Noname manuscript No.
(will be inserted by the editor)

Empirical Risk Minimization: Probabilistic Complexity and Stepsize Strategy

Chin Pang Ho · Panos Parpas

Received: date / Accepted: date

Abstract Empirical risk minimization (ERM) is recognized as a special form in standard convex optimization. When using a first order method, the Lipschitz constant of the empirical risk plays a crucial role in the convergence analysis and stepsize strategies for these problems. We derive the probabilistic bounds for such Lipschitz constants using random matrix theory. We show that, on average, the Lipschitz constant is bounded by the ratio of the dimension of the problem to the amount of training data. We use our results to develop a new stepsize strategy for first order methods. The proposed algorithm, Probabilistic Upper-bound Guided stepsize strategy (PUG), outperforms the regular stepsize strategies with strong theoretical guarantee on its performance.

Keywords Empirical risk minimization · Complexity analysis · Stepsize strategy

1 Introduction

Empirical risk minimization (ERM) is one of the most powerful tools in applied statistics, and is regarded as the canonical approach to regression analysis. In the context of machine learning and big data analytics, various important problems such as support vector machines, (regularized) linear regression, and logistics regression can be cast as ERM problems, see for e.g. [17]. In an ERM

The work of the authors was supported, in part, by EPSRC grants EP/M028240/1 and EP/K040723, by FP7 Marie Curie Career Integration Grant (PCIG11-GA-2012-321698 SOC-MP-ES), and by the Imperial College Junior Research Fellowship.

C. P. Ho
Imperial College Business School, Imperial College London, Ayrton Road, London SW7 2AZ
E-mail: c.ho12@imperial.ac.uk

P. Parpas
Department of Computing, Imperial College London, 180 Queen's Gate, London, SW7 2AZ

problem, a training set with m instances, $\{(\mathbf{a}_i, b_i)\}_{i=1}^m$, is given, where $\mathbf{a}_i \in \mathbb{R}^n$ is an input and $b_i \in \mathbb{R}$ is the corresponding output, for $i = 1, 2, \dots, m$. The ERM problem is then defined as the following convex optimization problem,

$$\min_{\mathbf{x}\in\mathbb{R}^n} \left\{ F(\mathbf{x}) \triangleq \frac{1}{m}\sum_{i=1}^m \phi_i(\mathbf{a}_i^T\mathbf{x}) + g(\mathbf{x}) \right\}, \tag{1}$$

where each loss function ϕ_i is convex with a Lipschitz continuous gradient, and the regularizer $g : \mathbb{R}^n \to \mathbb{R}$ is a continuous convex function which is possibly nonsmooth. Two common loss functions are

- Quadratic loss function: $\phi_i(x) = \dfrac{1}{2}(x - b_i)^2$.
- Logistic loss function: $\phi_i(x) = \log(1 + \exp(-xb_i))$.

One important example of g is the scaled 1-norm $\omega\|\mathbf{x}\|_1$ with a scaling factor $\omega \in \mathbb{R}^+$. This particular case is known as ℓ_1 regularization, and it has various applications in statistics [3], machine learning [18], signal processing [6], etc. The regularizer g acts as an extra penalty function to regularize the solution of (1). ℓ_1 regularization encourages sparse solutions, i.e. it favors solutions $\mathbf{x}$ with few non-zero elements. This phenomenon can be explained by the fact that the ℓ_1 norm is the tightest convex relaxation of the ℓ_0 norm, i.e. the cardinality of the non-zero elements of $\mathbf{x}$ [5].

In general, if the regularizer g is nonsmooth, subgradient methods are used to solve (1). However, subgradient methods are not advisable if g is simple enough, and one can achieve higher efficiency by generalizing existing algorithms for unconstrained differentiable convex programs. Much research has been undertaken to efficiently solve ERM problems with simple g's. Instead of assuming the objective function is smooth and continuously differentiable, they aim to solve problems of the following form

$$\min_{\mathbf{x}\in\mathbb{R}^n} \{F(\mathbf{x}) \triangleq f(\mathbf{x}) + g(\mathbf{x})\}, \tag{2}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is a convex function with L-Lipschitz continuous gradient, and $g : \mathbb{R}^n \to \mathbb{R}$ is a continuous convex function which is nonsmooth but simple. By simple we mean that a proximal projection step can be performed either in closed form or is at least computationally inexpensive. Norms, and the ℓ_1 norm in particular satisfy this property. A function f is said to have a L-Lipschitz continuous gradient if

$$\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\| \leq L\|\mathbf{x} - \mathbf{y}\|, \quad \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^n. \tag{3}$$

For the purpose of this paper, we denote the matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$ to be a dataset such that the i^{th} row of $\mathbf{A}$ is $\mathbf{a}_i^T$, and so in the case of ERM problems,

$$f(\mathbf{x}) = \frac{1}{m}\sum_{i=1}^m \phi_i(\mathbf{a}_i^T\mathbf{x}) = \frac{1}{m}\sum_{i=1}^m \phi_i(\mathbf{e}_i^T\mathbf{A}\mathbf{x}), \tag{4}$$

where $\mathbf{e}_i \in \mathbb{R}^m$ has 1 on its i^{th} component and 0's elsewhere. f is called the empirical risk in ERM. We assume that each ϕ_i have a γ_i-Lipschitz continuous gradient and

$$\gamma \triangleq \max\{\gamma_1, \gamma_2, \dots, \gamma_m\}.$$

Many algorithms [1,4,9,13,20] have been developed to solve (1) and (2). One famous example is the Fast Iterative Shrinkage-Thresholding Algorithm (FISTA) [1], which is a generalization of the optimal method proposed by Nesterov [10] for unconstrained differentiable convex programs. FISTA, with backtracking stepsize strategy, is known to converge according to the following rate,

$$F(\mathbf{x}_k) - F(\mathbf{x}_\star) \leq \frac{2\eta L \|\mathbf{x}_0 - \mathbf{x}_\star\|^2}{(k+1)^2}, \tag{5}$$

where $\mathbf{x}_\star$ is a solution of (2), and η is the parameter which is used in the backtracking stepsize strategy. The convergence result in (5) contains three key components: the distance between the initial guess and the solution $\|\mathbf{x}_0 - \mathbf{x}_\star\|$, the number of iterations k, and the Lipschitz constant L. While it is clear that the first two components are important to explain the convergence behavior, the Lipschitz constant, L, is relatively mysterious.

The appearance of L in (5) is due to algorithm design. In each iteration, one would have to choose a constant $\tilde{L}$ to compute the stepsize that is proportional to $1/\tilde{L}$, and $\tilde{L}$ has to be large enough to satisfy the properties of the Lipschitz constant locally [1,13]. Since the global Lipschitz constant condition (3) is a more restrictive condition, the Lipschitz constant L always satisfies the requirement of $\tilde{L}$, and so L is used in convergence analysis. We emphasize that the above requirement of $\tilde{L}$ is not unique for FISTA. For most first order methods that solve (2), L also appears in their convergence rates for the same reason.

Despite L being an important quantity in both convergence analysis and stepsize strategy, it is usually unknown and the magnitude could be arbitrary for a general nonlinear function; one could artificially construct a small dimensional function with large Lipschitz constant, and a high dimensional function with small Lipschitz constant.

Therefore, L is often treated as a constant [10,11] that is independent of the dimensions of the problem, and so the convergence result shown in (5) is considered to be "dimension-free" because both $\|\mathbf{x}_0 - \mathbf{x}_\star\|$ and k are independent of the dimension of the problem. Dimension-free convergence shows that for certain types of optimization algorithms, the number of iterations required to achieve a certain accuracy is independent of the dimension of the model. For large scale optimization models that appear in machine learning and big data applications, algorithms with dimension-free convergence are extremely attractive [1,2,16].

On the other hand, since L is considered to be an arbitrary constant, stepsize strategies for first order methods were developed independent of the knowledge of L. As we will show later, for adaptive strategies that try to use small $\tilde{L}$ (large stepsize), extra function evaluations will be needed. If one try to eliminate the extra function evaluations, then $\tilde{L}$ has to be sufficiently large, and thus the stepsize would be small. This trade-off is due to the fact that L is unknown.

In this paper, we take the first steps to show that knowledge of L can be obtained in the case of ERM because of its statistical properties. For the ERM problem, it is known that the Lipschitz constant is highly related to $\|\mathbf{A}\|$ [1,14], and so understanding the properties of $\|\mathbf{A}\|$ is the goal of this paper. If $\mathbf{A}$ is arbitrary, then $\|\mathbf{A}\|$ would also be arbitrary and analyzing $\|\mathbf{A}\|$ would be impossible. However, for ERM problems that appear in practice, $\mathbf{A}$ is structured. Since $\mathbf{A}$ is typically constructed from a dataset then it is natural to assume that the rows of $\mathbf{A}$ are independent samples of some random variables. This particular structure of $\mathbf{A}$, allows us to consider $\mathbf{A}$ as a non-arbitrary but random matrix. We are therefore justified to apply techniques from random matrix theory to derive the statistical bounds for the Lipschitz constant.

The contributions of this paper is twofold:

(a) We obtain the average/probabilistic complexity bounds which provide better understanding of how the dimension, size of training set, and correlation affect the computational complexity. In particular, we showed that in the case of ERM, the complexity is not "dimension-free".

(b) The derived statistical bounds can be computed/estimated with almost no cost, which is an attractive benefit for algorithms. We develop a novel stepsize strategy called Probabilistic Upper-bound Guided stepsize strategy (PUG). We show that PUG may save unnecessary cost of function evaluations by adaptively choosing $\tilde{L}$ intelligently. Promising numerical results are provided at the end of this paper.

