Fast Stochastic AUC Maximization with $O(1/n)$-Convergence Rate

Mingrui Liu [1] **Xiaoxuan Zhang** [1] **Zaiyi Chen** [2] **Xiaoyu Wang** [3] **Tianbao Yang** [1]

Abstract

In this paper, we consider statistical learning with AUC (area under ROC curve) maximization in the classical stochastic setting where one random data drawn from an unknown distribution is revealed at each iteration for updating the model. Although consistent convex surrogate losses for AUC maximization have been proposed to make the problem tractable, it remains an challenging problem to design fast optimization algorithms in the classical stochastic setting since the convex surrogate loss depends on random pairs of examples from positive and negative classes. Building on a saddle point formulation for a consistent square loss, this paper proposes a novel stochastic algorithm to improve the standard $O(1/\sqrt{n})$ convergence rate to $\widetilde{O}(1/n)$ convergence rate without strong convexity assumption or any favorable statistical assumptions (e.g., low noise), where n is the number of random samples. To the best of our knowledge, this is the first stochastic algorithm for AUC maximization with a statistical convergence rate as fast as $O(1/n)$ up to a logarithmic factor. Extensive experiments on eight large-scale benchmark data sets demonstrate the superior performance of the proposed algorithm comparing with existing stochastic or online algorithms for AUC maximization.

1. Introduction

Area under ROC curve (AUC) (Metz, 1978; Hanley & McNeil, 1982; 1983) is a commonly used metric for evaluating the performance of a classifier. ROC (receiver operating characteristic) curve is the "`true positive rate - false positive rate`" curve, and AUC represents the probability that examples in positive class are scored higher than those in negative class (Hanley & McNeil, 1982). Compared with misclassification rate, AUC is more favorable in the applications with imbalanced datasets and in which the real-valued classifier is used for ranking. However, the algorithms designed to minimize the misclassification error rate may not lead to maximization of AUC (Cortes & Mohri, 2004).

In a batch learning setting where all training data is given beforehand, one may formulate AUC maximization as a convex empirical optimization problem based on the training data using a convex surrogate loss. The resulting optimization problem can be solved by employing existing algorithms (Herschtal & Raskutti, 2004; Cortes & Mohri, 2004; Ferri et al., 2011). However, such an approach has two limitations: (i) it is not applicable to many applications in which a random data is received in a sequential manner (e.g., online advertisement); (ii) little is known about the statistical convergence rate of the empirical maximizer to the true maximizer due to the i.i.d assumption of pairwise data does not hold. Therefore, a stochastic algorithm with a provable convergence rate in **the classical stochastic setting** for minimizing an expected convex surrogate loss for AUC maximization is desirable for addressing these two limitations. It is also useful for tackling large-scale data by one pass of training data.

Nevertheless, the pairwise nature in the definition of AUC makes it challenging to design algorithms suitable for the classical stochastic setting. To address this challenge, several online and stochastic algorithms have been developed based on convex surrogate loss (Zhao et al., 2011; Gao et al., 2013; Ying et al., 2016). An interesting observation in (Ying et al., 2016) is that the AUC maximization using a consistent square loss is equivalent to a stochastic min-max saddle point problem, for which a primal-dual style of stochastic gradient algorithm can be employed to yield an $\widetilde{O}(1/\sqrt{n})$ convergence rate for minimizing an expected square loss, where n is the number of samples. How to improve the convergence rate of stochastic optimization of AUC remains an open problem though.

Fast rate such as $O(1/n)$ of stochastic algorithms has been established for expected convex risk minimization in literature (Hazan & Kale, 2011a; Srebro et al., 2010; Bach & Moulines, 2013). However, these studies either impose strong assumptions about the problem (e.g., strong convex-

[1]Department of Computer Science, The University of Iowa, IA 52242, USA [2]University of Science and Technology of China [3]Intellifusion. Correspondence to: Mingrui Liu <mingrui-liu@uiowa.edu>, Tianbao Yang <tianbao-yang@uiowa.edu>.

Proceedings of the 35^{th} International Conference on Machine Learning, Stockholm, Sweden, PMLR 80, 2018.

ity assumption, low-noise assumption, etc.), and/or their analysis is limited to certain settings that is not applicable to AUC maximization. Therefore, a stochastic algorithm with a provable fast rate as $O(1/n)$ without imposing strong assumptions should be considered as a significant contribution for AUC maximization.

