n]$ of size s, which results in a stochastic variance-reduced gradient g_t at x_t as

$$g_t = \alpha_t \tilde{g}_T + \tfrac{1}{s}\textstyle\sum_{l\in S_t}\left[\nabla f_l(x_t) - \alpha_t \nabla f_l(x_0)\right].$$

To control the progress of the algorithm depending on the variance of g_t, we introduce a step size $\eta \in (0, 1]$. Using the step size η, we derive the following update rule

$$x_{t+1} \leftarrow (1-\eta)x_t + \eta g_t/\|x_t\|^{p-2}.$$

Note that we divide g_t by $\|x_t\|^{p-2}$ to match its scale with x_t since $\nabla f(x) \propto \nabla^2 f(x)x$ and $\nabla^2 f(x)$ is scale invariant with degree $p-2$ [Kim et al., 2019, Proposition 3].

Summarizing all the above, we obtain Algorithm 1.

Algorithm 1 Stochastic SCI-PI (S-SCI-PI)

Parameter: step size $\eta \in (0, 1]$, batch size s, epoch length m
randomly initialize outer iterate $\tilde{x}_0 \in \partial\mathcal{B}_d$
for $T = 0, 1, \ldots$ **do**
  $x_0 \leftarrow \tilde{x}_T$, $\tilde{g}_T \leftarrow \nabla f(x_0)$
  for $t = 0, 1, \ldots, m-1$ **do**
    $\alpha_t \leftarrow |x_t^T x_0|^{p-1}/\|x_0\|^{2(p-1)}$
    sample $S_t \subset [n]$ of size s uniformly at random
    $g_t \leftarrow \alpha_t \tilde{g}_T + s^{-1}\sum_{l\in S_t}\left[\nabla f_l(x_t) - \alpha_t \nabla f_l(x_0)\right]$
    $x_{t+1} \leftarrow (1-\eta)x_t + \eta g_t/\|x_t\|^{p-2}$
  end for
  $\tilde{x}_{T+1} \leftarrow x_m$
end for

3 Convergence Analysis

For the analysis of the algorithm, we assume that every f_i is twice continuously differentiable on an open set containing $\mathcal{B}_{d,\infty} \triangleq \{y \in \mathbb{R}^d : \|y\|_\infty \le 1\}$. Let x^* be a local optimal solution satisfying $\nabla f(x^*) = \lambda^* x^*$, (λ_i, v_i) be an eigen-pair of $\nabla^2 f(x^*)$ and $\sigma = \|\nabla^2 f(x^*)\|$. Due to the eigenvector property of the scale invariant problem, x^* is an eigenvector of $\nabla^2 f(x^*)$. Without loss of generality, we let $x^* = v_1$. Since x^* is a local maximum, we have $\lambda^* > \overline{\lambda} = \max_{2\le i\le d}|\lambda_i|$ [Kim et al., 2019, Proposition 4].

Let H_i be the Hessian of $\nabla_i f$ and $F_i(y^1, \cdots, y^d) = (\lambda^* - \lambda_1)\mathbb{1}_{i=1} I + \sum_{j=1}^d v_{ij} H_j(y^j)$. Also, we let $G_S(y^1, \cdots, y^d)$ be the matrix such that $\nabla\nabla_j g_S(y^j)^T$ is the j^{th} row of $G_S(y^1, \cdots, y^d)$ where $g_S = f_S - f$.

Next, we introduce some constants that are used to derive bounds in the analysis. First, let M_1 and M_2 be constants such that

$$M_1 = \max_{\substack{x\in\mathcal{B}_d,\\ y^1,\cdots,y^d\in\mathcal{B}_{d,\infty}}} \sqrt{\textstyle\sum_{i=1}^d (x^T F_i(y^1, \cdots, y^d)x)^2}, \quad M_2 = \max_{\substack{x,z\in\mathcal{B}_d,\\ y^1,\cdots,y^d\in\mathcal{B}_{d,\infty}}} \left|\textstyle\sum_{i=1}^d z_i x^T H_i(y^1, \cdots, y^d)x\right| \tag{4}$$

and let $M = \max(M_1, M_2)$. These constants measure local smoothness of the objective function f near the local optimal solution x^*. Let B_s be the set of all mini-batch samples $S \subset [n]$ of size $s = |S|$. We define quantities K and L as

$$K = \max_{y^1,\cdots,y^d \in \mathcal{B}_\infty} E_{S\sim B_s}\left[\|G_S(y^1,\cdots,y^d)\|^2\right], \quad L = \max_{S\in B_s,\, y^1,\cdots,y^d \in \mathcal{B}_\infty} \|G_S(y^1,\cdots,y^d)\|^2 \tag{5}$$

and let L_0 be an upper bound of L which can be obtained by setting $s = 1$ (an easy calculation establish this). They measure deviation of f_S from its mean f with respect to stochastic sample S of size s. K measures the mean squared deviation (variance) of f_S and L is concerned with the maximum squared deviation of f_S from f. As the batch size s is increasing, both K and L are decreasing, and both of them become zero when $s = n$. While K decreases as a factor of $1/s$, L is a non-trivial function of s. Therefore, if some f_i is extremely irregular (i.e. $|f_i - f|$ has an extremely large value around the solution), we would have to use a batch size close to n to ensure that L is smaller than some level.

Next we present the convergence analysis for S-SCI-PI. We first analyze a single inner iteration which computes x_{t+1} from x_t. Let $\alpha(\eta) = 1 - \eta + \eta\lambda^*$, $\beta(\eta) = 1 - \eta + \eta\bar{\lambda}$, $y_k = x_k/\|x_k\|$ and $\Delta_t = 1 - y_t^T x^*$. Since the optimality gap is expressed as $\sum_{i=2}^d (x_t^T v_k)^2/(x_t^T v_1)^2$, it is important to analyze how $x_t^T v_k$ changes after each iteration. The following lemma provides an expression of $x_{t+1}^T v_k$ as a sum of three components.

Lemma 3.1. *For $1 \le k \le d$ and any t, if $x_t^T x_0 \ge 0$, then we have*

$$x_{t+1}^T v_k = \left(1 - \eta + \eta(\lambda_k + (\lambda^* - \lambda_1)\mathbb{1}_{k=1})\right) x_t^T v_k + \frac{1}{2}\eta\|x_t\|(y_t - x^*)^T F_k(\hat{y}_t^1,\cdots,\hat{y}_t^d)(y_t - x^*)$$
$$+\eta\left(G_{S_t}(\bar{y}_t^1,\cdots,\bar{y}_t^d)\left(x_t - (x_t^T y_0)y_0\right)\right)^T v_k$$

for some $\hat{y}^1,\cdots,\hat{y}^d, \bar{y}^1,\cdots,\bar{y}^d \in \mathcal{N}(y_t, x^*) \triangleq \left\{z \,|\, z_j = \mu_j x_j^* + (1-\mu_j)y_{tj}, \mu_j \in [0,1], j \in [d]\right\}$ where $[d] \triangleq \{1,\cdots,d\}$.

In Lemma 3.1, the first term represents the growth of $x_t^T v_k$. The multiplicative factor is $1-\eta+\eta\lambda^*$ if $k = 1$ and $1-\eta+\eta\lambda_k$ otherwise. The second component is attributed to the difference of the Hessians at x_t and x^*. As x_t closes on x^*, this term goes to zero. The last term is the stochastic error. The stochastic error is affected by the batch size s and how closely x_t is aligned with x_0 where we compute the full gradient. The following lemma provides a condition on η, L, M and x_0 to ensure that $y_t^T x^*$ is not smaller than $y_0^T x^*$ for every stochastic realization.

Lemma 3.2. *For any positive integer m, if the step size η, s and x_0 are chosen to satisfy*

$$\Delta_0 \le \min\left\{1 - \frac{1}{\sqrt{2}}, \frac{(\lambda^* - \bar{\lambda})^2}{4(M + 2\sqrt{L_0})^2}\right\} \tag{6}$$

and either one of the following condition holds:

$$L \le \frac{(\lambda^* - \bar{\lambda} - 2M\sqrt{\Delta_0})^2}{32}, \tag{7}$$

$$\eta \le 1/\max(1, \nu_1, \nu_2, \nu_3), \tag{8}$$

where

$$\nu_1 = 1 - \lambda^* + 2\theta_1 m\sqrt{2\Delta_0} \tag{9a}$$
$$\nu_2 = m\lambda^* + 1 - (m+1)(\bar{\lambda} + M\sqrt{\Delta_0}) \tag{9b}$$
$$\nu_3 = \frac{128L\theta_1\lambda^* m^2}{\theta_2^2\bar{\lambda}\Delta_0\sqrt{\Delta_0}} + 1 - \left(\bar{\lambda} + M\sqrt{\Delta_0}\right) \tag{9c}$$
$$\theta_1 = \lambda^* + \sigma + M\sqrt{\frac{\Delta_0}{2}} + 2\sqrt{L} \tag{10a}$$
$$\theta_2 = \lambda^* - \bar{\lambda} - 2\sqrt{\Delta_0}\left(M + 2\sqrt{L}\right), \tag{10b}$$

then we have $x_t^T x_0 \ge 0$ and $\Delta_t \le \Delta_0$ for all $0 \le t \le m$.

Note that $\lambda^* - \bar{\lambda}$ is a generalized eigen-gap at the solution and L and Δ_0 are decreasing functions of the batch size s and the dot product $y_0^T x^*$. Given that Δ_0 is moderately small, we can satisfy conditions (7) or (8) by increasing the batch size s or decreasing the step size η, respectively. Conditioning on x_t, the next lemma derives expectation bounds for several quantities involving $(x_{t+1}^T v_k)^2$ and norms.

Lemma 3.3. *For any positive integer m, if η, s and x_0 satisfy (6), (7) (or (8)) and*

$$\eta \le 1/\max(1, 1 - \lambda^* + \sqrt{2}M\Delta_0) \tag{11}$$

then we have

$$E[\|x_{t+1}\|^2|x_t] \leq \big[\big(\alpha(\eta)+\eta M\Delta_t\big)^2+\eta^2K\big]\|x_t\|^2,$$
$$E\big[\sum_{k=2}^{d}(x_{t+1}^Tv_k)^2|x_t\big] \leq \big(\beta(\eta)+\eta M\sqrt{\Delta_t}\big)^2\sum_{k=2}^{d}(x_t^Tv_k)^2+8\eta^2K\|x_t\|^2\sum_{k=2}^{d}(y_0^Tv_k)^2,$$
$$E[(x_{t+1}^Tv_1)^2|x_t] \geq \left[\alpha(\eta)-\frac{\eta M\Delta_t}{1-\Delta_t}\right]^2(x_t^Tv_1)^2$$

for any $0 \leq t \leq m$.

Using induction on the single iteration bound in Lemma 3.3, we derive an upper bound for $E[\sum_{k=2}^{d}(x_t^Tv_k)^2]$ and a lower bound $E[(x_t^Tv_1)^2]$ as functions of $E[\sum_{k=2}^{d}(x_0^Tv_k)^2]$ and $E[(x_0^Tv_1)^2]$ for a single outer iteration.