Many research on bounding extreme singular values using random matrix theory have been taken in recent years, e.g. see [15,8,19]. However, we would like to emphasize that developments in random matrix theory is not our objective. Instead, we would like to consider this topic as a new and important application of random matrix theory. To the best of our knowledge, no similar work has been done in understanding how the statistics of the training set would affect the Lipschitz constant, computational complexity, and stepsize.

2 Preliminaries

This paper studies the Lipschitz constant L of the empirical risk f given in (4). In order to satisfy condition (3), one could select an arbitrarily large L, however, this would create a looser bound on the complexity (see for e.g. (5)). Moreover, L also plays a big role in stepsize strategy for first order algorithms. In many cases such as FISTA, algorithms use stepsize that is proportional to $1/L$. Therefore, a smaller L is always preferable because it does not only imply lower computational complexity, but also allows a larger stepsize for algorithms. While the lowest possible L that satisfies (3) is generally very difficult to compute, in this section, we will estimate the upper and lower bounds of L using the dataset $\mathbf{A}$.

Notice that the Lipschitz constant condition (3) is equivalent to the following condition.

$$f(\mathbf{y}) \leq f(\mathbf{x}) + \nabla f(\mathbf{x})^T(\mathbf{y}-\mathbf{x}) + \frac{L}{2}\|\mathbf{y}-\mathbf{x}\|^2, \quad \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^n. \tag{6}$$

Therefore, a L that satisfies (6) also satisfies (3), and vice versa.

Proposition 1 *Suppose f is of the form (4), then L satisfies the Lipschitz constant condition (6) with*

$$\begin{aligned} L &\leq \left\| Diag\left(\sqrt{\frac{\gamma_1}{m}}, \cdots, \sqrt{\frac{\gamma_m}{m}}\right)\mathbf{A}\right\|^2 \\ &\leq \left\| Diag\left(\sqrt{\frac{\gamma_1}{m}}, \cdots, \sqrt{\frac{\gamma_m}{m}}\right)\right\|^2 \|\mathbf{A}\|^2 \leq \frac{\gamma}{m}\|\mathbf{A}\|^2. \end{aligned}$$

Proof See Proposition 2.1 in [14]. □

Proposition 1 provides an upper bound for L, where γ is the maximum Lipschitz constant of loss functions, and it is usually known or easy to compute. For example, it is known that $\gamma = 1$ for quadratic loss functions, and $\gamma = \max_i b_i^2/4$ for logistics loss functions.

The upper bound of L is tight for the class of ERM problems. We can prove that by considering the example of least squares, where we have

$$L = \frac{\gamma}{m}\|\mathbf{A}\|^2 = \frac{1}{m}\|\mathbf{A}\|^2.$$

In order to derive the lower bound of L, we need the following assumption.

Assumption 2 *There exists a positive constant $\tau > 0$ such that*

$$\phi_i(x) + \phi_i'(x)(y-x) + \frac{\tau}{2}|y-x|^2 \leq \phi_i(y), \quad \forall x, y \in \mathbb{R},$$

for $i = 1, 2, \ldots, m$.

The above assumption requires the strongly-convex loss function ϕ_i, which is not restrictive in practical setting. In particular, quadratic loss function satisfies Assumption 2, and the logistics loss function satisfies Assumption 2 within a bounded box $[-b, b]$ for any positive $b \in \mathbb{R}^+$. With the above assumption, we derive the lower bound of L using $\mathbf{A}$.

Proposition 3 *Suppose f is of the form (4) with ϕ_i satisfying Assumption 2 for $i = 1, 2, \ldots, m$, then L satisfies the Lipschitz constant condition (6) with*

$$\frac{\tau \lambda_{\min}(\mathbf{A}^T\mathbf{A})}{m} \leq L.$$

Proof By Assumption 2, for $i = 1, 2, \ldots, m$,

$$\phi_i(\mathbf{e}_i^T\mathbf{A}\mathbf{y}) \geq \phi_i(\mathbf{e}_i^T\mathbf{A}\mathbf{x}) + \phi_i'(\mathbf{e}_i^T\mathbf{A}\mathbf{x})(\mathbf{e}_i^T\mathbf{A}\mathbf{y} - \mathbf{e}_i^T\mathbf{A}\mathbf{x}) + \frac{\tau}{2}|\mathbf{e}_i^T\mathbf{A}\mathbf{y} - \mathbf{e}_i^T\mathbf{A}\mathbf{x}|^2.$$

Therefore,

$$\begin{aligned} f(\mathbf{y}) &\geq \frac{1}{m}\sum_{i=1}^{m}\left(\phi_i(\mathbf{e}_i^T\mathbf{A}\mathbf{x}) + \phi_i'(\mathbf{e}_i^T\mathbf{A}\mathbf{x})(\mathbf{e}_i^T\mathbf{A}\mathbf{y} - \mathbf{e}_i^T\mathbf{A}\mathbf{x}) + \frac{\tau}{2}|\mathbf{e}_i^T\mathbf{A}\mathbf{y} - \mathbf{e}_i^T\mathbf{A}\mathbf{x}|^2\right), \\ &= f(\mathbf{x}) + \frac{1}{m}\sum_{i=1}^{m}\left(\mathbf{e}_i^T\mathbf{A}\phi_i'(\mathbf{e}_i^T\mathbf{A}\mathbf{x})(\mathbf{y}-\mathbf{x}) + \frac{\tau}{2}|\mathbf{e}_i^T\mathbf{A}\mathbf{y} - \mathbf{e}_i^T\mathbf{A}\mathbf{x}|^2\right), \\ &= f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y}-\mathbf{x}\rangle + \frac{\tau}{2m}\|\mathbf{A}\mathbf{y} - \mathbf{A}\mathbf{x}\|^2, \\ &\geq f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y}-\mathbf{x}\rangle + \frac{\tau\lambda_{\min}(\mathbf{A}^T\mathbf{A})}{2m}\|\mathbf{y}-\mathbf{x}\|^2. \end{aligned}$$

□

From Proposition 1 and 3, we bound L using the largest and lowest eigenvalues of $\mathbf{A}^T\mathbf{A}$. Even though $\mathbf{A}$ can be completely different for different dataset, the statistical properties of $\mathbf{A}$ can be obtained via random matrix theory.

3 Complexity Analysis using Random Matrix Theory

In this section, we will study the statistical properties of $\|\mathbf{A}\|^2 = \|\mathbf{A}^T\mathbf{A}\| = \lambda_{\max}(\mathbf{A}^T\mathbf{A})$ as well as $\lambda_{\min}(\mathbf{A}^T\mathbf{A})$. Recall that $\mathbf{A}$ is an $m \times n$ matrix containing m observations, and each observation contains n measurements which are independent samples from n random variables, i.e. we assume the rows of the matrix $\mathbf{A}$ are samples from a vector of n random variables $\boldsymbol{\xi}^T = (\xi_1, \xi_2, \cdots, \xi_n)$ with covariance matrix $\boldsymbol{\Sigma} = \mathbb{E}\left[\boldsymbol{\xi}\boldsymbol{\xi}^T\right]$. To simplify the analysis, we assume, without loss of generality, that the observations are normalized, and so all the random variables have mean zero and unit variance. Therefore, $\mathbb{E}[\xi_i] = 0$ for $i = 1, 2, \cdots, n$, and the diagonal elements of $\boldsymbol{\Sigma}$ are all 1's. This assumption is useful and simplifies the arguments and the analysis of this section but it is not necessary. The results from this section could be generalized without the above assumption, but it does not give further insights for the purposes of this section. In particular, this assumption will be dropped for the proposed stepsize strategy PUG, and so PUG is vaild for all the datasets used in practice.

3.1 Statistical Bounds

We will derive both the upper and lower bounds for the average $\|\mathbf{A}\|^2$, and show that the average $\|\mathbf{A}\|^2$ increases nearly linearly in both m and n. The main tools for the proofs below can be found in [19].

3.1.1 Lower Bounds

The following Lemma follows from Jensen's inequality and plays a fundamental role on what is to follow.

Lemma 4 *For a sequence $\{\mathbf{Q}_k : k = 1, 2, \cdots, m\}$ of random matrices,*

$$\lambda_{\max}\left(\sum_k \mathbb{E}[\mathbf{Q}_k]\right) \leq \mathbb{E}\left[\lambda_{\max}\left(\sum_k \mathbf{Q}_k\right)\right].$$

Proof For details, see [19].

With Lemma 4, we can derive a lower bound on the expected $\|\mathbf{A}^T\mathbf{A}\|$.

We will start by proving the lower bound in the general setting, where the random variables are correlated with general covariance matrix $\mathbf{\Sigma}$; then, we will add assumptions on $\mathbf{\Sigma}$ to derive lower bounds in different cases.