The proposed algorithm referred to as FSAUC is a **F**ast **S**tochastic algorithm for true **AUC** maximization. It is based on the min-max saddle point formulation as observed in (Ying et al., 2016). However, different from (Ying et al., 2016), we develop a novel multi-stage scheme for running primal-dual stochastic gradient method with adaptively changing parameters and initial solutions for both primal and dual variables. The adaptive multi-stage scheme not only leverages the primal solution from previous stages but also utilizes the empirical estimation of feature vectors of examples from both positive and negative classes for initializing the dual variables. The convergence analysis of the proposed algorithm hinges on a quadratic growth property of a reformulated AUC objective and a novel synthesis of the adaptive scheme and the convergence result of primal-dual stochastic gradient method. To summarize, the major contributions of this work are:

- We propose a novel fast stochastic algorithm that can be run in the classical stochastic setting for maximizing AUC with a known number of total samples n. We establish an $\widetilde{O}(1/n)$ [1] convergence rate for the proposed algorithm, where n is the total number of samples. This is the first $\widetilde{O}(1/n)$ convergence result for stochastic AUC optimization.
- We evaluate the proposed algorithm on eight large-scale benchmark datasets. The results show that our algorithm significantly outperforms two state-of-the-art stochastic/online AUC methods, namely SOLAM algorithm (Ying et al., 2016) and OPAUC algorithm (Gao et al., 2013).

2. Related Work

(Zhao et al., 2011) is probably the first work on online maximization of AUC. In order not to store all received positive and negative examples, they proposed to use reservoir sampling to maintain a buffer of positive and negative examples. At each iteration, they run gradient update based on an online empirical version of AUC defined on the buffered data to update the models. A regret bound was established with the cost function at each iteration t defined based on the example received at the t-th iteration and all data in the buffer. Even with the optimal buffer size, their regret bound is worse than $\sqrt{n}$. Gao et al. (2013) proposed another online algorithm without resorting to the reservoir sampling. Their algorithm hinges on the observation that if a consistent square loss is used, the gradient of an online empirical version of AUC can be computed based on first and second order statistics of received data. Hence, their algorithm needs to maintain a covariance matrix of received examples, which renders it not practical for high-dimensional data. Although a randomized version is proposed for maintaining a low rank version of the covariance matrix, it still has considerable computational overhead. In terms of guarantee, they established a regret bound in the order of $\sqrt{T}$ for general case and a possible $O(1)$ regret bound under the low-noise condition. However, these regret bounds do not directly imply a convergence rate for statistical AUC maximization. The reason is that the data that define each cost function are not independent. The stochastic algorithm proposed in a recent work (Ying et al., 2016) is the first with a convergence guarantee for statistical AUC optimization and is also the first one without storing any historical examples or their covariance matrix. As aforementioned, their algorithm is based on a novel min-max saddle point formulation of AUC maximization and has a convergence rate of $\widetilde{O}(1/\sqrt{n})$.

Fast rate of stochastic optimization such as $O(1/n)$ has been studied for standard classification and regression problems under some conditions. For example, Hazan & Kale (2011a) proposed a method with an $O(1/n)$ convergence rate under a (weak) strong convexity assumption. Their algorithm requires knowing the strong convexity parameter. For statistical learning problems, Srebro et al. (2010) established an $O(1/n)$ convergence rate for smooth loss functions under low-noise assumption (e.g., the optimal risk value is close to zero). However, their algorithm requires knowing a good estimation of the optimal risk value. Bach & Moulines (2013) established the first $O(1/n)$ convergence rate of stochastic algorithms for minimizing expected square loss for regression and expected logistic loss for classification without the strong convexity assumption. However, all these algorithms and analysis are not applicable to the AUC maximization in the classical stochastic setting.

Finally, we note that the multi-stage scheme of the proposed algorithm is similar to that developed in (Hazan & Kale, 2011b; Ghadimi & Lan, 2013; Juditsky et al., 2014; Xu et al., 2017) for minimizing strongly or uniformly convex functions or problems with error bound conditions. However, the differences between the proposed work and these works are that (i) our development is based on primal dual stochastic method for solving a stochastic saddle point problem of AUC maximization with particular treatment of dual variables; while their algorithms are for solving stochastic convex minimization problems; (ii) we do not require strong convexity assumption with known parameters as in (Hazan & Kale, 2011b; Ghadimi & Lan, 2013); (iii)

[1] The $\widetilde{O}(\cdot)$ notation hides logarithmic factors.

we do not assume uniform strong convexity assumption as in (Juditsky et al., 2014). Instead, we prove a quadratic growth condition for the studied problem that is a special case of error bound condition studied in (Xu et al., 2017). However, the stochastic algorithms proposed in (Xu et al., 2017) require a target accuracy level and cannot be applied to one pass learning setting for large-scale data.