Lemma 3.4. *For any positive integer* m*, if* η*,* s *and* x_0 *satisfy* (6), (7) *(or* (8)*),* (11) *and*

$$\eta \leq \max 1/(1, 1-\lambda^*-M\sqrt{\Delta_0}+\sqrt{Km}), \tag{12}$$

then we have

$$E\big[\sum_{k=2}^{d}(x_t^Tv_k)^2\big] \leq E\big[\sum_{k=2}^{d}(x_0^Tv_k)^2\big]\big[\big(\beta(\eta)+\eta M\sqrt{\Delta_0}\big)^{2t}+16\eta^2Kt\big(\alpha(\eta)+\eta M\sqrt{\Delta_0}\big)^{2(t-1)}\big],$$
$$E[(x_t^Tv_1)^2] \geq \left[\alpha(\eta)-\frac{\eta M\Delta_0}{1-\Delta_0}\right]^{2t}E[(x_0^Tv_1)^2].$$

The inequalities in Lemma 3.4 are important since we can combined them to yield a bound on the optimality gap which is expressed as $\delta_t \triangleq E[\sum_{k=2}^{d}(x_t^Tv_k)^2]/E[(x_t^Tv_1)^2]$. In the next lemma, we show that under some conditions on η, m, s and x_0, the optimality gap decreases at least by $1-\rho$ after each outer iteration.

Lemma 3.5. *For any positive integer* m*, if* η*,* s *and* x_0 *satisfy* (6), (7) *(or* (8)*) and*

$$\eta \leq 1/\max(1, \nu_4, \nu_5) \tag{13}$$

where

$$\nu_4 = 1-\lambda^*-M\sqrt{\Delta_0}+\max\big(\sqrt{Km},\, 64K/\big(\lambda^*-\bar{\lambda}-2M\sqrt{\Delta_0}\big)\big) \tag{14}$$
$$\nu_5 = 1-\lambda^*+M\sqrt{\Delta_0}+\max\big(2m\big(\lambda^*-\bar{\lambda}-2M\sqrt{\Delta_0}\big),\, 4mM\sqrt{\Delta_0}/\log 2\big), \tag{15}$$

then we have $\delta_m \leq (1-\rho)\cdot\delta_0$ *where*

$$0 < \rho = \frac{\eta m\big(\lambda^*-\bar{\lambda}-2M\sqrt{\Delta_0}\big)}{2\big(1-\eta+\eta(\lambda^*-M\sqrt{\Delta_0})\big)} < 1. \tag{16}$$

Finally, we analyze the entire algorithm. Let $\tilde{\Delta}_0 = 1-\tilde{x}_0^Tx^*$, $\tilde{\delta}_s = E[\sum_{k=2}^{d}(\tilde{x}_s^Tv_k)^2]/E[(\tilde{x}_s^Tv_1)^2]$. By repeatedly applying Lemma 3.5, the following theorem states that $\tilde{\delta}_s$ decreases at a liner rate under some conditions on η, m, $|S|$ and $\tilde{x}_0$.

Theorem 3.6. *For any positive integer* m*, if* η*,* s *and* $\tilde{x}_0$ *satisfy* (6), (7) *(or* (8)*) and* (13) *with* $\Delta_0 = \tilde{\Delta}_0$*, then for any* $\epsilon > 0$*, after* $\tau = \lceil(1/\rho)\log(\tilde{\delta}_0/\epsilon)\rceil$ *epochs of S-SCI-PI (Algorithm 1), we have* $\tilde{\delta}_\tau \leq \epsilon$.

Theorem 3.6 states that for any epoch length m, if $\widetilde{x}_0$ is moderately close to x^* and the step size η and the batch size s satisfies certain conditions, the optimality gap vanishes at an exponential rate. If there are few irregular f_i and the cost of sampling is low, we can satisfy (7) by making L small. In this case, η can take a large value and we are able to obtain rapid convergence. On the other hand, if there are many irregular data samples and sampling is expensive, we may not satisfy (7). Nevertheless, we can always ensure linear convergence of Algorithm 1 by choosing a small enough step size η (conditions (8), (11), (12), (13)) as in Shamir [2015].

4 Application: KL-divergence NMF

Let $V \in \mathbb{R}_+^{N\times M}$ be a given non-negative matrix, which we want to compress into the product of $W \in \mathbb{R}_+^{N\times K}$ and $H \in \mathbb{R}_+^{K\times M}$. The KL-NMF problem is defined as

$$\min \sum_{i,j}\left[V_{ij}\log\frac{V_{ij}}{(WH)_{ij}} - V_{ij} + (WH)_{ij}\right] \quad \text{subject to} \quad W_{ik}\geq 0,\ H_{kj}\geq 0,\ \forall i,j,k.$$

The above objective function is called the (generalized) Kullback-Leibler divergence $D_{KL}(V\|WH)$. Let H_j be the j-th column of H. Note that the objective function $D_{KL}(V\|WH)$ is separable in $H_1,\cdots,H_M$ and thus

$$H_j^{\text{new}} = \arg\max \ \textstyle\sum_i [V_{ij}\log(WH_j)_i - (WH_j)_i] \quad \text{subject to} \quad H_j \geq 0 \tag{17}$$

serves as the j-th subproblem.

Lemma 4.1. *[Kim et al., 2019, Lemma 9] The j-th KL-NMF subproblem* (17) *is equivalent to:*

$$X_j^{\text{new}} = \arg\max \ \textstyle\sum_{i=1}^N V_{ij}\log(LX_j)_i \quad \text{subject to} \quad X_j \in \mathcal{S}^d \triangleq \{x : \sum_{k=1}^d x_i = 1,\ x\succeq 0\} \tag{18}$$

where $L_{ik} = W_{ik}/(\sum_{i'} W_{i'k})$. The original solution H_j^{new} can be recovered via

$$H_{kj}^{\text{new}} = \frac{\sum_i V_{ij}}{\sum_i W_{ik}} X_{jk}^{\text{new}}, \quad k\in[d]. \tag{19}$$

By re-parameterizing X_j by Y_j^2, we can convert (18) into a finite-sum scale invariant problem (1). Therefore, the KL divergence NMF subproblem for H, namely

$$H^{\text{new}} = \arg\max \ [V_{ij}\log(WH)_{ij} - (WH)_{ij}] \quad \text{subject to} \quad H\geq 0$$

can be solved by applying S-SCI-PI for each column of H.

Our "vanilla" S-SCI-PI updates H by sampling a mini-batch $S \leftarrow \text{sample}(N,s)$ and running

$$H_{kj}^{\text{new}} \leftarrow H_{kj}\left[(1-\eta) + \eta\sum_{i\in S}\frac{L_{ik}V_{ij}}{(LH)_{ij}}\right]^2 \ \forall k,\ j, \quad H^{\text{new}} \leftarrow \text{column-rescale}(H^{\text{new}}), \tag{20}$$

where $\text{sample}(N,s)$ is sampling s elements from $[N]$ with or without replacement, and $\text{column-rescale}(X)$ is rescaling the columns of X to have sum 1. The update for W is similar due to $D_{KL}(V\|WH) = D_{KL}(V^T\|H^TW^T)$.

Our final remark is that we solve the j-th KL-NMF problem for $j\in[M]$ simultaneously as a single optimization problem. Let $X = [X_1,\cdots,X_M]$ be the concatenation of the M column vectors $X_1,\cdots,X_M\in\mathbb{R}^K$. Lemma 4.1 states that in the exact alternating minimization algorithm, the update of H amounts to solving

$$\min \quad \textstyle\sum_{i=1}^{NM} \text{vec}(V)_i \log[(I_M\otimes L)\text{vec}(X)]_i \quad \text{subject to} \quad \text{vec}(X_j)\in\mathcal{S}^K, \quad j\in[M] \tag{21}$$

where $\text{vec}(X) = (X_{11},\cdots,X_{K1},\cdots,X_{1M},\cdots,X_{KM})$ is a vectorization of $X\in\mathbb{R}^{K\times M}$, $\text{vec}(V)\in\mathbb{R}^{KM}$ is defined similarly and $I_M\otimes L = \text{kron}(I_M,L)\in\mathbb{R}^{NM\times KM}$ is the Kronecker product of I_M and L.

This allows us to exploit fast matrix multiplication routines (i.e. efficient matrix computation library such as OpenBLAS or intel MKL) in solving the aggregated problem (21), instead of solving the j-th subproblem sequentially for $j\in[M]$.

4.1 Related Algorithms

Let $Z = WH$ henceforth. We omit the update of W since it can be derived similarly.

Multiplicative Update (MU/EM) [Lee and Seung, 2001]: MU updates all H_{kj}'s simultaneously by

$$H_{kj}^{\text{new}} = H_{kj}\frac{\sum_i W_{ik}V_{ij}/Z_{ij}}{\sum_i W_{ik}}$$

for all k and j. Let us emphasize that the MU update is identical to the standard EM algorithm for the estimation of mixture proportions.

Cyclic Coordinate Descent (CCD/SCD) [Hsieh and Dhillon, 2011, Muzzarelli et al., 2019]: For all j and k, CCD/CSD runs coordinate-wise updates of H

$$H_{kj}^{\text{new}} = \max\left\{0, H_{kj} - \frac{\sum_i W_{ik}(1 - V_{ij}/Z_{ij})}{\sum_i V_{ij} W_{ik}^2 / Z_{ij}^2}\right\}$$

sequentially in a pre-fixed cyclic order.

Projected Gradient Descent (PGD) [Lin, 2007]: Given element-wise step sizes α_{kj}'s, PGD updates all H_{kj}'s simulataneously via

$$H_{kj}^{\text{new}} = \max\left\{0, H_{kj} - \alpha_{kj} \sum_i W_{ik}\left(1 - \frac{V_{ij}}{Z_{ij}}\right)\right\}.$$

Note that Multiplicative Update (MU) is a special case of PGD when $\alpha_{kj} = H_{kj}/(\sum_i W_{ik})$, which does not require projection onto the non-negative orthant. Also, CCD updates H_{kj} one at a time with a coordinate-wise optimal step size $\alpha_{kj} = 1/\sum_i (V_i W_{ik}^2 / Z_{ij}^2)$. By contrast, PGD uses a single step size $\alpha_j = \alpha_{1j} = \cdots = \alpha_{Kj}$ for each column j for fast line searches.

Let us highlight that S-SCI-PI and all the comparison methods belong to the family of alternating minimization algorithms, which update H given W and then update W given H iteratively.

4.2 Practical Considerations

In this part, we compare several sampling schemes for the update of H. Since the sampling scheme for the update of W can be similarly discussed, we omit it.

Vector-wise Sampling: We construct $V_S \in \mathbb{R}_+^{s \times M}$ and $W_S \in \mathbb{R}_+^{s \times D}$ by sampling rows of $V \in \mathbb{R}_+^{N \times M}$ and $W \in \mathbb{R}_+^{N \times D}$ uniformly at random, respectively. The stochastic gradient reads

$$\nabla_S^{\text{row}} f(H) = \frac{n}{s} W_S^T [V_S \oslash (W_S H)].$$

For a dense data matrix V, we prefer to use this vector-wise (or row-wise) sampling scheme for the update of H, since it allows us to exploit fast matrix multiplication libraries.

Element-wise Sampling: We vectorize the problem by introducing the element-wise iterator $i = (i_1, i_2) \in [N] \times [M] = [NM]$. This yields

$$f(H) = \sum_{i \in \mathcal{I}} V_{i_1, i_2} \log \sum_{k=1}^{D} W_{i_1, k} H_{k, i_2}$$

where $\mathcal{I}$ is the subset of $[NM]$ such that $V_{i_1,i_2} \neq 0$ if and only if $i = (i_1, i_2) \in \mathcal{I}$. In other words, $\mathcal{I}$ is the index set of the nonzero elements in V.

We construct S by sampling s elements of $\mathcal{I}$ uniformly at random, and consider the stochastic gradient as

$$\nabla_S^{\text{elem}} f(H) = \frac{|\mathcal{I}|}{s} \sum_{i \in S} \sum_k \frac{W_{i_1,k} V_{i_1,i_2}}{\sum_{k'=1}^{D} W_{i_1,k'} H_{k',i_2}} E_{k,i_2}$$

where E_{k,i_2} is the standard basis matrix having 1 at (k, i_2)-th entry and 0 otherwise.

For a sparse data matrix V, we prefer this element-wise sampling scheme for H over the row-wise sampling scheme, since each column has a different sparsity pattern.