Theorem 5 *Let $\mathbf{A}$ be an $m \times n$ random matrix in which its rows are independent samples of some random variables $\boldsymbol{\xi}^T = (\xi_1, \xi_2, \cdots, \xi_n)$ with $\mathbb{E}[\xi_i] = 0$ for $i = 1, 2, \cdots, n$, and covariance matrix $\mathbf{\Sigma}$. Denote $\mu_{\max} = \lambda_{\max}(\mathbf{\Sigma})$ then*

$$m\mu_{\max} = m\lambda_{\max}(\mathbf{\Sigma}) \leq \mathbb{E}\left[\|\mathbf{A}\|^2\right]. \tag{7}$$

In particular, if $\xi_1, \xi_2, \cdots, \xi_n$ are some random variables with zero mean and unit variance, then

$$\max\{m\mu_{\max}, n\} \leq \mathbb{E}\left[\|\mathbf{A}\|^2\right]. \tag{8}$$

Proof We first try to prove (7). Denote $\mathbf{a}_i^T$ as the i^{th} row of $\mathbf{A}$. We can rewrite $\mathbf{A}^T\mathbf{A}$ as

$$\mathbf{A}^T\mathbf{A} = \sum_{k=1}^{m} \mathbf{a}_k\mathbf{a}_k^T,$$

where $\mathbf{a}_k\mathbf{a}_k^T$,'s are independent random matrices with $\mathbb{E}\left[\mathbf{a}_k\mathbf{a}_k^T\right] = \mathbf{\Sigma}$. Therefore,

$$\begin{aligned}\mathbb{E}\left[\lambda_{\max}\left(\mathbf{A}^T\mathbf{A}\right)\right] &= \mathbb{E}\left[\lambda_{\max}\left(\sum_{k=1}^{m} \mathbf{a}_k\mathbf{a}_k^T\right)\right] \\ &\geq \lambda_{\max}\left(\sum_{k=1}^{m} \mathbb{E}\left[\mathbf{a}_k\mathbf{a}_k^T\right]\right) = m\lambda_{\max}(\mathbf{\Sigma}).\end{aligned}$$

In order to prove (8), we use the fact that

$$\mathbb{E}\left[\|\mathbf{A}\|^2\right] = \mathbb{E}\left[\|\mathbf{A}^T\|^2\right] = \mathbb{E}\left[\|\mathbf{A}\mathbf{A}^T\|\right] \geq \|\mathbb{E}\left[\mathbf{A}\mathbf{A}^T\right]\|,$$

where the last inequality is obtained by applying Jensen's inequality. Therefore, we can write $\mathbf{A}\mathbf{A}^T$ as

$$\mathbf{A}\mathbf{A}^T = \sum_{i=1}^{m}\sum_{j=1}^{m} \mathbf{a}_i^T\mathbf{a}_j\mathbf{Y}_{i,j},$$

where $\mathbf{Y}_{i,j} \in \mathbb{R}^{m\times m}$ is a matrix such that $(\mathbf{Y}_{i,j})_{p,q} = 1$ if $i = p$ and $j = q$, and otherwise $(\mathbf{Y}_{i,j})_{p,q} = 0$. By the assumption that each entry of $\mathbf{A}$ are random variable with zero mean and unit variance, we obtain

$$\mathbb{E}\left[\mathbf{a}_i^T\mathbf{a}_i\right] = \mathbb{E}\left[a_{i,1}^2 + a_{i,2}^2 + \cdots + a_{i,n}^2\right] = \mathbb{E}\left[a_{i,1}^2\right] + \mathbb{E}\left[a_{i,2}^2\right] + \cdots + \mathbb{E}\left[a_{i,n}^2\right] = n,$$

for $i = 1, 2, \cdots, m$, and for $i \neq j$,

$$\mathbb{E}\left[\mathbf{a}_i^T\mathbf{a}_j\right] = \mathbb{E}\left[a_{i,1}\right]\mathbb{E}\left[a_{j,1}\right] + \mathbb{E}\left[a_{i,2}\right]\mathbb{E}\left[a_{j,2}\right] + \cdots + \mathbb{E}\left[a_{i,n}\right]\mathbb{E}\left[a_{j,n}\right] = 0.$$

Therefore,

$$\mathbb{E}[\|\mathbf{A}\|^2] \geq \\ \|\mathbb{E}[\mathbf{A}\mathbf{A}^T]\| = \left\|\mathbb{E}\left[\sum_{i=1}^m\sum_{j=1}^m \mathbf{a}_i^T\mathbf{a}_j\mathbf{Y}_{i,j}\right]\right\| = \left\|\sum_{i=1}^m n\mathbf{Y}_{i,i}\right\| = \|n\mathbf{I}_n\| = n.$$

□

Theorem 5 provides a lower bound of the expected $\|\mathbf{A}^T\mathbf{A}\|$. The inequality in (7) is a general result and makes minimal assumptions on the covariance $\mathbf{\Sigma}$. Note that the lower bound is independent of n. The reason is that this general setting covers cases where $\mathbf{\Sigma}$ is not full rank: some ξ_i's could be fixed 0's instead of having unit variance. In fact, when all ξ_i's are 0's for $i = 1, 2, \cdots, n$, which implies $\mathbf{\Sigma} = \mathbf{0}_{n\times n}$, the bound (7) is tight because $\mathbf{A} = \mathbf{0}_{m\times n}$. For the setting that we consider in this paper, equation (8) is a tighter bound than (7) and depends on both m and n. In the case where all variables are independent, we could simplify the results above into the following.

Corollary 6 *Let $\mathbf{A}$ be an $m\times n$ random matrix in which its rows are independent samples of some random variables $\boldsymbol{\xi}^T = (\xi_1, \xi_2, \cdots, \xi_n)$ with $\mathbb{E}[\xi_i] = 0$, $\mathbb{E}[\xi_i^2] = 1$, and ξ_i's are independent for $i = 1, 2, \cdots, n$, then*

$$\max\{m, n\} \leq \mathbb{E}\left[\|\mathbf{A}\|^2\right]. \tag{9}$$

Proof Since all random variables are independent, $\mathbf{\Sigma} = \mathbf{I}_n$ and so $\mu_{\max} = \lambda_{\max}(\mathbf{\Sigma}) = 1$. □

3.1.2 Upper Bounds

In order to compute an upper bound of the expected $\|\mathbf{A}^T\mathbf{A}\|$, we first compute its tail bounds. The idea of the proof is to rewrite the $\mathbf{A}^T\mathbf{A}$ as a sum of independent random matrices, and then use the existing results in random matrix theory to derive the tail bounds of $\|\mathbf{A}^T\mathbf{A}\|$. We then compute the upper bound of the expected value. Notice that our approach for computing the tail bounds, in principle, is the same as in [19]. However, we present a tail bound that is easier to be integrated into the upper bound of the expected $\|\mathbf{A}^T\mathbf{A}\|$. That is, the derived bound can be directly used to bound $\|\mathbf{A}^T\mathbf{A}\|$ without any numerical constant.

In order to compute the tail bounds, the following two Lemmas will be used.

Lemma 7 ([19]) *Suppose that* $\mathbf{Q}$ *is a random positive semi-definite matrix that satisfies* $\lambda_{\max}(\mathbf{Q}) \leq 1$. *Then*

$$\mathbb{E}\left[e^{\theta \mathbf{Q}}\right] \preccurlyeq \mathbf{I} + (e^{\theta} - 1)(\mathbb{E}\left[\mathbf{Q}\right]), \quad \textit{for } \theta \in \mathbb{R},$$

where $\mathbf{I}$ *is the identity matrix in the correct dimension.*

Lemma 8 ([19]) *Consider a sequence* $\{\mathbf{Q}_k : k = 1, 2, \cdots, m\}$ *of independent, random, self-adjoint matrices with dimension* n. *For all* $t \in \mathbb{R}$,

$$\mathbb{P}\left\{\lambda_{\max}\left(\sum_{k=1}^{m} \mathbf{Q}_k\right) \geq t\right\} \\ \leq n \inf_{\theta > 0} \exp\left(-\theta t + m \ \log \lambda_{\max}\left(\frac{1}{m}\sum_{k=1}^{m} \mathbb{E} e^{\theta \mathbf{Q}_k}\right)\right). \quad (10)$$

Combining the two results from random matrix theory, we can derive the following theorem for the tail bound of $\|\mathbf{A}^T\mathbf{A}\|$.

Theorem 9 *Let* $\mathbf{A}$ *be an* $m \times n$ *random matrix in which its rows are independent samples of some random variables* $\boldsymbol{\xi}^T = (\xi_1, \xi_2, \cdots, \xi_n)$ *for* $i = 1, 2, \cdots, n$, *and covariance matrix* $\mathbf{\Sigma} = \mathbb{E}\left[\boldsymbol{\xi}\boldsymbol{\xi}^T\right]$. *Denote* $\mu_{\max} = \lambda_{\max}(\mathbf{\Sigma})$ *and suppose*

$$\lambda_{\max}\left[\boldsymbol{\xi}\boldsymbol{\xi}^T\right] \leq R \quad \textit{almost surely.} \quad (11)$$

Then, for any $\theta, t \in \mathbb{R}^+$,

$$\mathbb{P}\left\{\lambda_{\max}\left(\mathbf{A}^T\mathbf{A}\right) \geq t\right\} \leq n \exp\left[-\theta t + m \ \log\left(1 + (e^{\theta R} - 1)\mu_{\max}/R\right)\right]. \quad (12)$$

In particular,

$$\mathbb{P}\left\{\lambda_{\max}\left(\mathbf{A}^T\mathbf{A}\right) \geq t\right\} \leq n \left[\frac{\mu_{\max}(mR - t)}{t(R - \mu_{\max})}\right]^{\frac{t}{R}} \left[1 + \frac{t - \mu_{\max} m}{mR - t}\right]^{m}. \quad (13)$$