3. Algorithm and Main Result

3.1. Preliminaries and Notations

Let $\mathbf{z} = (\mathbf{x}, y) \sim \mathcal{P}$ denote a random data following an unknown distribution $\mathcal{P}$, where $\mathbf{x} \in \mathcal{X} \subseteq \mathbb{R}^d$ represents the feature vector and $y \in \{1, -1\}$ represents the label. Denote by $\mathcal{Z} = \mathcal{X} \times \{1, -1\}$ and by $p = \Pr(y = 1) = \mathbb{E}_y[\mathbb{I}_{[y=1]}]$, where $\mathbb{I}$ is an indicator function. We assume that $\sup_{\mathbf{x}\in\mathcal{X}} \|\mathbf{x}\|_2 \leq \kappa$. Let $\bar{\mathbf{x}} \in \mathbb{R}^{d+2}$ denote an augmented feature vector with the last two components being 0, and let $\mathcal{B}(x_0, r) = \{x : \|x - x_0\|_2 \leq r\}$ be an ℓ_2-ball centered at x_0 with a radius r.

Given a score function $h : \mathbb{R}^d \to \mathbb{R}$, the AUC at the population level (referred to as the true AUC in this paper) is defined as:

$$\text{AUC}(h) = \Pr(h(\mathbf{x}) \geq h(\mathbf{x}')|y = 1, y' = -1),$$

where $\mathbf{z} = (\mathbf{x}, y)$ and $\mathbf{z}' = (\mathbf{x}', y)$ are a pair of random data. The AUC maximization problem is to find an optimal score function in a hypothesis class such that AUC is maximized. Since the problem is non-convex, it is usually solved by using consistent convex surrogate loss. A common choice used by previous studies (Ying et al., 2016; Gao et al., 2013) is the square loss. In this paper, we consider learning a linear function $h(\mathbf{x}) = \mathbf{w}^\top \mathbf{x}$ to maximize the AUC using a square loss, i.e.,

$$\min_{\mathbf{w}\in\mathbb{R}^d} L(\mathbf{w}) \triangleq \mathbb{E}_{\mathbf{z},\mathbf{z}'}[(1 - \mathbf{w}^\top(\mathbf{x} - \mathbf{x}'))^2|y = 1, y' = -1]. \tag{1}$$

Since the loss function depends on a random pair of data making it difficult to handle in the classical stochastic setting, a solution is to cast the above problem into an equivalent saddle point problem (Ying et al., 2016):

$$\min_{\substack{\mathbf{w}\in\mathbb{R}^d \\ (a,b)\in\mathbb{R}^2}} \max_{\alpha\in\mathbb{R}} \{f(\mathbf{w}, a, b, \alpha) := \mathbb{E}_{\mathbf{z}}[F(\mathbf{w}, a, b, \alpha; \mathbf{z})]\},$$

where

$$\begin{aligned} F(\mathbf{w}, a, b, \alpha; \mathbf{z}) &= (1 - p)(\mathbf{w}^\top \mathbf{x} - a)^2 \mathbb{I}_{[y=1]} \\ &+ p(\mathbf{w}^\top \mathbf{x} - b)^2 \mathbb{I}_{[y=-1]} - p(1 - p)\alpha^2 \\ &+ 2(1 + \alpha)(p\mathbf{w}^\top \mathbf{x} \mathbb{I}_{[y=-1]} - (1 - p)\mathbf{w}^\top \mathbf{x} \mathbb{I}_{[y=1]}). \end{aligned}$$

Assuming the optimal solution $\mathbf{w}_*$ sits in a bounded domain such that $\|\mathbf{w}_*\|_1 \leq R$, we can restrict the primal and dual variables to constrained domains $\Omega_1 = \{(\mathbf{w}, a, b) : \|\mathbf{w}\|_1 \leq R, |a| \leq R\kappa, |b| \leq R\kappa\}, \Omega_2 = \{\alpha \in \mathbb{R} : |\alpha| \leq 2R\kappa\}$, i.e.,

$$\min_{(\mathbf{w},a,b)\in\Omega_1} \max_{\alpha\in\Omega_2} \{f(\mathbf{w}, a, b, \alpha) := \mathbb{E}_{\mathbf{z}}[F(\mathbf{w}, a, b, \alpha; \mathbf{z})]\}. \tag{2}$$