Numerical Issues: The KL-NMF objective function and its gradient are unstable when entries of V and WH are close to 0. As reported in Kim et al. [2019] and based on the experiments in Section 5, MU and the full-batch version of F-SCI-PI are numerically stable. On the other hand, S-SCI-PI has certain numerical issues since randomness of the stochastic gradient may lead entries of W and H arbitrary close to 0. Thus we have added safeguard to avoid this by rejecting the stochastic gradient when it produces a numeric error (i.e. when any of the elements of stochastic gradient is negative). In such a case we proceed to the next iteration.

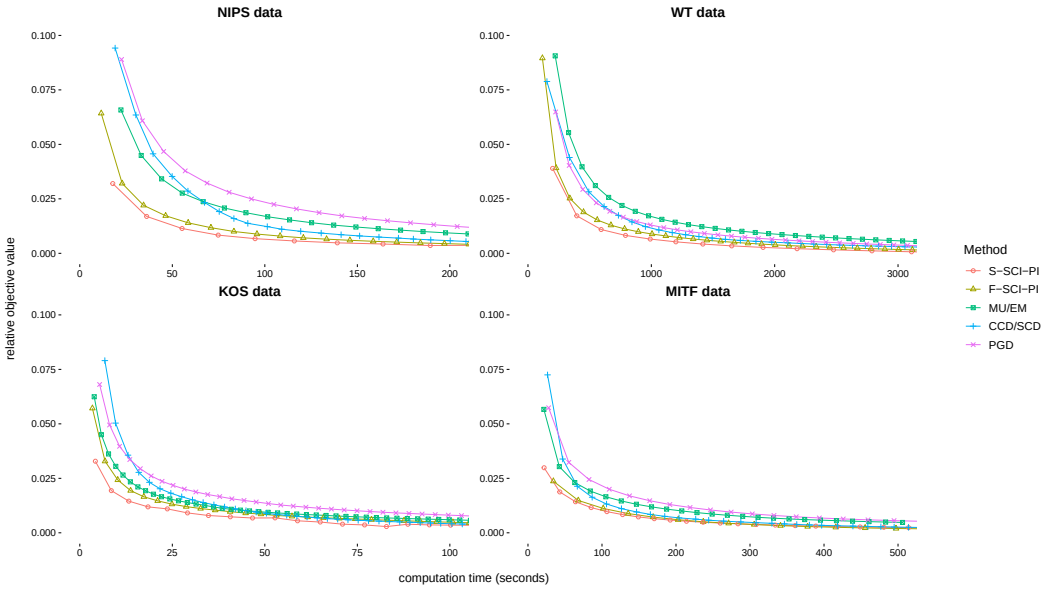

Figure 1: Convergence plots (relative error vs. computation time) of one-step alternating minimization algorithms on real data sets. A few initial data points are removed for increased visibility.

5 Experiment

We test the proposed algorithm S-SCI-PI on synthetic and real-world data sets. All experiments are implemented on a standard laptop (2.6 GHz Intel Core i7 processor and 16GB of RAM) using the C++ programming language. We use 4 real data sets publicly available online and 3 synthetic data sets generated from Poisson distributions. The description is provided in Appendix A. We set $K = 20$ features. All the reported values are averaged over 10 independent replicates started at different initial points, each of which is obtained by running 5 MU/EM steps on a Uniform(0,1) random matrix. For S-SCI-PI, we perform grid search on the parameters by selecting the best parameters among different batch proportions $s/n \in \{0.0001, 0.001, 0.01, 0.1, 1\}$, epoch lengths $m \in \{10, 100, 1000\}$ and step sizes $\eta \in \{0.01, 0.1, 1\}$.

KL-NMF (one-step alternating minimization): The one-step alternating minimization scheme is to update H via only a single iteration of each algorithm and then update W similarly. We compare S-SCI-PI, F-SCI-PI, MU/EM, CCD/SCD and PGD.

For dense data sets (WT, MITF), we apply vector-wise sampling only on the columns (of dimension 19,200 and 2,429, respectively) since the other dimension is small (287 and 361, respectively). For sparse data sets (NIPS, KOS), the element-wise sampling scheme is applied to both dimensions, which turns out to be more effective.

Figure 1 displays the relative errors with respect to the computation time for the 4 real data sets. Overall, S-SCI-PI with the chosen batch and epoch size improves the convergence over F-SCI-PI. However, S-SCI-PI does not outperform F-SCI-PI for the MITF data set, which has a relatively small number of columns (2,429). Also, both S-SCI-PI and F-SCI-PI exhibit much faster convergence than MU/EM. This clearly attests that S-SCI-PI is an excellent practical option for the KL-NMF problem.

KL-NMF subproblem (exact alternating minimization): The exact alternating minimization scheme is to update H until it reaches the exact coordinate minimizer and then update W similarly. Instead of solving the entire KL-NMF problem, we solve a single KL-NMF subproblem to optimality and plot relative objective values over time to compare the speed of convergence.

Figure 2 displays the results for the real world data sets. It shows that S-SCI-PI is an overall winner solving the KL divergence subproblems and hence an efficient method for exact alternating minimization. However, it does not outperform F-SCI-PI significantly on the sparse NIPS and KOS data set. As reported in Hsieh and Dhillon [2011], CCD/SCD is faster than MU/EM for the dense WT data set. However, our result on NIPS and KOS shows that CCD/SCD is much slower than

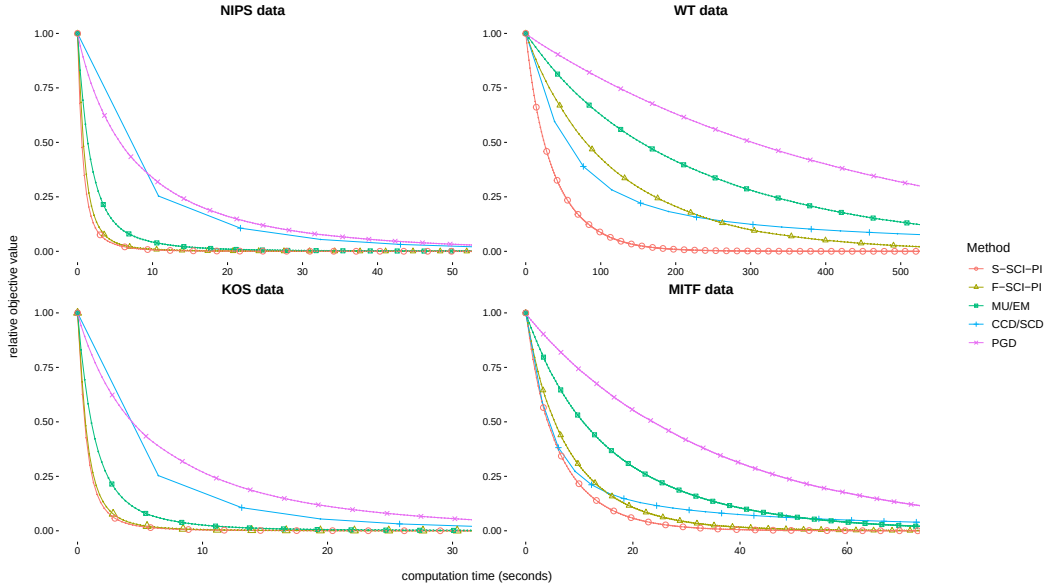

Figure 2: Convergence plots (relative error vs. computation time) for the KL-NMF subproblem.

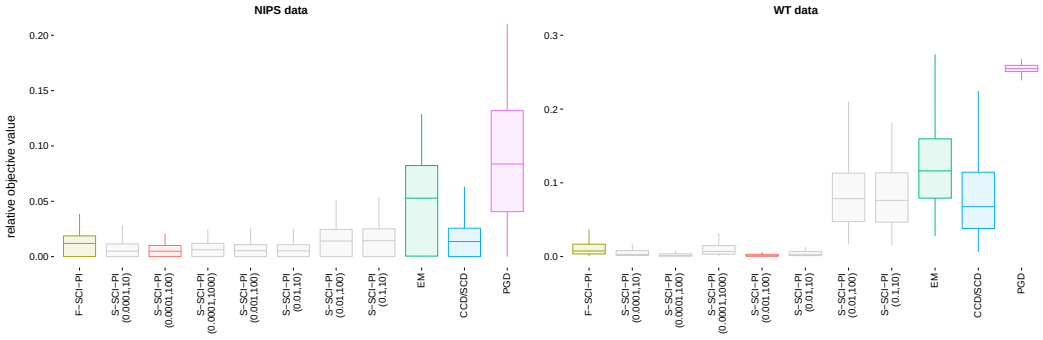

Figure 3: Boxplots of the relative errors after 30 seconds across 10 independent replicates

S-SCI-PI mainly due to the expensive coordinate updates.

Robustness of S-SCI-PI: Lastly, we compare the performance of S-SCI-PI for select choices of batch proportion s/n and epoch length m. For each choice of s/n and m, we find the best step size η using the same grid search as above. For the NIPS and WT data sets, we run the algorithms for 30 seconds with 10 independent replications and report boxplots of 10 relative objective values in Figure 3. In this figure, S-SCI-PI (n/s,m) stands for S-SCI-PI with batch proportion n/s and epoch length m. The boxplots show that the performance of S-SCI-PI is robust to batch proportion s/n and epoch length m given that step size η is appropriately selected. Again, we emphasize that S-SCI-PI has a remarkable improvement over the full gradient approach (F-SCI-PI) for the dense WT data set.

6 Final Remarks

We introduce a stochastic variance-reduced algorithm (S-SCI-PI) to solve finite-sum scale invariant problems for the first time in the literature and provide its convergence analysis. Our analysis shows that under some conditions on the step size, epoch length, batch size and initial iterate, the algorithm achieves linear convergence in expectation. Using S-SCI-PI, we introduce a stochastic approach to solve the KL-NMF problem. The experimental results reveal that S-SCI-PI exhibits robust and superior performance over state-of-the-art methods.

References

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. Society for Industrial and Applied Mathematics, 2015.

Murat A Erdogdu, Asuman Ozdaglar, Pablo A Parrilo, and Nuri Denizcan Vanli. Convergence Rate of Block-Coordinate Maximization Burer-Monteiro Method for Solving Large SDPs. *arXiv preprint arXiv:1807.04428*, 2018.

Cédric Févotte and Jérôme Idier. Algorithms for Nonnegative Matrix Factorization with the β-divergence. *Neural Computation*, 23(9):2421–2456, 2011.

Dan Garber, Elad Hazan, and Tengyu Ma. Online Learning of Eigenvectors. In *International Conference on Machine Learning*, pages 560–568, 2015.

Moritz Hardt and Eric Price. The noisy power method: A meta algorithm with applications. In *Advances in Neural Information Processing Systems*, pages 2861–2869, 2014.

Cho-Jui Hsieh and Inderjit S Dhillon. Fast coordinate descent methods with variable selection for non-negative matrix factorization. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1064–1072, 2011.

Aapo Hyvarinen. Fast ICA for Noisy Data using Gaussian Moments. In *Proceedings of the 1999 IEEE International Symposium on Circuits and Systems VLSI*, volume 5, pages 57–61. IEEE, 1999.

Aapo Hyvärinen, Juha Karhunen, and Erkki Oja. *Independent Component Analysis*, volume 46. John Wiley & Sons, 2004.

Rie Johnson and Tong Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In *Advances in Neural Information Processing Systems*, pages 315–323, 2013.

Ian Jolliffe. *Principal Component Analysis*. Wiley Online Library, 2002.

Cheolmin Kim and Diego Klabjan. A Simple and Fast Algorithm for L1-norm Kernel PCA. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.