Proof Denote $\mathbf{a}_i^T$ as the i^{th} row of $\mathbf{A}$. We can rewrite $\mathbf{A}^T\mathbf{A}$ as

$$\mathbf{A}^T\mathbf{A} = \sum_{k=1}^{m} \mathbf{a}_k \mathbf{a}_k^T.$$

Notice that $\mathbf{a}_k\mathbf{a}_k^T$'s are independent, random, positive-semidefinite matrices, and $\mathbb{E}\left[\mathbf{a}_k\mathbf{a}_k^T\right] = \mathbf{\Sigma}$, for $k = 1, 2, \cdots, m$. Also, Using the Lemma 8, for any $\theta > 0$, we have

$$\mathbb{P}\left\{\lambda_{\max}\left(\mathbf{A}^T\mathbf{A}\right) \geq t\right\} = \mathbb{P}\left\{\lambda_{\max}\left(\sum_{k=1}^{m} \mathbf{a}_k\mathbf{a}_k^T\right) \geq t\right\}, \\ \leq n \exp\left[-\theta t + m \ \log \lambda_{\max}\left(\frac{1}{m}\sum_{k=1}^{m} \mathbb{E} e^{\theta \mathbf{a}_k \mathbf{a}_k^T}\right)\right].$$

Notice that $\lambda_{\max}(\mathbf{a}_k\mathbf{a}_k^T) \leq R$, by rescaling on Lemma 7, we have,

$$\mathbb{E}\left[e^{\tilde{\theta}(1/R)(\mathbf{a}_k\mathbf{a}_k^T)}\right] \preccurlyeq \mathbf{I}_n + (e^{\tilde{\theta}} - 1)\left(\mathbb{E}\left[(1/R)\left(\mathbf{a}_k\mathbf{a}_k^T\right)\right]\right), \quad \text{for any } \tilde{\theta} \in \mathbb{R},$$

and thus

$$\mathbb{E}\left[e^{\theta(\mathbf{a}_k\mathbf{a}_k^T)}\right] \preccurlyeq \mathbf{I}_n + \frac{(e^{\theta R} - 1)}{R}\mathbb{E}\left[\mathbf{a}_k\mathbf{a}_k^T\right] = \mathbf{I}_n + \frac{(e^{\theta R} - 1)}{R}\mathbf{\Sigma}, \quad \text{for any } \theta \in \mathbb{R}.$$

$$\begin{aligned}\mathbb{P}\left\{\lambda_{\max}\left(\mathbf{A}^T\mathbf{A}\right) \geq t\right\} &\leq n\exp\left[-\theta t + m\ \log\lambda_{\max}\left(\frac{1}{m}\sum_{k=1}^{m}\mathbf{I}_n + \frac{(e^{\theta R} - 1)}{R}\mathbf{\Sigma}\right)\right], \\ &= n\exp\left[-\theta t + m\ \log\left(1 + (e^{\theta R} - 1)\lambda_{\max}(\mathbf{\Sigma})/R\right)\right], \\ &= n\exp\left[-\theta t + m\ \log\left(1 + (e^{\theta R} - 1)\mu_{\max}/R\right)\right].\end{aligned}$$

Using standard calculus, the upper bound is minimized when

$$\theta^\star = \frac{1}{R}\log\left[\frac{t(R - \mu_{\max})}{\mu_{\max}(mR - t)}\right].$$

Therefore,

$$\begin{aligned}\mathbb{P}\left\{\lambda_{\max}\left(\mathbf{A}^T\mathbf{A}\right) \geq t\right\} &\leq n\exp\left[-\theta^\star t + m\ \log\left(1 + (e^{\theta^\star R} - 1)\mu_{\max}/R\right)\right], \\ &= n\left[\frac{\mu_{\max}(mR - t)}{t(R - \mu_{\max})}\right]^{\frac{t}{R}}\left[1 + \frac{t - \mu_{\max}m}{mR - t}\right]^m.\end{aligned}$$

□

For matrices which contain samples from unbounded random variables, assumption (11) in Theorem 9 might seem to be restrictive; however, in practice, assumption (11) is mild due to the fact that datasets that are used in the problem (1) are usually normalized and bounded. Therefore, it is reasonable to assume that an observation will be discarded if its magnitude is larger than some constant.

The tail bound (13) is the tightest bound over all possible θ's in (12), but it is difficult to interpret the relationships between the variables. The following corollary takes a less optimal θ in (12), but yields a bound that is easier to understand.

Corollary 10 *In the same setting as Theorem 9, we have*

$$\mathbb{P}\left\{\lambda_{\max}\left(\mathbf{A}^T\mathbf{A}\right) \geq t\right\} \leq n\exp\left[\frac{2m\mu_{\max} - t}{R}\right]. \tag{14}$$

In particular, for $\epsilon \in \mathbb{R}^+$, we have

$$\mathbb{P}\left\{\lambda_{\max}\left(\mathbf{A}^T\mathbf{A}\right) \leq 2m\mu_{\max} - R\log\left(\frac{\epsilon}{n}\right)\right\} \geq 1 - \epsilon. \tag{15}$$

Proof Using equation (12), and the fact that $log(y) \leq y - 1$, $\forall y \in \mathbb{R}^+$, we have

$$\mathbb{P}\left\{\lambda_{\max}\left(\mathbf{A}^T\mathbf{A}\right) \geq t\right\} \leq n\exp\left[-\theta t + \frac{m\mu_{\max}}{R}(e^{\theta R} - 1)\right]. \tag{16}$$

The above upper bound is minimized when $\theta = (1/R)\log\left[t/(m\mu_{\max})\right]$, and so

$$\begin{aligned}
\mathbb{P}\left\{\lambda_{\max}\left(\mathbf{A}^T\mathbf{A}\right) \geq t\right\} &\leq n\exp\left[-\frac{t}{R}\log\left[\frac{t}{m\mu_{\max}}\right] + \frac{m\mu_{\max}}{R}\left(\frac{t}{m\mu_{\max}} - 1\right)\right], \\
&= n\exp\left[\frac{t}{R}\left(\log\left[\frac{m\mu_{\max}}{t}\right] + 1 - \frac{m\mu_{\max}}{t}\right)\right], \\
&= n\exp\left[\frac{t}{R}\left(\log\left[\frac{m\mu_{\max}e}{t}\right] - \frac{m\mu_{\max}}{t}\right)\right], \\
&\leq n\exp\left[\frac{t}{R}\left(\frac{m\mu_{\max}e}{t} - 1 - \frac{m\mu_{\max}}{t}\right)\right], \\
&\leq n\exp\left[\frac{1}{R}\left(m\mu_{\max}(e-1) - t\right)\right], \\
&\leq n\exp\left[\frac{1}{R}\left(2m\mu_{\max} - t\right)\right].
\end{aligned}$$

Set $\epsilon = n\exp\left[\frac{1}{R}\left(2m\mu_{\max} - t\right)\right]$, we obtain $t = 2m\mu_{\max} - R\log\left(\frac{\epsilon}{n}\right)$. □

The bound in (15) follows directly from (14) and shows that with high probability $1-\epsilon$ (for small ϵ), $\lambda_{\max}\left(\mathbf{A}^T\mathbf{A}\right)$ is less than $2m\mu_{\max} + R\log(n) - R\log(\epsilon)$. Applying the results in Corollary 10 provides the upper bound of the expected $\|\mathbf{A}^T\mathbf{A}\|$.

Corollary 11 *In the same setting as Theorem 9, we have*

$$\mathbb{E}\left[\lambda_{\max}(\mathbf{A}^T\mathbf{A})\right] \leq 2m\mu_{\max} + R\log(n) + R. \tag{17}$$

Proof Using the equation (14), and the fact that

$$1 \leq n\exp\left[\frac{2m\mu_{\max} - t}{R}\right] \quad \text{when} \quad t \leq 2m\mu_{\max} - R\log\left[\frac{1}{n}\right],$$

we have

$$\begin{aligned}
\mathbb{E}\left[\lambda_{\max}(\mathbf{A}^T\mathbf{A})\right] &= \int_0^\infty \mathbb{P}\{\lambda_{\max}(\mathbf{A}^T\mathbf{A}) > t\}\ \mathrm{d}t, \\
&\leq \int_0^{2m\mu_{\max} - R\log\left[\frac{1}{n}\right]} 1\ \mathrm{d}t \\
&\quad + \int_{2m\mu_{\max} - R\log\left[\frac{1}{n}\right]}^\infty n\exp\left[\frac{2m\mu_{\max} - t}{R}\right]\ \mathrm{d}t, \\
&= 2m\mu_{\max} - R\log\left[\frac{1}{n}\right] + R.
\end{aligned}$$

□

Therefore, for a matrix $\mathbf{A}$ which is constructed by a set of normalized data, we obtain the bound

$$\max\{m\mu_{\max}, n\} \leq \mathbb{E}\left[\|\mathbf{A}\|^2\right] \leq 2m\mu_{\max} + R\log(n) + R. \tag{18}$$

The result in (18) might look confusing because for small m and large n, the lower bound is of the order of n while the upper bound is of the order of $\log(n)$. The reason is that we have to take into account the factor of dimensionality in the constant R. To illustrate this, we prove the following corollary.