Please note that adding the ℓ_1-ball constraint on $\mathbf{w}$ does not restrict the performance of the learned model because (i) if R is large enough such that the optimal solution $\mathbf{w}_*$ to (1) satisfies $\|\mathbf{w}_*\|_1 \leq R$, then $\mathbf{w}_*$ is also the optimal solution to (2) [2]; (ii) for a finite number of samples, the constraint can serve as regularization for improving the generalization performance. We use ℓ_1-ball constraint because it allows us to show that the proposed stochastic optimization algorithm has an convergence rate of $\widetilde{O}(1/n)$.

Next, we will present the proposed algorithm and its convergence results. We will also provide a convergence analysis on a core component of the proposed algorithm.

3.2. Algorithm and its Convergence

The proposed algorithm is presented in Algorithm 1, which is referred to as FSAUC. The parameteres R_0, G, D_0, β_0 will be explained shortly. The algorithm divides the updates into m stages and each stage has $\lfloor n/m \rfloor$ updates. Each stage of FSAUC runs a primal dual style stochastic gradient (PDSG) method outlined in Algorithm 2, which is similar to that in (Ying et al., 2016) except for three differences: (i) the step size is given as a constant for each call of PDSG; (ii) each update of the primal variable $\mathbf{v} = (\mathbf{w}^\top, a, b)^\top$ and the dual variable α is projected into an intersection of their constrained domain and an ℓ_2 ball centered at the initial solution $\mathbf{v}_1$, α_1; (iii) upon receiving a random data $\mathbf{z}_t$, the variables $\widehat{A}_\pm, T_\pm, \widehat{p}$ are updated according to Algorithm 3, where $T_\pm$ represent the number of positive/negative examples received so far; $\widehat{A}_\pm$ represents the cumulative (augmented) feature vector for the positive class and negative class; and $\widehat{p}$ represents the estimated positive ratio based on the received examples. The parameters for each call of PDSG is adaptively changing, including the initial solutions $\mathbf{v}_1, \alpha_1$, the radius of ℓ_2 balls for the primal variables and the dual variables, and the step size η.

The initial parameter R_0 is a bound of Euclidean distance from $\widehat{\mathbf{v}}_0$ to the optimal set of (2) in terms of $\mathbf{v}$. The initial parameter D_0 is for setting a constrained domain of the dual variable. The initial parameter β_0 is for setting the initial step size. The parameter G is an upper bound of Euclidean norms of $G(\mathbf{u}, \mathbf{z}), \widehat{G}_t(\mathbf{u}, \mathbf{z}_t), g(\mathbf{u})$ for any $\mathbf{u} \in \Omega_1 \times \Omega_2$ and $\mathbf{z}, \mathbf{z}_t \in \mathcal{Z}$, which are defined in (3). The function $\hat{F}_t$ at the line 5 and line 6 of the Algorithm 2 given

[2] This can be shown that if $\|\mathbf{w}_*\|_1 \leq 1$, then the optimal solutions to a, b, α satisfy the prescribed bounds in light of their closed-form solutions depending on $\mathbf{w}_*$.

Algorithm 1 FSAUC

1: Set $m = \lfloor \frac{1}{2}\log_2 \frac{2n}{\log_2 n} \rfloor - 1$, $n_0 = \lfloor n/m \rfloor$, $R_0 = 2\sqrt{1+2\kappa^2}R$, $G > \max((1+4\kappa)\kappa(R+1), 2\kappa(2R+1+2R\kappa), 2\kappa(4\kappa R + 11R + 1))$, $\beta_0 = (1+8\kappa^2)$, $D_0 = 2\sqrt{2}\kappa R_0$
2: Initialize $\widehat{\mathbf{v}}_0 = 0 \in \mathbb{R}^{d+2}$, $\widehat{\alpha}_0 = 0$,
3: **for** $k = 1, \ldots, m$ **do**
4: Set $\eta_k = \frac{\sqrt{\beta_{k-1}}}{\sqrt{3n_0 G}} R_{k-1}$
5: Call PDSG to obtain $(\widehat{\mathbf{v}}_k, \widehat{\alpha}_k, \beta_k, R_k, D_k) = \text{PDSG}(\widehat{\mathbf{v}}_{k-1}, \widehat{\alpha}_{k-1}, R_{k-1}, D_{k-1}, n_0, \eta_k)$
6: **end for**
7: **return** $\widehat{\mathbf{v}}_m$