Cheolmin Kim and Diego Klabjan. Stochastic Variance-Reduced Algorithms for PCA with Arbitrary Mini-Batch Sizes. In *International Conference on Artificial Intelligence and Statistics*, 2020.

Cheolmin Kim, Youngseok Kim, and Diego Klabjan. Scale invariant power iteration. *arXiv preprint arXiv:1905.09882*, 2019.

Youngseok Kim, Peter Carbonetto, Matthew Stephens, and Mihai Anitescu. A Fast Algorithm for Maximum Likelihood Estimation of Mixture Proportions Using Sequential Quadratic Programming. *arXiv preprint arXiv:1806.01412*, 2018.

Daniel D Lee and H Sebastian Seung. Algorithms for Non-negative Matrix Factorization. In *Advances in Neural Information Processing Systems*, pages 556–562, 2001.

Qi Lei, Kai Zhong, and Inderjit S Dhillon. Coordinate-wise Power method. In *Advances in Neural Information Processing Systems*, pages 2064–2072, 2016.

Chih-Jen Lin. Projected Gradient Methods for Non-negative Matrix Factorization. *Neural Computation*, 19(10):2756–2779, 2007.

Laura Muzzarelli, Susanne Weis, Simon B Eickhoff, and Kaustubh R Patil. Rank selection in non-negative matrix factorization: systematic comparison and a new mad metric. In *2019 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2019.

Erkki Oja. Simplified Neuron Model as a Principal Component Analyzer. *Journal of Mathematical Biology*, 15(3):267–273, 1982.

Ohad Shamir. A Stochastic PCA and SVD Algorithm with an Exponential Convergence Rate. In *International Conference on Machine Learning*, pages 144–152, 2015.

Ohad Shamir. Fast Stochastic Algorithms for SVD and PCA: Convergence Properties and Convexity. In *International Conference on Machine Learning*, pages 248–256, 2016.

Yu-Xiong Wang and Yu-Jin Zhang. Nonnegative Matrix Factorization: A Comprehensive Review. *IEEE Transactions on Knowledge and Data Engineering*, 25(6):1336–1353, 2013.

Peng Xu, Bryan He, Christopher De Sa, Ioannis Mitliagkas, and Chris Re. Accelerated Stochastic Power Iteration. In *International Conference on Artificial Intelligence and Statistics*, pages 58–67, 2018.

A Description of Data Sets

Table 1: Summary of data sets used for KL-NMF

Type	Name	# of samples	# of features	# of nonzeros	Sparsity
Synthetic	Pois1	1,000	1,000	900,000	0.90
Synthetic	Pois2	3,000	3,000	900,000	0.10
Synthetic	Pois3	9,000	9,000	900,000	0.01
Real	NIPS	1,500	12,419	280,000	0.985
Real	WT	287	19,200	5,510,000	0.000
Real	KOS	3,430	6,906	950,000	0.960
Real	MITF	361	2,429	877,000	0.000

The 4 real data sets in the table are retrieved from `https://archive.ics.uci.edu/ml/datasets/bag+of+words`, `https://www.microsoft.com/en-us/research/project` and `https://cbcl.mit.edu/cbcl`. They have already been used in the previous papers such as Hsieh and Dhillon [2011], Kim et al. [2019]. We preprocess the real data sets by removing few rows and columns having sums less than 20 for NIPS and KOS data sets.

For synthetic data, $V \in \mathbb{R}^{N\times M}$ generated from i.i.d. Poisson random variables, i.e. $V_{ij} \sim \text{Poisson}(-\log(1-\rho))$. Here ρ denotes sparsity or proportion of nonzero entries of V. This corresponds to the null signal case since in this case KL-NMF is the maximum likelihood estimation problem when $WH = 0$.

B Proofs

In what follows, we frequently use the fact that for $0 < \eta \leq 1$, $\eta \leq 1/\max(1,\nu)$ implies

$$n\nu \leq 1. \tag{22}$$

Using $\Delta_0 \leq 1 - 1/\sqrt{2}$ which follows from (6), we often use

$$\frac{\sqrt{\Delta_0}}{1-\Delta_0} \leq 1, \quad \frac{1}{1-\Delta_0} \leq \sqrt{2}. \tag{23}$$

Proof of Lemma 3.1. From the update rule in Algorithm 1, we have

$$\begin{aligned} x_{t+1} &= (1-\eta)x_t + \frac{\eta}{\|x_t\|^{p-2}}\left(\nabla f_{S_t}(x_t) - \alpha_t \nabla f_{S_t}(y_0) + \alpha_t \tilde{g}\right) \\ &= (1-\eta)x_t + \frac{\eta}{\|x_t\|^{p-2}}\nabla f(x_t) \\ &\quad + \frac{\eta}{\|x_t\|^{p-2}}\left[\nabla f_{S_t}(x_t) - \nabla f(x_t) - \alpha_t\left(\nabla f_{S_t}(y_0) - \nabla f(y_0)\right)\right]. \end{aligned} \tag{24}$$

Since $\nabla_i f$ is twice continuously differentiable on an open set containing $\partial\mathcal{B}_d$, using the Taylor theorem, we obtain

$$\nabla_i f(y_t) = \nabla_i f(x^*) + \nabla\nabla_i f(x^*)(y_t - x^*) + \frac{1}{2}\left(y_t - x^*\right)^T H_i(\hat{y}_t^i)\left(y_t - x^*\right) \tag{25}$$

where $\hat{y}_t^i \in \mathcal{N}(y_t, x^*)$. Since f is scale invariant with the degree of p, by [Kim et al., 2019, Proposition 3], ∇f is scale invariant with the degree of $p-1$, leading to

$$\frac{\nabla f(x_t)^T z}{\|x_t\|^{p-1}} = \nabla f(x^*)^T z + (y_t - x^*)^T \nabla^2 f(x^*) z + \frac{1}{2}(y_t - x^*)^T \sum_{i=1}^{d} z_i H_i(\hat{y}_t^i)(y_t - x^*) \tag{26}$$

for any vector $z \in \mathbb{R}^d$. For $k = 1$, using $v_1 = x^*$, we have

$$\begin{aligned} &\nabla f(x^*)^T v_1 = \nabla f(x^*)^T x^* = \lambda^*, \\ &(y_t - x^*)^T \nabla^2 f(x^*) v_1 = (y_t - x^*)^T \nabla^2 f(x^*) x^* = \lambda_1(y_t^T x^* - 1), \end{aligned}$$

which from (26) with $z = v_1$ results in

$$\begin{aligned}\frac{\nabla f(x_t)^T v_1}{\|x_t\|^{p-1}} &= \lambda^* - \lambda_1(1 - y_t^T x^*) + \frac{1}{2}(y_t - x^*)^T \sum_{i=1}^d v_{1i} H_i(\hat{y}_t^i)(y_t - x^*) \\ &= \lambda^* y_t^T x^* + (\lambda^* - \lambda_1)(1 - y_t^T x^*) + \frac{1}{2}(y_t - x^*)^T \sum_{i=1}^d v_{1i} H_i(\hat{y}_t^i)(y_t - x^*) \\ &= \lambda^* y_t^T x^* + \frac{1}{2}(y_t - x^*)^T \big[(\lambda^* - \lambda_1) I + \sum_{i=1}^d v_{1i} H_i(\hat{y}_t^i)\big](y_t - x^*) \\ &= \lambda^* y_t^T x^* + \frac{1}{2}(y_t - x^*)^T F_1(\hat{y}_t^1, \cdots, \hat{y}_t^d)(y_t - x^*)).\end{aligned} \tag{27}$$

For $2 \le k \le d$, from (26) with $z = v_k$, $(x^*)^T v_k = v_1^T v_k = 0$ and $\nabla f(x^*)^T v_k = \lambda^* v_1^T v_k = 0$, we have

$$\frac{\nabla f(x_t)^T v_k}{\|x_t\|^{p-1}} = \lambda_k y_t^T x^* + \frac{1}{2}(y_t - x^*)^T F_k(\hat{y}_t^1, \cdots, \hat{y}_t^d)(y_t - x^*). \tag{28}$$

Since ∇f_l is scale invariant with the degree of $p-1$ for each $l \in [n]$, we have

$$\nabla f_l(x_t) = \|x_t\|^{p-1} \nabla f_l(y_t), \quad \alpha_t \nabla f_l(y_0) = \|x_t\|^{p-1} (y_t^T y_0)^{p-1} \nabla f_l(y_0),$$

which leads to

$$\frac{1}{\|x_t\|^{p-1}} \left(\nabla f_{S_t}(x_t) - \nabla f(x_t) - \alpha_t \left(\nabla f_{S_t}(y_0) - \nabla f(y_0)\right)\right) = \nabla g_{S_t}(y_t) - \nabla g_{S_t}\big((y_t^T y_0) y_0\big).$$

Using the Taylor approximation of $\nabla_k g_{S_t}$ around $(y_t^T y_0) y_0$, we have

$$\nabla_k g_{S_t}(y_t) - \nabla_k g_{S_t}((y_t^T y_0) y_0) = \nabla \nabla_k g_{S_t}\big(\bar{y}_t^k\big)^T \big(y_t - (y_t^T y_0) y_0\big)$$

where $\bar{y}_t^k \in \mathcal{N}(y_t, (y_t^T y_0) y_0)$. This leads to

$$\frac{1}{\|x_t\|^{p-2}} \left(\nabla f_{S_t}(x_t) - \nabla f(x_t) - \alpha_t \left(\nabla f_{S_t}(y_0) - \nabla f(y_0)\right)\right) = G_{S_t}(\bar{y}_t^1, \cdots, \bar{y}_t^d)\big(x_t - (x_t^T y_0) y_0\big). \tag{29}$$

Using (24), (27), (28) and (29), we have

$$\begin{aligned}x_{t+1}^T v_k = &\left(1 - \eta + \eta(\lambda_k + (\lambda^* - \lambda_1)\mathbb{1}_{k=1})\right) x_t^T v_k + \frac{1}{2}\eta \|x_t\| (y_t - x^*)^T F_k(\hat{y}_t^1, \cdots, \hat{y}_t^d)(y_t - x^*) \\ &+ \eta \left(G_{S_t}(\bar{y}_t^1, \cdots, \bar{y}_t^d)\big(x_t - (x_t^T y_0) y_0\big)\right)^T v_k.\end{aligned} \tag{30}$$

□

Proof of Lemma 3.2. We prove by induction. Suppose that we have $\Delta_s \le \Delta_0$ for $s \le t < m$. Since $\Delta_0 \le 1 - 1/\sqrt{2}$, this implies that $y_t^T x^* \ge 1/\sqrt{2}$ and $y_0^T x^* \ge 1/\sqrt{2}$. Therefore, we have

$$\begin{aligned}y_t^T y_0 &= \left[(y_t^T x^*) x^* + y_t - (y_t^T x^*) x^*\right]^T \left[(y_0^T x^*) x^* + y_0 - (y_0^T x^*) x^*\right] \\ &= (y_t^T x^*)(y_0^T x^*) + (y_t - (y_t^T x^*) x^*)^T (y_0 - (y_0^T x^*) x^*) \\ &\ge (y_t^T x^*)(y_0^T x^*) - \|y_t - (y_t^T x^*) x^*\| \|y_0 - (y_0^T x^*) x^*\| \\ &\ge (y_t^T x^*)(y_0^T x^*) - \sqrt{1 - (y_t^T x^*)^2}\sqrt{1 - (y_0^T x^*)^2} \\ &\ge 0,\end{aligned}$$

which leads to

$$\|x_t - (x_t^T y_0) y_0\|^2 = \|x_t\|^2 (1 - (y_k^T y_0)^2) \le 2\|x_t\|^2 (1 - y_t^T y_0) = \|x_t\|^2 \|y_t - y_0\|^2.$$