Corollary 12 *Let $\mathbf{A}$ be an $m \times n$ random matrix in which its rows are independent samples of some random variables $\boldsymbol{\xi}^T = (\xi_1, \xi_2, \cdots, \xi_n)$ with $\mathbb{E}[\xi_i] = 0$ for $i = 1, 2, \cdots, n$, and covariance matrix $\boldsymbol{\Sigma} = \mathbb{E}\left[\boldsymbol{\xi}\boldsymbol{\xi}^T\right]$. Denote $\mu_{\max} = \lambda_{\max}(\boldsymbol{\Sigma})$ and suppose $|\xi_i| \leq c$ almost surely for $i = 1, 2, \cdots, n$. Then*

$$\lambda_{\max}\left[\boldsymbol{\xi}\boldsymbol{\xi}^T\right] \leq c^2 n \quad \textit{almost surely.} \tag{19}$$

and so

$$\max\{m\mu_{\max}, n\} \leq \mathbb{E}\left[\|\mathbf{A}\|^2\right] \leq 2m\mu_{\max} + c^2 n\log(n) + c^2 n \tag{20}$$

Proof Since $\boldsymbol{\xi}\boldsymbol{\xi}^T$ is a symmetric rank 1 matrix, we have

$$\lambda_{\max}(\boldsymbol{\xi}\boldsymbol{\xi}^T) = \|\boldsymbol{\xi}\boldsymbol{\xi}^T\| \leq n\|\boldsymbol{\xi}\boldsymbol{\xi}^T\|_{\max} = n \max_{1\leq i,j\leq n}\{|\xi_i\xi_j|\} \leq c^2 n \quad \text{almost surely.}$$

□

Therefore, R increases linearly in n for bounded $\boldsymbol{\xi}$. Recall that the lower bound of the expected $\|\mathbf{A}\|^2$ is linear in both m and n, and the upper bound in (20), is almost linear in both m and n. Therefore, our results on the bounds for the expected Lipschitz constant are nearly-optimal up to some constant.

On the other hand, in order to obtain the lower bound of L, we also need tail bound of $\lambda_{\min}(\mathbf{A}^T\mathbf{A})$, which is provided in the following theorem.

Theorem 13 *Let $\mathbf{A}$ be an $m \times n$ random matrix in which its rows are independent samples of some random variables $\boldsymbol{\xi}^T = (\xi_1, \xi_2, \cdots, \xi_n)$ for $i = 1, 2, \cdots, n$, and covariance matrix $\boldsymbol{\Sigma} = \mathbb{E}\left[\boldsymbol{\xi}\boldsymbol{\xi}^T\right]$. Denote $\mu_{\min} = \lambda_{\min}(\boldsymbol{\Sigma})$ and suppose $|\xi_i| \leq c$ almost surely for $i = 1, 2, \ldots, n$.*

Then, if $\mu_{\min} \neq 0$, for any $\epsilon \in \left(n\exp\left[-\frac{m\mu_{\min}}{2nc^2}\right]\ , n\right)$

$$\mathbb{P}\left\{\lambda_{\min}\left(\mathbf{A}^T\mathbf{A}\right) \leq m\mu_{\min} - \sqrt{2c^2 nm\mu_{\min}\log\left(\frac{n}{\epsilon}\right)}\right\} \leq \epsilon.$$

Proof Suppose $|\xi_i| \leq c$ almost surely for $i = 1, 2, \ldots, n$. Then using Corollary 12 we have

$$\lambda_{\max}\left[\boldsymbol{\xi}\boldsymbol{\xi}^T\right] \leq c^2 n = R \quad \text{almost surely.}$$

Using the Theorem 1.1 from [19], for any $\theta \in (0, 1)$ we have

$$\begin{aligned}\mathbb{P}\left\{\lambda_{\min}\left(\mathbf{A}^T\mathbf{A}\right) \leq \theta m\mu_{\min}\right\} &\leq n\left[\frac{\exp[\theta - 1]}{\theta^\theta}\right]^{m\mu_{\min}/R}, \\ &= n\exp\left[\left(-(1-\theta) - \theta\log(\theta)\right)\left(\frac{m\mu_{\min}}{R}\right)\right].\end{aligned}$$

Notice that $\theta > 0$ and

$$2\log(\theta) \geq 2\left(1 - \frac{1}{\theta}\right) = \frac{2(\theta-1)}{\theta} \geq \frac{(\theta+1)(\theta-1)}{\theta} = \frac{\theta^2-1}{\theta},$$

and so

$$\begin{aligned}\mathbb{P}\left\{\lambda_{\min}\left(\mathbf{A}^T\mathbf{A}\right) \leq \theta m\mu_{\min}\right\} &\leq n\exp\left[\left(-(1-\theta) - \theta\log(\theta)\right)\left(\frac{m\mu_{\min}}{R}\right)\right], \\ &\leq n\exp\left[\left(-(1-\theta) - \theta\frac{\theta^2-1}{2\theta}\right)\left(\frac{m\mu_{\min}}{R}\right)\right], \\ &= n\exp\left[-\frac{1}{2}\left(\theta-1\right)^2\left(\frac{m\mu_{\min}}{R}\right)\right].\end{aligned}$$

For $\mu_{\min} \neq 0$, we let $\epsilon = n\exp\left[-\left(\theta-1\right)^2\frac{m\mu_{\min}}{2R}\right]$ and so

$$\theta = 1 - \sqrt{\frac{2R}{m\mu_{\min}}\log\left(\frac{n}{\epsilon}\right)}.$$

In particular, suppose $\epsilon \in \left(n\exp\left[-\frac{m\mu_{\min}}{2R}\right]\ , n\right)$, then

$$0 < 1 - \sqrt{\frac{2R}{m\mu_{\min}}\log\left(\frac{n}{\epsilon}\right)} < 1$$

Therefore,

$$\mathbb{P}\left\{\lambda_{\min}\left(\mathbf{A}^T\mathbf{A}\right) \leq m\mu_{\min} - \sqrt{2Rm\mu_{\min}\log\left(\frac{n}{\epsilon}\right)}\right\} \leq \epsilon,$$

for $\epsilon \in \left(n\exp\left[-\frac{m\mu_{\min}}{2R}\right]\ , n\right)$ □

For the tail bound in Theorem 13 to be meaningful, m has to be sufficiently large compared to n. In such cases, the smallest eigenvalue $\lambda_{\min}\left(\mathbf{A}^T\mathbf{A}\right)$ is at least $\mathcal{O}(m - \sqrt{nm\log n})$ with high probability.

3.2 Complexity Analysis

In this section, we will use the probabilistic bounds of L to study the complexity of solving ERM. We focus only on FISTA for illustrative purpose and clear presentation of the idea of the proposed approach. But the approach developed in this section can be applied to other algorithms as well.

By the assumption that $\mathbf{A}$ is a random matrix, we also have the solution $\mathbf{x}_\star$ as a random vector. Notice that the study of randomization of $\mathbf{x}_\star$ is not covered this paper. In particular, if the statistical properties of $\mathbf{x}_\star$ can be obtained, existing optimization algorithms might not be needed to solve the

ERM problem. Therefore, in this paper, we remove this consideration by denoting a constant M such that $\|\mathbf{x}_0 - \mathbf{x}_\star\|^2 \leq M$. In such case, we have the FISTA convergence rate

$$F(\mathbf{x}_k) - F(\mathbf{x}_\star) \leq \frac{2\eta LM}{(k+1)^2}, \tag{21}$$

where $\mathbf{x}_\star$ is the solution of (1), and $\eta > 1$ is the parameter which is used in the backtracking stepsize strategy.

Using Proposition 1 and Corollary 12, we know

$$\max\left\{\gamma\mu_{\max}, \frac{\gamma n}{m}\right\} \leq \frac{\gamma}{m}\mathbb{E}\left[\|\mathbf{A}\|^2\right] \leq 2\gamma\mu_{\max} + \frac{\gamma}{m}(c^2 n \log{(n)} + c^2 n). \tag{22}$$

Thus, on average,

$$F(\mathbf{x}_k) - F(\mathbf{x}_\star) \leq \frac{2\eta M}{(k+1)^2}\left(2\gamma\mu_{\max} + \frac{\gamma}{m}(c^2 n \log{(n)} + c^2 n)\right). \tag{23}$$

In (22)-(23), the lower bound of $(\gamma/m)\mathbb{E}\left[\|\mathbf{A}\|^2\right]$ is linear in n/m, and upper bound is nearly-linear in n/m. This suggests that the average complexity of ERM is bounded by the ratio of the dimensions to the amount of data. In particular, problems with overdetermined systems ($m >> n$) can be solved more efficiently than problems with underdetermined systems ($m < n$).

Another critical factor of the complexity is $\mu_{\max} = \lambda_{\max}(\mathbf{\Sigma})$, where $\mathbf{\Sigma}$ is the covariance matrix of the rows of $\mathbf{A}$. In the ideal situation of regression analysis, all inputs should be statistically linearly independent. In such cases, since we assume the diagonal elements of $\mathbf{\Sigma}$ are 1's, $\mu_{\max} = 1$. It is, however, almost impossible to ensure this situation for practical applications. In practice, since $\mathbf{\Sigma} \in \mathbb{R}^{n\times n}$, $\mu_{\max} = \lambda_{\max}(\mathbf{\Sigma}) = \|\mathbf{\Sigma}\|$ is likely to increase as n increases.

Similarly we can compute the probabilistic lower bound of L in the case that m is sufficiently larger than n. Using Theorem 13, we can show that L is bounded above by

$$\mathcal{O}\left(\mu_{\min} - \sqrt{(n\log n)/m}\right).$$

We emphasize the lower bound of L is not equivalent to the lower bound of the complexity. However, since the stepsize of first order method algorithms is proportional to $1/L$, this result indicates that the upper bound of the stepsize which potentially could guarantee convergence.