By the triangular inequality, $(a+b)^2 \le 2(a^2 + b^2)$ and $\Delta_t \le \Delta_0$, we have

$$\|y_t - y_0\|^2 \le 2(\|y_t - x^*\|^2 + \|y_0 - x^*\|^2) \le 4\|y_0 - x^*\|^2.$$

From $y_0^T x^* \geq 0$, we further obtain

$$\|x_t - (x_t^T y_0)y_0\|^2 \leq 4\|x_t\|^2\|y_0 - x^*\|^2 = 8\|x_t\|^2(1 - y_0^T x^*) \tag{31}$$

$$\leq 8\|x_t\|^2(1 - (y_0^T x^*)^2) = 8\|x_t\|^2 \sum_{k=2}^{d}(y_0^T v_k)^2. \tag{32}$$

Using Lemma 3.1, the definitions of M and L, (31) and that $\Delta_t \leq \Delta_0$, we have

$$\begin{aligned} x_{t+1}^T v_1 &\geq (1 - \eta + \eta\lambda^*)\, x_t^T v_1 - \frac{1}{2}\eta M\|x_t\|\|y_t - x^*\|^2 - \eta\sqrt{L}\|x_t - (x_t^T y_0)y_0\| \\ &\geq (1 - \eta + \eta\lambda^*)\, x_t^T v_1 - \eta M(1 - y_t^T x^*)\|x_t\| - \eta\sqrt{8L(1 - y_0^T x^*)}\|x_t\| \\ &\geq \left[1 - \eta + \eta\left(\lambda^* - \frac{M\Delta_0}{1 - \Delta_0} - \frac{\sqrt{8L\Delta_0}}{1 - \Delta_0}\right)\right] y_0^T x^*\|x_t\|. \end{aligned} \tag{33}$$

By (23), (6) and that $L \leq L_0$, we have

$$\lambda^* - \frac{M\Delta_0}{1 - \Delta_0} - \frac{\sqrt{8L\Delta_0}}{1 - \Delta_0} \geq \lambda^* - \big(M + 4\sqrt{L}\big)\sqrt{\Delta_0} \geq \lambda^* - \frac{(\lambda^* - \bar{\lambda})\big(M + 4\sqrt{L}\big)}{2M + 4\sqrt{L_0}} \geq 0.$$

This leads to $x_{t+1}^T v_1 \geq 0$.

Now, we prove that $\Delta_{t+1} \leq \Delta_0$. Since $\{v_1, \ldots, v_d\}$ forms an orthogonal basis, we have $\|x_t\|^2 = \sum_{k=1}^{d}(x_t^T v_k)^2$. Since

$$\sum_{k=2}^{d}(1 - \eta + \eta\lambda_k)^2(x_t^T v_k)^2 \leq (1 - \eta + \eta\bar{\lambda})^2 \sum_{k=2}^{d}(x_t^T v_k)^2 \tag{34}$$

$$\sum_{k=1}^{d}\left(1 - \eta + \eta(\lambda_k + (\lambda^* - \lambda_1)\mathbb{1}_{k=1})\right)^2 (x_t^T v_k)^2 \leq (1 - \eta + \eta\lambda^*)^2\|x_t\|^2 \tag{35}$$

$$\begin{aligned} \sum_{k=2}^{d}\left[(y_t - x^*)^T F_k(\hat{y}_t^1, \cdots, \hat{y}_t^d)(y_t - x^*)\right]^2 &\leq \sum_{k=1}^{d}\left[(y_t - x^*)^T F_k(\hat{y}_t^1, \cdots, \hat{y}_t^d)(y_t - x^*)\right]^2 \\ &\leq M^2\|y_t - x^*\|^4 \end{aligned} \tag{36}$$

$$\begin{aligned} \sum_{k=2}^{d}\left[v_k^T G_{S_t}(\bar{y}_t^1, \cdots, \bar{y}_t^d)\big(x_t - (x_t^T y_0)y_0\big)\right]^2 &\leq \sum_{k=1}^{d}\left[v_k^T G_{S_t}(\bar{y}_t^1, \cdots, \bar{y}_t^d)\big(x_t - (x_t^T y_0)y_0\big)\right]^2 \\ &\leq L\|x_t - (x_t^T y_0)y_0\|^2 \end{aligned} \tag{37}$$

where (37) follows from $\|\sum_{k=1}^{d} v_k v_k^T\| = 1$. By Lemma 3.1 and the Cauchy-Schwarz inequality, we have

$$\sum_{k=2}^{d}(x_{t+1}^T v_k)^2 \leq \Big[(1 - \eta + \eta\bar{\lambda})\sqrt{\sum_{k=2}^{d}(x_t^T v_k)^2} + \frac{1}{2}\eta M\|x_t\|\|y_t - x^*\|^2 + \eta\sqrt{L}\|x_t - (x_t^T y_0)y_0\|\Big]^2 \tag{38}$$

$$\|x_{t+1}\|^2 = \sum_{k=1}^{d}(x_{t+1}^T v_k)^2 \leq \Big[1 - \eta + \eta\lambda^* + \frac{1}{2}\eta M\|y_0 - x^*\|^2 + \eta\sqrt{L}\|y_t - (y_t^T y_0)y_0\|\Big]^2\|x_t\|^2. \tag{39}$$

First, we consider the case when (7) holds. From $\Delta_t \leq \Delta_0 \leq 1$, we have $0 \leq y_t^T x^* \leq 1$ and $\sum_{k=2}^{d}(y_t^T v_k)^2 = 1 - (y_t^T x^*)^2 \leq 1 - (y_0^T x^*)^2 = \sum_{k=2}^{d}(y_0^T v_k)^2$, resulting in

$$\|y_t - x^*\|^2 \leq 2\sqrt{1 - y_t^T x^*}\sqrt{1 - (y_t^T x^*)^2} \leq 2\sqrt{\Delta_t}\sqrt{\sum_{k=2}^{d}(y_0^T v_k)^2} \leq 2\sqrt{\Delta_0}\sqrt{\sum_{k=2}^{d}(y_0^T v_k)^2}. \tag{40}$$

Plugging (32) and (40) into (38), we have

$$\sum_{k=2}^{d}(x_{t+1}^T v_k)^2 \leq \left[1 - \eta + \eta\left(\bar{\lambda} + M\sqrt{\Delta_0} + 2\sqrt{2L}\right)\right]^2\|x_t\|^2 \sum_{k=2}^{d}(y_0^T v_k)^2. \tag{41}$$

Combining (33) and (41), we have

$$\frac{\sum_{k=2}^d (x_{t+1}^T v_k)^2}{(x_{t+1}^T v_1)^2} \leq \left[\frac{1-\eta+\eta\big(\bar{\lambda}+M\sqrt{\Delta_0}+2\sqrt{2L}\big)}{1-\eta+\eta\left(\lambda^*-M\Delta_0/(1-\Delta_0)-2\sqrt{2L\Delta_0}/(1-\Delta_0)\right)}\right]^2 \frac{\sum_{k=2}^d (y_0^T v_k)^2}{(y_0^T v_1)^2}. \quad (42)$$

Using (23) and (7), we have

$$\lambda^* - \frac{M\Delta_0}{1-\Delta_0} - \frac{2\sqrt{2L\Delta_0}}{1-\Delta_0} - \big(\bar{\lambda}+M\sqrt{\Delta_0}+2\sqrt{2L}\big) \geq (\lambda^*-\bar{\lambda}) - 2M\sqrt{\Delta_0} - 4\sqrt{2L} \geq 0.$$

Therefore, from (42), we finally have

$$\frac{1-(y_{t+1}^T x^*)^2}{(y_{t+1}^T x^*)^2} = \frac{\sum_{k=2}^d (y_{t+1}^T v_k)^2}{(y_{t+1}^T v_1)^2} = \frac{\sum_{k=2}^d (x_{t+1}^T v_k)^2}{(x_{t+1}^T v_1)^2} \leq \frac{\sum_{k=2}^d (y_0^T v_k)^2}{(y_0^T v_1)^2} = \frac{1-(y_0^T x^*)^2}{(y_0^T x^*)^2},$$

which leads to $\Delta_{t+1} = 1 - y_{t+1}^T x^* \leq 1 - y_0^T x^* = \Delta_0$.

Next, we derive $\Delta_{t+1} \leq \Delta_0$ from (8). From (31) and (39), we have

$$\|x_{t+1}\|^2 \leq \Big[1-\eta+\eta\Big(\lambda^* + \frac{1}{2}M\|y_0-x^*\|^2 + 2\sqrt{L}\|y_0-x^*\|\Big)\Big]^2 \|x_t\|^2.$$

Using induction, this leads to

$$\|x_{t+1}\|^2 \leq \Big[1-\eta+\eta\Big(\lambda^* + \frac{1}{2}M\|y_0-x^*\|^2 + 2\sqrt{L}\|y_0-x^*\|\Big)\Big]^{2(t+1)} \|x_0\|^2. \quad (43)$$

On the other hand, from (24), (29), (31) and the definition of L, we have

$$\begin{aligned} x_{t+1}^T y_0 &= (1-\eta)x_t^T y_0 + \frac{\eta \nabla f(x_t)^T y_0}{\|x_t\|^{p-2}} + \eta y_0^T G_{S_t}(\bar{y}_t^1, \cdots, \bar{y}_t^d)(x_t - (x_t^T y_0) y_0) \\ &\geq (1-\eta)x_t^T y_0 + \frac{\eta \nabla f(x_t)^T y_0}{\|x_t\|^{p-2}} - 2\eta\sqrt{L}\|y_0-x^*\|\|x_t\|. \end{aligned}$$

Using $z = y_0$ in (26) and using $\nabla f(x^*) = \lambda^* x^*$ and the definition of M, we have

$$\begin{aligned} \frac{\nabla f(x_t)^T y_0}{\|x_t\|^{p-1}} &= \nabla f(x^*)^T y_0 + (y_t - x^*)^T \nabla^2 f(x^*) y_0 + \frac{1}{2}(y_t-x^*)^T \sum_{i=1}^d y_{0i} H_i(\hat{y}_t^i)(y_t - x^*) \\ &= \lambda^* y_t^T y_0 + (y_t - x^*)^T \left(\nabla^2 f(x^*) - \lambda^* I\right) y_0 - \frac{1}{2}M\|y_t - x^*\|^2 \\ &\geq \lambda^* y_t^T y_0 - (\lambda^* + \sigma)\|y_t - x^*\| - \frac{1}{2}M\|y_t - x^*\|^2 \\ &\geq \lambda^* y_t^T y_0 - (\lambda^* + \sigma)\|y_0 - x^*\| - \frac{1}{2}M\|y_0 - x^*\|^2 \end{aligned}$$

where the last inequality follows from

$$\|y_t - x^*\|^2 = 2(1 - y_t^T x^*) \leq 2(1 - y_0^T x^*) = \|y_0 - x^*\|^2.$$

This results in

$$\begin{aligned} x_{t+1}^T y_0 &\geq (1-\eta+\eta\lambda^*)x_t^T y_0 - \eta\big(\lambda^* + \sigma + \frac{1}{2}M\|y_0-x^*\| + 2\sqrt{L}\big)\|y_0-x^*\|\|x_t\| \\ &= (1-\eta+\eta\lambda^*)x_t^T y_0 - \eta\theta_1\sqrt{2\Delta_0}\|x_t\|. \end{aligned}$$

Combining with (43), we obtain

$$\begin{aligned} x_{t+1}^T y_0 &\geq (1-\eta+\eta\lambda^*)x_t^T y_0 \\ &\quad - \eta\theta_1\sqrt{2\Delta_0}\Big[1-\eta+\eta\Big(\lambda^* + \frac{1}{2}M\|y_0-x^*\|^2 + 2\sqrt{L}\|y_0-x^*\|\Big)\Big]^t \|x_0\| \\ &\geq (1-\eta+\eta\lambda^*)x_t^T y_0 - \eta\theta_1\sqrt{2\Delta_0}\Big[1-\eta+\eta\lambda^* + \eta\theta_1\sqrt{2\Delta_0}\Big]^t \|x_0\|. \end{aligned}$$

By recursion, we further have

$$x_{t+1}^T y_0 \geq (1-\eta+\eta\lambda^*)^{t+1}\|x_0\| \tag{44}$$
$$- \eta\theta_1\sqrt{2\Delta_0}\sum_{i=1}^{t+1}\left(1-\eta+\eta\lambda^*\right)^{i-1}\left[1-\eta+\eta\lambda^*+\eta\theta_1\sqrt{2\Delta_0}\right]^{t+1-i}\|x_0\|$$
$$= \left[2(1-\eta+\eta\lambda^*)^{t+1} - \left(1-\eta+\eta\lambda^*+\eta\theta_1\sqrt{2\Delta_0}\right)^{t+1}\right]\|x_0\|. \tag{45}$$

Also, by the definition of ν_1 and requirement (8) that $\eta\nu_1 \leq 1$ which yields

$$\bar{x} = \frac{\eta\theta_1\sqrt{2\Delta_0}}{1-\eta+\eta\lambda^*} \leq \frac{1}{2m},$$

it is easy to establish that

$$(1+x)^t \leq \exp(xt) \leq 2xt+1 \tag{46}$$

for any $0 \leq x \leq 1/2t$. Since $t < m$, by considering $x = \bar{x}$, we obtain from (45) inequality

$$x_{t+1}^T y_0 \geq (1-\eta+\eta\lambda^*)^{t+1}(1-2m\bar{x}) \geq 0.$$

Since $\|x_t - (x_t^T y_0)y_0\|^2 = \|x_t\|^2 - (x_t^T y_0)^2$, using (43), (45) and elementary algebraic manipulations, we have

$$\|x_t - (x_t^T y_0)y_0\|^2 \leq 4(1-\eta+\eta\lambda^*)^{2t}\left[\left(1+\frac{\eta\theta_1\sqrt{2\Delta_0}}{1-\eta+\eta\lambda^*}\right)^t - 1\right]\|x_0\|^2.$$

By (8), (9a) and (22), we have $\eta(1-\lambda^*+2\theta_1 m\sqrt{2\Delta_0}) \leq 1$ or

$$\frac{\eta\theta_1 m\sqrt{2\Delta_0}}{1-\eta+\eta\lambda^*} \leq \frac{1}{2}.$$

Since

$$\frac{\eta\theta_1 t\sqrt{2\Delta_0}}{1-\eta+\eta\lambda^*} \leq \frac{\eta\theta_1 m\sqrt{2\Delta_0}}{1-\eta+\eta\lambda^*} \leq \frac{1}{2} < 1,$$

using (46), we obtain

$$\|x_t - (x_t^T y_0)y_0\|^2 \leq 8\eta\theta_1(1-\eta+\eta\lambda^*)^{2t-1}\sqrt{2\Delta_0}t\|x_0\|^2. \tag{47}$$

Plugging (40) and (47) into the square root of (38) and then apply recursion, we have

$$\begin{aligned}
\sqrt{\sum_{k=2}^d (x_{t+1}^T v_k)^2} &\leq \left[1-\eta+\eta(\bar{\lambda}+M\sqrt{\Delta_0})\right]\sqrt{\sum_{k=2}^d (x_t^T v_k)^2} \\
&\quad + \eta\sqrt{\frac{8\eta L\theta_1\sqrt{2\Delta_0}}{1-\eta+\eta\lambda^*}}(1-\eta+\eta\lambda^*)^t t\|x_0\| \\
&\leq \left[1-\eta+\eta(\bar{\lambda}+M\sqrt{\Delta_0})\right]^{t+1}\sqrt{\sum_{k=2}^d (x_0^T v_k)^2} \\
&\quad + \eta\sqrt{\frac{8\eta L\theta_1\sqrt{2\Delta_0}}{1-\eta+\eta\lambda^*}}\sum_{i=1}^t i\,(1-\eta+\eta\lambda^*)^i\left[1-\eta+\eta(\bar{\lambda}+M\sqrt{\Delta_0})\right]^{t-i}\|x_0\|.
\end{aligned} \tag{48}$$

For a positive integer t and a non-negative real number $r \geq 0$ such that $rt \leq 1$, we have

$$(1+r)^t - 1 = r\left((1+r)^{t-1} + (1+r)^{t-2} + \cdots + 1\right) \geq rt$$

and (46) with $x = r$, which results in

$$\begin{aligned}
\sum_{i=1}^t (1+r)^i i &= \frac{1+r}{r^2}\left(t(1+r)^{t+1} - (t+1)(1+r)^t + 1\right) \\
&\leq \frac{1+r}{r^2}\left(t(1+r)^{t+1} - t(1+r)^t - rt\right) \\
&= \frac{(1+r)t}{r}\left((1+r)^t - 1\right) \\
&\leq 2(1+r)t^2.
\end{aligned} \tag{49}$$

By (8), (9b) and (22), we have

$$\frac{1-\eta+\eta\lambda^*}{1-\eta+\eta(\bar{\lambda}+M\sqrt{\Delta_0})}-1\leq\frac{1}{m}.$$

Also, by (6), we have $\lambda^*-\bar{\lambda}-M\sqrt{\Delta_0}\geq 0$, leading to

$$\frac{1-\eta+\eta\lambda^*}{1-\eta+\eta(\bar{\lambda}+M\sqrt{\Delta_0})}-1=\frac{\eta(\lambda^*-\bar{\lambda}-M\sqrt{\Delta_0})}{1-\eta+\eta(\bar{\lambda}+M\sqrt{\Delta_0})}\geq 0.$$

Therefore, using (49), we have

$$\begin{aligned}
&\sum_{i=1}^{t} i\,(1-\eta+\eta\lambda^*)^i\big[1-\eta+\eta(\bar{\lambda}+M\sqrt{\Delta_0})\big]^{t-i}\\
&=\Big[1-\eta+\eta(\bar{\lambda}+M\sqrt{\Delta_0})\Big]^t\sum_{i=1}^{t} i\,\left[\frac{1-\eta+\eta\lambda^*}{1-\eta+\eta(\bar{\lambda}+M\sqrt{\Delta_0})}\right]^i\\
&\leq 2(1-\eta+\eta\lambda^*)t^2\big[1-\eta+\eta(\bar{\lambda}+M\sqrt{\Delta_0})\big]^{t-1}.
\end{aligned}\tag{50}$$

Plugging (50) into (48), we obtain

$$\begin{aligned}
\sqrt{\sum_{k=2}^{d}(x_{t+1}^T v_k)^2}\leq\big[1-\eta+\eta(\bar{\lambda}+M\sqrt{\Delta_0})\big]^{t+1}\sqrt{\sum_{k=2}^{d}(x_0^T v_k)^2}\\
+2\eta\sqrt{8(1-\eta+\eta\lambda^*)\eta L\theta_1\sqrt{2\Delta_0}}\,t^2\big[1-\eta+\eta(\bar{\lambda}+M\sqrt{\Delta_0})\big]^{t-1}\|x_0\|.
\end{aligned}\tag{51}$$

On the other hand, from (33) and

$$\begin{aligned}
(1-y_t^T v_1)\|x_t\|&=\frac{1-y_t^T v_1}{y_t^T v_1}x_t^T v_1\leq\frac{1-y_0^T v_1}{y_0^T v_1}x_t^T v_1=\frac{\Delta_0}{1-\Delta_0}x_t^T v_1,\\
\sqrt{1-y_0^T v_1}\|x_t\|&=\frac{\sqrt{1-y_0^T v_1}}{y_t^T v_1}x_t^T v_1\leq\frac{\sqrt{1-y_0^T v_1}}{y_0^T v_1}x_t^T v_1=\frac{\sqrt{1-\Delta_0}}{1-\Delta_0}x_t^T v_1,
\end{aligned}$$

we have

$$x_{t+1}^T v_1\geq\left[1-\eta+\eta\left(\lambda^*-\frac{M\Delta_0}{1-\Delta_0}-\frac{2\sqrt{2L\Delta_0}}{1-\Delta_0}\right)\right]^{t+1}x_0^T v_1.\tag{52}$$

Combining (51) and (52), we have

$$\begin{aligned}
\frac{\sqrt{\sum_{k=2}^{d}(x_{t+1}^T v_k)^2}}{x_{t+1}^T v_1}\leq&\left[\frac{1-\eta+\eta(\bar{\lambda}+M\sqrt{\Delta_0})}{1-\eta+\eta\left[\lambda^*-(M\Delta_0+2\sqrt{2L\Delta_0})/(1-\Delta_0)\right]}\right]^{t+1}\frac{\sqrt{\sum_{k=2}^{d}(x_0^T v_k)^2}}{x_0^T v_1}\\
&+\frac{2\eta t^2\sqrt{8(1-\eta+\eta\lambda^*)\eta L\theta_1\sqrt{2\Delta_0}}\left[1-\eta+\eta(\bar{\lambda}+M\sqrt{\Delta_0})\right]^{t-1}}{\left(1-\eta+\eta\left[\lambda^*-(M\Delta_0+2\sqrt{2L\Delta_0})/(1-\Delta_0)\right]\right)^{t+1}y_0^T v_1}.
\end{aligned}\tag{53}$$

Since $0<\eta\leq 1$ and $\bar{\lambda}<\lambda^*$, we have

$$\frac{\bar{\lambda}}{\lambda^*}\leq\frac{1-\eta+\eta\bar{\lambda}}{1-\eta+\eta\lambda^*}\leq\frac{1-\eta+\eta(\bar{\lambda}+M\sqrt{\Delta_0})}{1-\eta+\eta\lambda^*}.\tag{54}$$

Let

$$\gamma=\frac{\lambda^*-\bar{\lambda}-M\sqrt{\Delta_0}-(M\Delta_0+2\sqrt{2L\Delta_0})/(1-\Delta_0)}{1-\eta+\eta\left[\lambda^*-(M\Delta_0+2\sqrt{2L\Delta_0})/(1-\Delta_0)\right]}.\tag{55}$$

By (23) and $\theta_2\geq 0$ due to (6), we have

$$\begin{aligned}
\frac{1}{1-\eta+\eta(\bar{\lambda}+M\sqrt{\Delta_0})}&=\frac{\gamma}{1-\eta\gamma}\left[\frac{1}{\lambda^*-\bar{\lambda}-M\sqrt{\Delta_0}-(M\Delta_0+2\sqrt{2L\Delta_0})/(1-\Delta_0)}\right]\\
&\leq\frac{\gamma}{\theta_2(1-\eta\gamma)}.
\end{aligned}\tag{56}$$

Using (54), (56) and that $y_0^T v_1 \geq 1/\sqrt{2}$, we have

$$\frac{2\eta t^2\sqrt{8(1-\eta+\eta\lambda^*)\eta L\theta_1\sqrt{2\Delta_0}}}{y_0^T v_1(1-\eta+\eta(\bar{\lambda}+M\sqrt{\Delta_0}))^2} \leq 8\sqrt{2}\sqrt{\frac{\lambda^*}{\bar{\lambda}}}\sqrt{\frac{\eta L\theta_1\sqrt{\Delta_0}}{1-\eta+\eta(\bar{\lambda}+M\sqrt{\Delta_0})}}\frac{\eta\gamma t^2}{\theta_2(1-\eta\gamma)}.$$