4 PUG: Probabilistic Upper-bound Guided stepsize strategy

The tail bounds in Section 3, as a by-product of the upper bound in Section 3.1.2, can also be used in algorithms. As mentioned in the introduction, L is an important quantity in the stepsize strategy since the stepsize is usually inversely proportional to L. However, in large scale optimization, the computational cost of evaluating $\|\mathbf{A}\|^2$ is very expensive. One could use backtracking techniques to avoid the evaluation of the Lipschitz constant; in each iteration, we find a large enough constant $\tilde{L}$ such that it satisfies the properties of the Lipschitz constant locally. In the case of FISTA [1], for the k^{th} iteration with incumbent $\mathbf{x}_k$ one has to find a $\tilde{L}$ such that

$$F\left(p_{\tilde{L}}(\mathbf{x}_k)\right) \leq Q_{\tilde{L}}\left(p_{\tilde{L}}(\mathbf{x}_k), \mathbf{x}_k\right), \tag{24}$$

where,

$$Q_{\tilde{L}}(\mathbf{x}, \mathbf{y}) \triangleq f(\mathbf{y}) + \langle \mathbf{x} - \mathbf{y}, \nabla f(\mathbf{y})\rangle + \frac{\tilde{L}}{2}\|\mathbf{x} - \mathbf{y}\|^2 + g(\mathbf{x}),$$

and $p_{\tilde{L}}(\mathbf{y}) \triangleq \arg\min_{\mathbf{x}}\{Q_{\tilde{L}}(\mathbf{x}, \mathbf{y}) : \mathbf{x} \in \mathbb{R}^n\}$. Equation (24) is identical to the Lipschitz constant condition (6) with specifically $\mathbf{y} = p_{\tilde{L}}(\mathbf{x}_k)$ and $\mathbf{x} = \mathbf{x}_k$. Therefore, (24) is a less restrictive condition compared to the Lipschitz constant condition (6). This indicates that $\tilde{L}$ could be much smaller than L, and so it yields to larger stepsize. On the other hand, for $\tilde{L} \geq L$, it is guaranteed that the local Lipschitz constant condition will be satisfied. In both cases, when computing L is intractable, we would not be able to distinguish the two cases by just having $\tilde{L}$ that satisfies (24).

As we can see, finding a good $\tilde{L}$ involves a series of function evaluations. In the next section, we will review the commonly used stepsize strategies.

4.1 Current Stepsize Strategies

To the best of our knowledge, current strategies fall into four categories:

(i). A fixed stepsize from estimation of $\|\mathbf{A}\|^2$.

(ii). Backtracking-type method with initial guess $\tilde{L}_0$, and monotonically increase $\tilde{L} = \eta^p \tilde{L}_0$ when it does not satisfy Lipschitz condition locally ($\eta > 1$, $p = 0, 1, \dots$). See [1] for details.

(iii). Adaptive-type method with initial guess $\tilde{L}_0$. Suppose $\tilde{L}_k$ is used for the k^{th} iteration, then find the smallest p such that $\tilde{L}_{k+1} = 2^p \tilde{L}_k$ satisfies Lipschitz condition locally ($p = -1, 0, 1, \dots$). See Nesterov's universal gradient methods [12] for details.

(iv). Adaptive stepsize strategy for a specific algorithm. See [7] for example.

Theorem 14 *Suppose $\tilde{L}$ is used as an initial guess for the k^{th} iteration, and we select the smallest $q \in \mathbb{N}$ such that $\tilde{L}_k = \eta^q \tilde{L}$ satisfies the local condition, for $\eta \geq 1$. To guarantee convergence, it requires*

$$q \geq \max\left\{\frac{1}{\log \eta}\left(\log L - \log \tilde{L}\right), 0\right\},$$

which is also the numbers of function evaluations required. We have

$$\begin{aligned} L \leq \tilde{L}_k \leq \eta L, &\quad \textit{if} \quad \tilde{L} \leq L, \\ L \leq \tilde{L}_k = \tilde{L}, &\quad \textit{if} \quad L \leq \tilde{L}. \end{aligned}$$

Proof To guarantee convergence, it requires q such that $\tilde{L}_k = \eta^q \tilde{L} \geq L$. If $\tilde{L} \leq L$, q should be selected such that $\eta^q \tilde{L} \leq \eta L$; otherwise $q - 1$ will be large enough to be selected, i.e. $\tilde{L}_k = \eta^{q-1} \tilde{L} \geq L$. □

Theorem 14 covers the setting of choice (i)-(iii), also referred to as the fixed stepsize strategy, backtracking method, and Nesterov's adaptive method, respectively. For fixed stepsize strategies, $\tilde{L} \geq L$ is selected for all iterations, which yields $q = 0$, and thus checking the local condition is not required [1]. For backtracking method, $\tilde{L} = \tilde{L}_{k-1}$ and $\eta > 1$ is a parameter of the strategy. Since $\tilde{L}_k$ is monotonically increasing in k, q is monotonically decreasing. Therefore, q at the k^{th} iteration is equivalent to the total number of (extra) function evaluations for the rest of the iterations.

On the other hand, for Nesterov's adaptive method, $\tilde{L} = \tilde{L}_{k-1}/2$ and $\eta = 2$. $\tilde{L}_k$ is not monotonically increasing in k, and in each iteration, q is the number of function evaluations in the worst case. Notice that once the worst case occurs (having q function evaluations) in the k^{th} iterations, q will be smaller since $\tilde{L}_k$ is sufficiently large. In Nesterov's universal gradient methods [12], Nesterov proved that for k iterations, the number of function evaluations is bounded by $\mathcal{O}(2k)$.

Theorem 14 illustrates the trade-off between three aspects: aggressiveness of initial guess $\tilde{L}$, recovering rate η, and the convergence rate. Methods with small (aggressive) initial guess $\tilde{L}$ have the possibility to result in larger stepsize. However, it will yield a larger q, the number of function evaluations in the worst case. One could reduce q by setting a larger η, and so $\tilde{L}$ could scale quickly towards L, but it will generate a slower rate of convergence (ηL). If one wants to preserve a good convergence rate (small η) with small number of function evaluations (small q), then $\tilde{L}$ could not be too small. In that case one has to give up on the opportunity of having large stepsizes. The fixed stepsize strategy is the extreme case of minimizing q by giving up the opportunity of having larger stepsizes.

The proposed stepsize strategy PUG tries to reduce $\tilde{L}$ as (iii), but guides $\tilde{L}$ to increase reasonably and quickly when it fails to satisfy the local condition. In particular, by replacing L with its probabilistic upper bound, aggressive $\tilde{L}$ and fast recovering rate are allowed without slowing the convergence. This above feature does not obey the trade-off that constraints choice (i)-(iii). Also, PUG is flexible compared to (iv). It can be applied to all algorithms that require L, as well as mini-batch and block-coordinate-type algorithms which require submatrix of $\mathbf{A}$.

4.2 PUG

In this section, we will use the tail bounds to develop PUG. Using equation (15), we first define the upper bound at different confidence level,

$$L \leq \mathcal{U}(\epsilon) \triangleq 2\gamma\mu_{\max} - \frac{\gamma R}{m} \log\left(\frac{\epsilon}{n}\right), \tag{25}$$

Algorithm 1 PUG

```
Input: $\tilde{L}_k$ from last iteration
Initialization: Set $\tilde{L} = \tilde{L}_k/2$, $\epsilon = \min\{0.1, \epsilon_0\}$ (Require: $\epsilon_0$ small enough such that $\mathcal{U}(\epsilon) > \tilde{L}$)
Set $\eta_{\text{PUG}} = \sqrt{\mathcal{U}(\epsilon)/\tilde{L}}$
while $\tilde{L}$ does not satisfy Lipschitz constant condition locally do
    Set $\tilde{L} = \eta_{\text{PUG}}\tilde{L}$
end while
Output: Lipschitz constant $\tilde{L}_{k+1} = \tilde{L}$
```

with probability of at least $1-\epsilon$. We point out that the probabilistic upper bound (15) does not rely on the assumption that the dataset is normalized with mean zero and unit variance, and so it is applicable to all types of datasets. The basic idea of PUG is to use the result in the following Theorem.

Theorem 15 *Suppose $\tilde{L}$ is used as an initial guess for the k^{th} iteration, and we denote*

$$\eta_{PUG,N} = \left(\frac{\mathcal{U}(\epsilon)}{\tilde{L}}\right)^{1/N},$$

where $\mathcal{U}(\epsilon)$ is defined as in (25). If we select the smallest $q \in \mathbb{N}$ such that $\tilde{L}_k = \eta^q_{PUG,N}\tilde{L}$ satisfies the local condition, then with probability of at least $1-\epsilon$, it requires $q = N$ to guarantee convergence. In particular, we have

$$\begin{aligned} L \leq \tilde{L}_k \leq \mathcal{U}(\epsilon), &\quad \text{if} \quad \tilde{L} \leq L, \\ L \leq \tilde{L}_k = \tilde{L}, &\quad \text{if} \quad L \leq \tilde{L}, \end{aligned}$$

with probability of at least $1-\epsilon$.

Proof To guarantee convergence, it requires q such that $\tilde{L}_k = \eta^q_{\text{PUG},N}\tilde{L} \geq L$. When $q = N$, $\tilde{L}_k = \mathcal{U}(\epsilon) \geq L$ with probability of at least $1-\epsilon$. □

Theorem 15 shows the potential advantage of PUG. With any initial guess $\tilde{L}$, PUG is able to scale $\tilde{L}$ quickly towards L without interfering with the probabilistic convergence rate. This unique feature allows an aggressive initial guess $\tilde{L}$ as Nesterov's adaptive strategy without low recovering rate nor slow convergence rate. Algorithm 1 provided details of PUG with $N = 2$. In Algorithm 1, the Lipschitz constant estimation from last iteration, $\tilde{L}_k$, is first divided by 2 to be the initial guess $\tilde{L}$, which is the same as the Nesterov's adaptive method. We point out that,

$$\mathcal{U}(\epsilon) \to \infty \quad \text{as} \quad \epsilon \to 0.$$

Therefore, the convergence of FISTA is guaranteed with PUG, even in the extreme case that $L \leq \mathcal{U}(\epsilon)$ with $\epsilon \approx 0$.