By (8), (9c) and (22), we have

$$\eta\left(\frac{128L\theta_1\lambda^* m^2}{\theta_2^2\bar{\lambda}\Delta_0\sqrt{\Delta_0}} + 1 - \left(\bar{\lambda}+M\sqrt{\Delta_0}\right)\right) \leq 1$$

or

$$\frac{\eta L\theta_1\sqrt{\Delta_0}}{1-\eta+\eta\left(\bar{\lambda}+M\sqrt{\Delta_0}\right)} \leq \frac{\theta_2^2\bar{\lambda}\Delta_0^2}{128\lambda^* m^2},$$

which results in

$$\frac{2\eta t^2\sqrt{8(1-\eta+\eta\lambda^*)\eta L\theta_1\sqrt{2\Delta_0}}}{y_0^T v_1(1-\eta+\eta(\bar{\lambda}+M\sqrt{\Delta_0}))^2} \leq \frac{\eta\gamma t^2\Delta_0}{(1-\eta\gamma)m} \leq \frac{\eta\gamma t^2}{(1-\eta\gamma)m}\frac{\sum_{k=2}^d (x_0^T v_k)^2}{(x_0^T v_1)^2}. \tag{57}$$

The last inequality follows from

$$\Delta_0 = 1 - y_0^T x^* \leq 1 - (y_0^T x^*)^2 \leq \frac{\sum_{k=2}^d (y_0^T v_k)^2}{(y_0^T v_1)^2} = \frac{\sum_{k=2}^d (x_0^T v_k)^2}{(x_0^T v_1)^2}.$$

Plugging (55) and (57) into (53), we have

$$\frac{\sqrt{\sum_{k=2}^d (x_{t+1}^T v_k)^2}}{x_{t+1}^T v_1} \leq (1-\eta\gamma)^{t+1}\left[1+\frac{\eta\gamma t^2}{(1-\eta\gamma)m}\right]\frac{\sqrt{\sum_{k=2}^d (x_0^T v_k)^2}}{x_0^T v_1}.$$

Using $1+nx \leq (1+x)^n$ for $x \geq 0$ and the fact that $\gamma \geq 0$ by (6), we have

$$\begin{aligned}(1-\eta\gamma)^{t+1}\left[1+\frac{\eta\gamma t^2}{(1-\eta\gamma)m}\right] &= 1 - \left[\left(1+\frac{\eta\gamma}{1-\eta\gamma}\right)^{t+1} - 1 - \frac{\eta\gamma t^2}{(1-\eta\gamma)m}\right](1-\eta\gamma)^{t+1} \\ &\leq 1 - \left(t+1-\frac{t^2}{m}\right)\eta\gamma(1-\eta\gamma)^t,\end{aligned}$$

which yields

$$\frac{\sqrt{\sum_{k=2}^d (x_{t+1}^T v_k)^2}}{x_{t+1}^T v_1} \leq \frac{\sqrt{\sum_{k=2}^d (x_0^T v_k)^2}}{x_0^T v_1}$$

due to $t < m$. We obtain

$$\frac{1-(y_{t+1}^T x^*)^2}{(y_{t+1}^T x^*)^2} = \frac{\sum_{k=2}^d (x_{t+1}^T v_k)^2}{(x_{t+1}^T v_1)^2} \leq \frac{\sum_{k=2}^d (x_0^T v_k)^2}{(x_0^T v_1)^2} = \frac{1-(y_0^T x^*)^2}{(y_0^T x^*)^2}$$

and we finally have $\Delta_{t+1} = 1 - y_{t+1}^T x^* \leq 1 - y_0^T x^* = \Delta_0$. □

Proof of Lemma 3.3. By Lemma 3.1, we have

$$\begin{aligned}x_{t+1}^T v_k = (1-\eta+\eta(\lambda_k+(\lambda^*-\lambda_1)\mathbb{1}_{k=1}))\, x_t^T v_k + \frac{1}{2}\eta\|x_t\|(y_t-x^*)^T F_k(\hat{y}_t^1,\cdots,\hat{y}_t^d)(y_t-x^*) \\ + \eta\big(G_{S_t}(\bar{y}_t^1,\cdots,\bar{y}_t^d)(x_t-(x_t^T y_0)y_0)\big)^T v_k.\end{aligned}$$

Since S_t is sampled uniformly at random, we have $E[f_{S_t}(y)] = f(y)$ for all $y \in \mathbb{R}^d$, which leads to

$$E[G_{S_t}(\bar{y}_t^1,\cdots,\bar{y}_t^d)] = E[E[G_{S_t}(\bar{y}_t^1,\cdots,\bar{y}_t^d)|\bar{y}_t^1,\bar{y}_t^2,\cdots,\bar{y}_t^d]] = 0.$$

Therefore,

$$E[(x_{t+1}^T v_1)^2 \,|\, x_t] = \big[(1-\eta+\eta\lambda^*)x_t^T v_1 + \frac{1}{2}\eta\|x_t\|(y_t-x^*)^T F_1(\hat{y}_t^1,\cdots,\hat{y}_t^d)(y_t-x^*)\big]^2 \\ + \eta^2(x_t-(x_t^T y_0)y_0)^T E[G_{S_t}(\bar{y}_t^1,\cdots,\bar{y}_t^d)^T v_1 v_1^T G_{S_t}(\bar{y}_t^1,\cdots,\bar{y}_t^d)](x_t-(x_t^T y_0)y_0). \tag{58}$$

In the same way, for $2 \le k \le d$, we have

$$E[(x_{t+1}^T v_k)^2 \,|\, x_t] = \big[(1-\eta+\eta\lambda_k)x_t^T v_k + \frac{1}{2}\eta\|x_t\|(y_t-x^*)^T F_k(\hat{y}_t^1,\cdots,\hat{y}_t^d)(y_t-x^*)\big]^2 \\ + \eta^2(x_t-(x_t^T y_0)y_0)^T E[G_{S_t}(\bar{y}_t^1,\cdots,\bar{y}_t^d)^T v_k v_k^T G_{S_t}(\bar{y}_t^1,\cdots,\bar{y}_t^d)](x_t-(x_t^T y_0)y_0). \tag{59}$$

Using the definition of M and $\|\sum_{k=1}^d v_k v_k^T\| = 1$, we have

$$\eta^2(x_t-(x_t^T y_0)y_0)^T \sum_{k=1}^d E[\|G_{S_t}(\bar{y}_t^1,\cdots,\bar{y}_t^d)^T v_k\|^2](x_t-(x_t^T y_0)y_0) \le \eta^2 K\|x_t-(x_t^T y_0)y_0\|^2. \tag{60}$$

Using (58), (59), (35), (36), (60) and the Cauchy-Schwarz inequality for the cross term as

$$\frac{1}{2}\eta\|x_t\|\sum_{k=1}^K(1-\eta+\eta(\lambda_k+(\lambda^*-\lambda_1)\mathbb{1}_{k=1}))x_k^T v_k(y_t-x^*)^T F_1(\hat{y}_t^1,\cdots,\hat{y}_t^d)(y_t-x^*) \\ \le \frac{1}{2}\eta M(1-\eta+\eta\lambda^*)\|x_t\|\|y_k-x^*\|^2, \tag{61}$$

we have

$$E[\|x_{t+1}\|^2|x_t] \le (1-\eta+\eta\lambda^*)^2\|x_t\|^2 + \frac{1}{2}\eta M(1-\eta+\eta\lambda^*)\|x_t\|\|y_k-x^*\|^2 \\ + \frac{1}{4}\eta^2 M^2\|x_t\|^2\|y_k-x^*\|^4 + \eta^2 K\|x_t-(x_t^T y_0)y_0\|^2. \tag{62}$$

Using $\|x_t-(x_t^T y_0)y_0\|^2 \le \|x_t\|^2$ in (62), we obtain

$$E[\|x_{t+1}\|^2|x_t] \le \Big[\big(1-\eta+\eta\lambda^*+\frac{1}{2}\eta M\|y_t-x^*\|^2\big)^2 + \eta^2 K\Big]\|x_t\|^2 \\ = \Big[\big(1-\eta+\eta\lambda^*+\eta M(1-y_t^T x^*)\big)^2 + \eta^2 K\Big]\|x_t\|^2, \tag{63}$$

which establishes the first statement.

In the same way, using (59), (34), (36), (60) and the Cauchy-Schwarz inequality similarly to (61), we have

$$E\Big[\sum_{k=2}^d (x_{t+1}^T v_k)^2|x_t\Big] \le \Big[(1-\eta+\eta\bar{\lambda})\sqrt{\sum_{k=2}^d (x_t^T v_k)^2} + \frac{1}{2}\eta M\|x_t\|\|y_t-x^*\|^2\Big]^2 \\ + \eta^2 K\|x_t-(x_t^T y_0)y_0\|^2. \tag{64}$$

By Lemma 3.2, we have $\Delta_t \le \Delta_0 \le 1-1/\sqrt{2}$ and thus $y_t^T x^* \ge 1/\sqrt{2}$ and $y_0^T x^* \ge 1/\sqrt{2}$. Since $y_t^T x^* \ge 0$, using (40), we have

$$\frac{1}{2}\eta M\|x_t\|\|y_t-x^*\|^2 \le \eta M\sqrt{\Delta_t}\sqrt{\sum_{k=2}^d (x_t^T v_k)^2}.$$

As a result of (32) which we can use since $\Delta_t \le \Delta_0$, we obtain

$$E\Big[\sum_{k=2}^d (x_{t+1}^T v_k)^2|x_t\Big] \le \Big[1-\eta+\eta\bar{\lambda}+\eta M\sqrt{\Delta_t}\Big]^2\sum_{k=2}^d (x_t^T v_k)^2 + 8\eta^2 K\|x_t\|^2\sum_{k=2}^d (y_0^T v_k)^2, \tag{65}$$

which shows the second statement in the lemma.

Lastly, from (58), we have

$$E[(x_{t+1}^T v_1)^2|x_t] \geq \left[(1-\eta+\eta\lambda^*)x_t^T v_1 + \frac{1}{2}\eta\|x_t\|(y_t-x^*)^T F_1(\hat{y}_t^1,\cdots,\hat{y}_t^d)(y_t-x^*)\right]^2$$

By (11) and (22), we have $\eta(1-\lambda^*+M\Delta_0\sqrt{2}) \leq 1$. Since $1/(1-\Delta_0) \leq \sqrt{2}$ by (6), we further have

$$\eta\left(\frac{M\Delta_0}{1-\Delta_0}+1-\lambda^*\right) \leq 1.$$

Due to $\Delta_t \leq \Delta_0$, this implies that

$$\begin{aligned}
(1-\eta+\eta\lambda^*)x_t^T v_1 - \frac{1}{2}\eta M\|x_t\|\|y_t-x^*\|^2 &= \Big[(1-\eta+\eta\lambda^*)(1-\Delta_t) - \eta M\Delta_t\Big]\|x_t\| \\
&= \Big[1-\eta\Big(\frac{M\Delta_t}{1-\Delta_t}+1-\lambda^*\Big)\Big](1-\Delta_t)\|x_t\| \\
&\geq \Big[1-\eta\Big(\frac{M\Delta_0}{1-\Delta_0}+1-\lambda^*\Big)\Big](1-\Delta_t)\|x_t\| \\
&\geq 0.
\end{aligned}$$

Since $(a+b)^2 \geq (a-c)^2$ holds if $a \geq c$ and $|b| \leq c$, we finally have

$$\begin{aligned}
E[(x_{t+1}^T v_1)^2|x_t] &\geq \Big[(1-\eta+\eta\lambda^*)x_t^T v_1 - \frac{1}{2}\eta M\|x_t\|\|y_t-x^*\|^2\Big]^2 \\
&= \left[1-\eta+\eta\lambda^* - \eta M\left(\frac{1-y_t^T x^*}{y_t^T x^*}\right)\right]^2 (x_t^T v_1)^2 \\
&= \Big[\alpha(\eta) - \frac{\eta M\Delta_t}{1-\Delta_t}\Big]^2 (x_t^T v_1)^2.
\end{aligned}$$