In the case where computing $\mu_{\max}$ is impractical, it could be bounded by

$$\mu_{\max} = \lambda_{\max}(\mathbf{\Sigma}) = \|\mathbf{\Sigma}^{\frac{1}{2}}\|^2 \leq \text{trace}(\mathbf{\Sigma}) = \sum_{i=1}^{n} \text{Var}(\xi_i). \tag{26}$$

With the assumption that ξ_i's have zero mean and unit variance, $\mu_{\max} \leq n$. For $\mathbf{A}$ that does not satisfy these assumptions due to different normalization process of the data, (26) could be used to bound $\mu_{\max}$. For the R in (25), one could use $c^2 n$ as in Corollary 12, or a tighter estimation would be

$$R = \sum_{i=1}^{n} \max_k a_{i,k}^2, \tag{27}$$

since $\lambda_{\max}\left[\boldsymbol{\xi}\boldsymbol{\xi}^T\right] = \|\boldsymbol{\xi}\boldsymbol{\xi}^T\| = \boldsymbol{\xi}^T\boldsymbol{\xi} = \sum_{i=1}^n \xi_i^2$.

4.3 Convergence Bounds: Regular Strategies v.s. PUG

Different stepsize strategies would lead to different convergence rates even for the same algorithm. Since PUG is based on the probabilistic upper bound $\mathcal{U}(\epsilon)$ in (25), it leads to a probabilistic convergence of FISTA. In particular,

$$F(\mathbf{x}_k) - F(\mathbf{x}_\star) \leq \frac{2M}{(k+1)^2}\left(2\gamma\mu_{\max} - \frac{\gamma R}{m}\log\left(\frac{\epsilon}{n}\right)\right), \tag{28}$$

with probability at least $1-\epsilon$. Equation (28) holds with probability at least $1-\epsilon$ because of equation (25), which holds for all iterations in FISTA. In particular, once the instance (matrix $\mathbf{A}$) is fixed, then we have know $L \leq \mathcal{U}(\epsilon)$, with probability at least $1-\epsilon$. If the probabilistic upper bound holds, then $\mathcal{U}(\epsilon)$ is the worst Lipschitz constant estimation computed by PUG in all iterations, and so (27) holds. Therefore, the above result could be obtained using the same argument as in the proof of convergence in [1].

When using regular stepsize strategies, FISTA results in convergence rates that is in the form of (21) with different η's ($\eta > 1$). For backtracking strategy, η would be an user-specified parameter. It is clear from (21) that convergence is better when η is close to 1. However, it would take more iterations and more function evaluations to find a satisfying stepsize, and these costs are not captured in (21). In the case of Nesterov's adaptive strategy [12], $\eta = 2$. Using the same analysis as in Section 3.2, L should be replaced with the upper bound in (22) for the average case, or $\mathcal{U}(\epsilon)$ in (25) for the probabilistic case. For the probabilistic case, those convergences are in the same order as in the case of using PUG, as shown in (28).

Therefore, PUG is competitive compared to other stepsize strategies in the probabilistic case. The strength of PUG comes from the fact that it is adaptive with strong theoretical guarantee that with high probability, $\tilde{L}_k$ will quickly be accepted at each iteration.

4.4 Mini-batch Algorithms and Block-coordinate Algorithms

For mini-batch algorithms, each iteration is performed using only a subset of the whole training set. Therefore, in each iteration, we consider a matrix

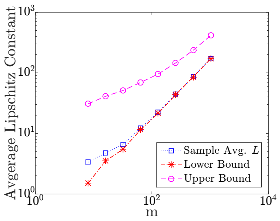

Fig. 1: Case I, $m = n$

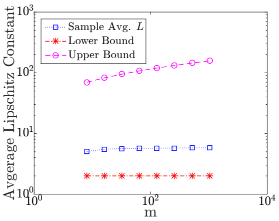

Fig. 2: Case II, $2m = n$

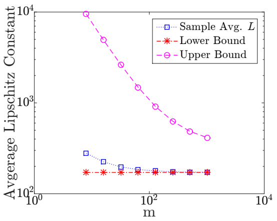

Fig. 3: Case III, $n = 1024$

that contains the corresponding subset. This matrix is a submatrix of $\mathbf{A}$ with the same structure, and therefore it is also a random matrix with smaller size $\bar{m}$-by-n, where $\bar{m} < m$. Using the existing results, we can conclude that the associated $\mathcal{U}(\epsilon)$ in each iteration would be larger than those in full-batch algorithms. As a result, the guaranteed stepsize for mini-batch algorithms tends to be smaller than full-batch algorithms.

On the other hand, block-coordinate algorithms do not update all dimensions at once in each iteration. Rather, a subset of dimensions will be selected to perform the update. In such setting, we only consider the variables (columns of $\mathbf{A}$) that are associated with the selected coordinates. We should consider a submatrix that is formed by columns of $\mathbf{A}$. This submatrix itself is also a random matrix with smaller size m-by-$\bar{n}$, where $\bar{n} < n$. Using the existing results, the guaranteed stepsize for block-coordinate algorithms tends to be larger.

Thus, with minor modifications PUG can be applied to mini-batch and block-coordinate algorithms.

5 Numerical Experiments

In the first part of this section, we will apply the bounds from Section 3 to illustrate the relationship between different parameters and L. Then, we will perform the PUG on two regression examples. The datasets used for the two regression examples can be found at *https://www.csie.ntu.edu.tw/˜cjlin/libsvmtools/datasets*.

5.1 Numerical Simulations for Average L

We consider three cases, and in each case we simulate $\mathbf{A}$'s in different dimension m's and n's. Each configuration is simulated with 1000 instances, and we study the sample average behaviors of L.

In the first case, we consider the most complicated situation and create random vector such that its entries are not identical nor independent. We use a mixture of three types of random variables (exponential, uniform, and multivariate normal) to construct the matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$. The rows of $\mathbf{A}$ are

	Backtracking	Nesterov	PUG
T	1.00x	0.32x	**0.22x**
nIter	1.00x	0.22x	**0.18x**
nFunEva	1.00x	0.29x	**0.21x**
Avg. $\tilde{L}$	1.00x	**0.16x**	0.24x

Table 1: Gisette

independent samples of $\boldsymbol{\xi}^T = (\xi_1, \xi_2, \cdots, \xi_n)$. We divide $\mathbf{A}$ into three parts with n_1, n_2, and n_3 columns. Note that $n_1 = n_2 = n_3 = n/3$ up to rounding errors. We assign $\boldsymbol{\xi}$ with the elements where

$$\xi_j \sim \begin{cases} \mathit{Exp}(1) - 1 & \text{if } j \leq n_1, \\ \mathcal{U}(-\sqrt{3}, \sqrt{3}) & \text{if } n_1 < j \leq n_1 + n_2, \end{cases} \tag{29}$$

and $(\xi_{n_1+n_2+1}, \xi_{n_1+n_2+2}, \cdots, \xi_n) \sim \mathcal{N}(\mathbf{0}_{n_3 \times 1}, \hat{\boldsymbol{\Sigma}})$. $\hat{\boldsymbol{\Sigma}}$ is a $n_3 \times n_3$ matrix with 1 on the diagonal and 0.5 otherwise. $\xi_1, \xi_2, \cdots, \xi_{n_1+n_2}$ are independent.

The scaling factors of the uniform distribution and exponential distribution are used to normalize the uniform random variables ξ_j such that $\mathbb{E}[\xi_j] = 0$, and $\mathbb{E}[\xi_j^2] = 1$. Some entries of $\mathbf{A}$ are normally distributed or exponentially distributed, and we approximate the upper bound of the entries with $c = 3$. From statistics, we know that with very high probability, this approximation is valid.

In Figure 1, we plot the sample average Lipschitz constant over 1000 instances. As expected, the expected Lipschitz constant is “trapped” between its lower and upper bound. We can see that the expected L increases when m and n increases with the ratio n/m is fixed. This phenomenon is due the fact that $\mu_{\max} = \lambda_{\max}(\boldsymbol{\Sigma})$ increases as n increases.

To further illustrate this, we consider the second case. The setting in this case is the same as the first case except that we replace $\hat{\boldsymbol{\Sigma}}$ with $\mathbf{I}_n$. So, all the variables are linearly independent. In the case, $\mu_{\max} = 1$ regardless the size of the $\mathbf{A}$. The ratio $n/m = 2$ in this example. From Figure 2, the sample average L does not increase rapidly as the size of $\mathbf{A}$ increases. These results match with the bound (22).

In the last case, we investigate the effect of the ratio n/m. The setting is same as the first case, but we keep $n = 1024$ and test with different m's. From Figure 3, the sample average L decreases as m increases. This result suggests that a large dataset is favorable in terms of complexity, especially for large-scale (large n) ERM problems.

5.2 Regularized Logistics Regression

We implement FISTA with three different stepsize strategies (i) the regular backtracking stepsize strategy, (ii) the Nesterov's adaptive stepsize strategy,

	Backtracking	Nesterov	PUG
T	1.00x	1.02x	**0.99x**
nIter	1.00x	0.69x	**0.68x**
nFunEva	1.00x	0.92x	**0.90x**
Avg. $\tilde{L}$	1.00x	0.54x	**0.52x**

Table 2: YearPredictionMSDt

and (iii) the proposed adaptive stepsize strategy PUG. We compare the three strategies with an example in a ℓ_1 regularized logistic regression problem, in which we solve the convex optimization problem

$$\min_{\mathbf{x}\in\mathbb{R}^n} \frac{1}{m}\sum_{i=1}^{m} \log(1+\exp(-b_i\mathbf{x}^T\mathbf{a}_i)) + \omega\|\mathbf{x}\|_1.$$

We use the dataset *gisette* for $\mathbf{A}$ and $\mathbf{b}$. *Gisette* is a handwritten digits dataset from the NIPS 2003 feature selection challenge. The matrix $\mathbf{A}$ is a 6000×5000 dense matrix, and so we have $n = 5000$ and $m = 6000$. The parameter ω is chosen to be the same as [9,21]. We chose $\tilde{L}_0 = 1$ for all three stepsize strategies. For backtracking stepsize strategy, we chose $\eta = 1.5$.

We compare our proposed probabilitic bound and the deterministic upper bound $\bar{L}$ using $\|\mathbf{A}\|^2 \le \text{trace}(\mathbf{A}^T\mathbf{A})$. We estimate $\mu_{\max} = 1289.415$ and $R = 4955$ using equation (26) and (27), respectively. We thus obtain our probabilistic bound $\mathcal{U}(0.1) = 646.941$, which is less than the deterministic upper bound $\bar{L} = 1163.345$.

Table 1 shows the performances of three stepsize strategies. T is the scaled computational time, nIter is the scaled number of iterations, nFunEva is the scaled number of function evaluations, and Avg. $\tilde{L}$ is the average of $\tilde{L}$ used. This result encourages the two adaptive stepsize strategies as the number of iterations needed and the computational time are significantly smaller compared to the regular backtracking algorithm. This is due to the fact that $\tilde{L}$ could be a lot smaller than the Lipschitz constant L in this example, and so the two adaptive strategies provide more efficient update for FISTA. As shown in Table 1, even though Nesterov's strategy yields smaller $\tilde{L}$ on average, it leads to higher number of function evaluations and so it takes more time than PUG.

5.3 Regularized Linear Regression

We also compare the three strategies with an example in a ℓ_1 regularized linear regression problem, a.k.a LASSO, in which we solve the convex optimization problem

$$\min_{\mathbf{x}\in\mathbb{R}^n} \frac{1}{2m}\sum_{i=1}^{m} (\mathbf{x}^T\mathbf{a}_i - b_i)^2 + \omega\|\mathbf{x}\|_1.$$

We use the dataset *YearPredictionMSDt* (testing dataset) for $\mathbf{A}$ and $\mathbf{b}$. *YearPredictionMSDt* has matrix $\mathbf{A}$ is a 51630×90 dense matrix, and so we have $n = 90$ and $m = 51630$. The parameter ω is chosen to be 10^{-6}. We chose $\tilde{L}_0 = 1$ for all three stepsize strategies. For backtracking stepsize strategy, we chose $\eta = 1.5$.

We compare our proposed probabilitic bound and the deterministic upper bound $\bar{L}$ using $\|\mathbf{A}\|^2 \leq \mathrm{trace}(\mathbf{A}^T\mathbf{A})$. We estimate $\mu_{\max} = 9.603 \times 10^6$ and $R = 4.644 \times 10^9$ using equation (26) and (27), respectively. We thus obtain our probabilistic bound $\mathcal{U}(0.1) = 1.982 \times 10^7$, which is less than the deterministic upper bound $\bar{L} = 2.495 \times 10^7$.

Table 2 shows the performance of three stepsize strategies, and the structure is same as Table 1. Unlike *Gisette*, adaptive strategies failed to provide small $\tilde{L}$ compared to L. Also, since n is very small, the cost of function evaluation is very cheap compared to *Gisette*. Therefore, both adaptive strategies do not outperform the backtracking strategy in this example. However, one can see that both adaptive strategies yielded to reduction in terms of the number of function evaluations. Therefore, one could expect they outperform backtracking strategy for larger/difficult instances.

6 Conclusions and Perspectives

The analytical results in this paper show the relationship between the Lipschitz constant and the training set of an ERM problem. These results provide insightful information about the complexity of ERM problems, as well as opening up opportunities for new stepsize strategies for optimization problems.

One interesting extension could be to apply the same approach to different machine learning models, such as neural networks, deep learning, etc.

References

1. Beck, A., Teboulle, M.: A fast iterative shrinkage-thresholding algorithm for linear inverse problems. SIAM Journal on Imaging Sciences **2**(1), 183–202 (2009). DOI 10.1137/080716542. URL http://dx.doi.org/10.1137/080716542
2. Beck, A., Tetruashvili, L.: On the convergence of block coordinate descent type methods. SIAM Journal on Optimization **23**(4), 2037–2060 (2013). DOI 10.1137/120887679. URL http://dx.doi.org/10.1137/120887679
3. Belloni, A., Chernozhukov, V., Wang, L.: Pivotal estimation via square-root Lasso in nonparametric regression. The Annals of Statistics **42**(2), 757–788 (2014). DOI 10.1214/14-AOS1204. URL http://dx.doi.org/10.1214/14-AOS1204
4. Burke, J.V., Ferris, M.C.: A Gauss-Newton method for convex composite optimization. Mathematical Programming **71**(2, Ser. A), 179–194 (1995). DOI 10.1007/BF01585997. URL http://dx.doi.org/10.1007/BF01585997
5. Candès, E.J., Romberg, J.K., Tao, T.: Stable signal recovery from incomplete and inaccurate measurements. Communications on Pure and Applied Mathematics **59**(8), 1207–1223 (2006). DOI 10.1002/cpa.20124. URL http://dx.doi.org/10.1002/cpa.20124
6. Donoho, D.L.: Compressed sensing. IEEE Transactions on Information Theory **52**(4), 1289–1306 (2006). DOI 10.1109/TIT.2006.871582. URL http://dx.doi.org/10.1109/TIT.2006.871582

7. Gonzaga, C.C., Karas, E.W.: Fine tuning Nesterov's steepest descent algorithm for differentiable convex programming. Mathematical Programming **138**(1-2, Ser. A), 141–166 (2013). DOI 10.1007/s10107-012-0541-z. URL http://dx.doi.org/10.1007/s10107-012-0541-z
8. Koltchinskii, V., Mendelson, S.: Bounding the smallest singular value of a random matrix without concentration. Int. Math. Res. Not. IMRN (23), 12,991–13,008 (2015)
9. Lee, J.D., Sun, Y., Saunders, M.A.: Proximal Newton-type methods for minimizing composite functions. SIAM Journal on Optimization **24**(3), 1420–1443 (2014). DOI 10.1137/130921428. URL http://dx.doi.org/10.1137/130921428
10. Nesterov, Y.: Introductory lectures on convex optimization, *Applied Optimization*, vol. 87. Kluwer Academic Publishers, Boston, MA (2004). DOI 10.1007/978-1-4419-8853-9. URL http://dx.doi.org/10.1007/978-1-4419-8853-9. A basic course
11. Nesterov, Y.: Gradient methods for minimizing composite functions. Mathematical Programming **140**(1, Ser. B), 125–161 (2013). DOI 10.1007/s10107-012-0629-5. URL http://dx.doi.org/10.1007/s10107-012-0629-5
12. Nesterov, Y.: Universal gradient methods for convex optimization problems. Mathematical Programming **152**(1-2, Ser. A), 381–404 (2015). DOI 10.1007/s10107-014-0790-0. URL http://dx.doi.org/10.1007/s10107-014-0790-0
13. Qin, Z., Scheinberg, K., Goldfarb, D.: Efficient block-coordinate descent algorithms for the group Lasso. Mathematical Programming Computation **5**(2), 143–169 (2013). DOI 10.1007/s12532-013-0051-x. URL http://dx.doi.org/10.1007/s12532-013-0051-x
14. Qu, Z., Richtarik, P.: Coordinate descent with arbitrary sampling ii: expected separable overapproximation. arXiv:1412.8063 (2014)
15. Rudelson, M., Vershynin, R.: Non-asymptotic theory of random matrices: extreme singular values. In: Proceedings of the International Congress of Mathematicians. Volume III, pp. 1576–1602. Hindustan Book Agency, New Delhi (2010)
16. Saha, A., Tewari, A.: On the nonasymptotic convergence of cyclic coordinate descent methods. SIAM Journal on Optimization **23**(1), 576–601 (2013). DOI 10.1137/110840054. URL http://dx.doi.org/10.1137/110840054
17. Shalev-Shwartz, S., Ben-David, S.: Understanding Machine Learning: From Theory to Algorithms. Understanding Machine Learning: From Theory to Algorithms. Cambridge University Press (2014). URL https://books.google.co.uk/books?id=ttJkAwAAQBAJ
18. Sun, T., Zhang, C.H.: Sparse matrix inversion with scaled lasso. Journal of Machine Learning Research **14**, 3385–3418 (2013)
19. Tropp, J.A.: User-friendly tail bounds for sums of random matrices. Foundations of Computational Mathematics **12**(4), 389–434 (2012). DOI 10.1007/s10208-011-9099-z. URL http://dx.doi.org/10.1007/s10208-011-9099-z
20. Yamakawa, E., Fukushima, M., Ibaraki, T.: An efficient trust region algorithm for minimizing nondifferentiable composite functions. SIAM Journal on Scientific and Statistical Computing **10**(3), 562–580 (1989). DOI 10.1137/0910036. URL http://dx.doi.org/10.1137/0910036
21. Yuan, G.X., Ho, C.H., Lin, C.J.: An improved glmnet for l1-regularized logistic regression. Journal of Machine Learning Research **13**(1), 1999–2030 (2012). URL http://dl.acm.org/citation.cfm?id=2503308.2343708