□

Proof of Lemma 3.4. By Lemma 3.2, we have $\Delta_t \leq \Delta_0$. Repeatedly applying Lemma 3.3, we have

$$\begin{aligned}
E[\|x_t\|^2|x_0] = E[E[\|x_t\|^2|x_{t-1}]|x_0] &\leq \big[\big(\alpha(\eta)+\eta M\Delta_0\big)^2 + \eta^2 K\big]E[\|x_{t-1}\|^2|x_0] \\
&\leq \big[\big(\alpha(\eta)+\eta M\Delta_0\big)^2 + \eta^2 K\big]^t \|x_0\|^2.
\end{aligned} \tag{66}$$

Using (66), we have

$$\begin{aligned}
E\Big[\|x_t\|^2 \sum_{k=2}^d (y_0^T v_k)^2\Big] &= E\Big[E\Big[\|x_t\|^2 \sum_{k=2}^d (y_0^T v_k)^2|x_0\Big]\Big] = E\Big[E\big[\|x_t\|^2|x_0\big] \sum_{k=2}^d (y_0^T v_k)^2\Big] \\
&= E\Big[\big[(\alpha(\eta)+\eta M\Delta_0)^2 + \eta^2 K\big]^t \|x_0\|^2 \sum_{k=2}^d (y_0^T v_k)^2\Big] \\
&= \big[(\alpha(\eta)+\eta M\Delta_0)^2 + \eta^2 K\big]^t E\Big[\sum_{k=2}^d (x_0^T v_k)^2\Big].
\end{aligned} \tag{67}$$

Using Lemma 3.3 and that $\Delta_t \leq \Delta_0$, we have

$$E\Big[\sum_{k=2}^d (x_t^T v_k)^2\Big] \leq \big(\beta(\eta)+\eta M\sqrt{\Delta_0}\big)^2 E\Big[\sum_{k=2}^d (x_{t-1}^T v_k)^2\Big] + 8\eta^2 K E\Big[\|x_{t-1}\|^2 \sum_{k=2}^d (y_0^T v_k)^2\Big]. \tag{68}$$

By induction on (68) using (67), we have

$$\begin{aligned}
E\Big[\sum_{k=2}^{d}(x_t^T v_k)^2\Big] &\le \big(\beta(\eta)+\eta M\sqrt{\Delta_0}\big)^2 E\Big[\sum_{k=2}^{d}(x_{t-1}^T v_k)^2\Big] \\
&\quad + 8\eta^2 K\big[(\alpha(\eta)+\eta M\Delta_0)^2+\eta^2 K\big]^{t-1} E\Big[\sum_{k=2}^{d}(x_0^T v_k)^2\Big] \\
&\le E\Big[\sum_{k=2}^{d}(x_0^T v_k)^2\Big]\Big[\big(\beta(\eta)+\eta M\sqrt{\Delta_0}\big)^{2t} \\
&\quad + 8\eta^2 K\sum_{s=1}^{t}\big(\alpha(\eta)+\eta M\sqrt{\Delta_0}\big)^{2(t-s)}\big[(\alpha(\eta)+\eta M\sqrt{\Delta_0})^2+\eta^2 K\big]^{s-1}\Big] \\
&\le E\Big[\sum_{k=2}^{d}(x_0^T v_k)^2\Big]\Bigg[\big(\beta(\eta)+\eta M\sqrt{\Delta_0}\big)^{2t} \\
&\quad + 8\big(\alpha(\eta)+\eta M\sqrt{\Delta_0}\big)^{2t}\Bigg[\bigg(1+\frac{\eta^2 K}{(\alpha(\eta)+\eta M\sqrt{\Delta_0})^2}\bigg)^{t}-1\Bigg]\Bigg] \\
&\le E\Big[\sum_{k=2}^{d}(x_0^T v_k)^2\Big]\Bigg[\big(\beta(\eta)+\eta M\sqrt{\Delta_0}\big)^{2t} \\
&\quad + 8\big(\alpha(\eta)+\eta M\sqrt{\Delta_0}\big)^{2t}\Bigg[\exp\bigg(\frac{\eta^2 Kt}{(\alpha(\eta)+\eta M\sqrt{\Delta_0})^2}\bigg)-1\Bigg].
\end{aligned}$$

By (12) and (22), we have $\eta\big(1-\lambda^*-M\sqrt{\Delta_0}+\sqrt{Km}\big)\le 1$, which leads to

$$0 \le \frac{\eta^2 Kt}{\big(\alpha(\eta)+\eta M\sqrt{\Delta_0}\big)^2} \le 1.$$

Using $\exp(x)-1\le 2x$ for $x\in[0,1]$, we have

$$E\Big[\sum_{k=2}^{d}(x_t^T v_k)^2\Big] \le E\Big[\sum_{k=2}^{d}(x_0^T v_k)^2\Big]\Big[\big(\beta(\eta)+\eta M\sqrt{\Delta_0}\big)^{2t}+16\eta^2 Kt\big(\alpha(\eta)+\eta M\sqrt{\Delta_0}\big)^{2(t-1)}\Big].$$

On the other hand, using $\Delta_t\le\Delta_0$ and Lemma 3.3, we have

$$E[(x_t^T v_1)^2] = E[E[(x_t^T v_1)^2|x_{t-1}]] \ge \bigg[\alpha(\eta)-\frac{\eta M\Delta_0}{1-\Delta_0}\bigg]^2 E[(x_{t-1}^T v_1)^2]. \tag{69}$$

By induction on (69) using $\Delta_t\le\Delta_0$, we finally have

$$E[(x_t^T v_1)^2] \ge \bigg[\alpha(\eta)-\frac{\eta M\Delta_0}{1-\Delta_0}\bigg]^{2t} E[(x_0^T v_1)^2].$$

□

Proof of Lemma 3.5. By (13) and (14), we have (12). Also, (13), (15) and the fact that $\sqrt{2\Delta_0}\le 1$ which holds from (6) imply (11). Therefore, by Lemma 3.4, we have

$$\delta_m \le \Bigg[\bigg(\frac{\beta(\eta)+\eta M\sqrt{\Delta_0}}{\alpha(\eta)-\eta M\Delta_0/(1-\Delta_0)}\bigg)^{2m}+\frac{16\eta^2 Km\big[\alpha(\eta)+\eta M\sqrt{\Delta_0}\big]^{2(m-1)}}{[\alpha(\eta)-\eta M\Delta_0/(1-\Delta_0)]^{2m}}\Bigg]\delta_0 \tag{70}$$

where

$$\delta_t = \frac{E[\sum_{k=2}^{d}(x_t^T v_k)^2]}{E[(x_t^T v_1)^2]}.$$

By (23) which follows from (6) and the fact that $(1+x)^m \le \exp(mx)$ for all $x \in \mathbb{R}$, we have

$$\begin{aligned}\left(\frac{\beta(\eta)+\eta M\sqrt{\Delta_0}}{\alpha(\eta)-\eta M\Delta_0/(1-\Delta_0)}\right)^{2m} &\le \left(1-\frac{\eta(\lambda^*-\bar{\lambda}-2M\sqrt{\Delta_0})}{1-\eta+\eta(\lambda^*-M\sqrt{\Delta_0})}\right)^{2m} \\ &\le \exp\left(-\frac{2\eta m(\lambda^*-\bar{\lambda}-2M\sqrt{\Delta_0})}{1-\eta+\eta(\lambda^*-M\sqrt{\Delta_0})}\right).\end{aligned}$$

Since (13), (15) and (22) imply

$$\frac{2\eta m(\lambda^*-\bar{\lambda}-2M\sqrt{\Delta_0})}{1-\eta+\eta(\lambda^*-M\sqrt{\Delta_0})} \le 1,$$

using the fact that $\exp(-x) \le 1-x/2$ for $0 \le x \le 1$, we have

$$\exp\left(-\frac{2\eta m(\lambda^*-\bar{\lambda}-2M\sqrt{\Delta_0})}{1-\eta+\eta(\lambda^*-M\sqrt{\Delta_0})}\right) \le 1-\frac{\eta m(\lambda^*-\bar{\lambda}-2M\sqrt{\Delta_0})}{1-\eta+\eta(\lambda^*-M\sqrt{\Delta_0})} = 1-2\rho. \tag{71}$$

On the other hand, by (23) and the fact that $(1+x)^n \le \exp(nx)$, we have

$$\begin{aligned}\frac{16\eta^2 Km\left[\alpha(\eta)+\eta M\sqrt{\Delta_0}\right]^{2(m-1)}}{[\alpha(\eta)-\eta M\Delta_0/(1-\Delta_0)]^{2m}} &\le \frac{16\eta^2 Km}{(\alpha(\eta)+\eta M\sqrt{\Delta_0})^2}\left(1+\frac{2\eta M\sqrt{\Delta_0}}{\alpha(\eta)-\eta M\sqrt{\Delta_0}}\right)^{2m} \\ &\le \frac{16\eta^2 Km}{(\alpha(\eta)+\eta M\sqrt{\Delta_0})^2}\exp\left(\frac{4\eta m M\sqrt{\Delta_0}}{\alpha(\eta)-\eta M\sqrt{\Delta_0}}\right).\end{aligned} \tag{72}$$

By (13), (14) and (22), we have

$$\eta\left(1-\lambda^*-M\sqrt{\Delta_0}+\frac{64K}{\lambda^*-\bar{\lambda}-2M\sqrt{\Delta_0}}\right) \le 1,$$

which leads to

$$\frac{\rho}{2}-\frac{16\eta^2 Km}{(\alpha(\eta)+\eta M\sqrt{\Delta_0})^2} \ge \frac{\eta m(\lambda^*-\bar{\lambda}-2M\sqrt{\Delta_0})}{4\big(1-\eta+\eta(\lambda^*+M\sqrt{\Delta_0})\big)}-\frac{16\eta^2 Km}{(1-\eta+\eta(\lambda^*+M\sqrt{\Delta_0}))^2} \ge 0. \tag{73}$$

In a similar way, by (13), (15) and (22), we have

$$\eta\left(1-\lambda^*+M\sqrt{\Delta_0}+\frac{4mM\sqrt{\Delta_0}}{\log 2}\right) \le 1,$$

which results in

$$\exp\left(\frac{4\eta m M\sqrt{\Delta_0}}{\alpha(\eta)-\eta M\sqrt{\Delta_0}}\right) \le 2. \tag{74}$$

Using (71), (72), (73) and (74) in (70), we finally have

$$\frac{E[\sum_{k=2}^d (x_m^T v_k)^2]}{E[(x_m^T v_1)^2]} \le (1-\rho)\cdot\frac{E[\sum_{k=2}^d (x_0^T v_k)^2]}{E[(x_0^T v_1)^2]}.$$

□

Proof of Theorem 3.6. Since η, s and $x_0 = \tilde{x}_0$ satisfy (6), (7) (or (8)) and (13), by Lemmas 3.2 and 3.5, we have

$$\tilde{\Delta}_1 = \Delta_m \le \Delta_0 = \tilde{\Delta}_0, \quad \tilde{\delta}_1 = \delta_m \le (1-\rho)\delta_0 = (1-\rho)\tilde{\delta}_0.$$

By repeatedly applying the same argument, we have $\tilde{\delta}_\tau \le (1-\rho)^\tau \tilde{\delta}_0$. Since $\tau \ge (1/\rho)\log(\tilde{\delta}_0/\epsilon)$, we finally obtain

$$\tilde{\delta}_\tau \le (1-\rho)^\tau \tilde{\delta}_0 \le \exp(-\tau\rho)\tilde{\delta}_0 \le \epsilon.$$

This completes the proof. □