by

$$\begin{aligned}\widehat{F}_t(\mathbf{v}, \alpha, \mathbf{z}_t) &= (1-\widehat{p})(\mathbf{w}^\top \mathbf{x}_t - a)^2 \mathbb{I}_{[y_t=1]} \\ &+ \widehat{p}(\mathbf{w}^\top \mathbf{x}_t - b)^2 \mathbb{I}_{[y_t=-1]} - \widehat{p}(1-\widehat{p})\alpha^2 \\ &+ 2(1+\alpha)(\widehat{p}\mathbf{w}^\top \mathbf{x}_t \mathbb{I}_{[y_t=-1]} - (1-\widehat{p})\mathbf{w}^\top \mathbf{x} \mathbb{I}_{[y_t=1]})\end{aligned}$$

is an estimation of $F(\mathbf{v}, \alpha, \mathbf{z}_t)$ using the current estimation $\widehat{p}$ of p. It is worth mentioning that the line 5 and line 6 of the Algorithm 2 need to execute a projection onto the intersection of two convex sets. We can use the alternating projection algorithm to generate a sequence which can linearly converges to the desired point. Actually it is easy to show that the two sets in our case are boundedly linearly regular (Definition 3.11 of (Bauschke & Borwein, 1993)) and the linear convergence is guaranteed by the Corollary 3.14 of (Bauschke & Borwein, 1993).

Finally, the convergence rate of FSAUC is presented in the following theorem.

Theorem 1. *Given $\delta \in (0,1)$, assume n is sufficiently large such that $n > \max(100, m\frac{32\ln(\frac{12}{\delta})}{(\min(p,1-p))^2})$. Then with probability at least $1-\delta$,*

$$\max_{\alpha\in\Omega_2} f(\widehat{\mathbf{v}}_m, \alpha) - \min_{\mathbf{v}\in\Omega_1}\max_{\alpha\in\Omega_2} f(\mathbf{v}, \alpha) \leq \tilde{O}\left(\frac{\ln(\frac{1}{\delta})}{n}\right),$$

where $\widetilde{O}(\cdot)$ suppresses logarithmic factor of $\log(n)$ and some constants of the problem independent of n.

Remark: In practice, we can maintain global variables $\widehat{A}_+, \widehat{A}_-, \widehat{p}$ and use them for updating parameters and initializing dual variables. The local versions used in Algorithm 2 are just for simplicity of analysis.

The above theorem implies the convergence of AUC maximization in terms of the square loss.

Corollary 1. *Under the same condition as in Theorem 1,*

Algorithm 2 PDSG$(\mathbf{v}_1, \alpha_1, r, D, T, \eta)$

1: Initialize variables $\widehat{A}_+ \in \mathbb{R}^{d+2}, \widehat{A}_- \in \mathbb{R}^{d+2}$, $T_+, T_-, \widehat{p} \in \mathbb{R}$ as zeros
2: **for** $t = 1, \ldots, T$ **do**
3: Receive a sample $\mathbf{z}_t = (\mathbf{x}_t, y_t)$
4: Update $\widehat{A}_\pm, T_\pm, \widehat{p}$ using the data $\mathbf{z}_t$
5: $\mathbf{v}_{t+1} = \Pi_{\Omega_1 \cap \mathcal{B}(\mathbf{v}_1, r)}(\mathbf{v}_t - \eta \partial_{\mathbf{v}} \widehat{F}_t(\mathbf{v}_t, \alpha_t, \mathbf{z}_t))$
6: $\alpha_{t+1} = \Pi_{\Omega_2 \cap \mathcal{B}(\alpha_1, D)}(\alpha_t + \eta \partial_\alpha \widehat{F}_t(\mathbf{v}_t, \alpha_t, \mathbf{z}_t))$
7: **end for**
8: Compute $\bar{\mathbf{v}}_T = \frac{\sum_{t=1}^T \mathbf{v}_t}{T}$ and $\widehat{\alpha} = (\frac{\widehat{A}_-}{T_-} - \frac{\widehat{A}_+}{T_+})^\top \bar{\mathbf{v}}_T$
9: Let $r = r/2$
10: Update β, D according to Lemma 1
11: **return** $(\bar{\mathbf{v}}_T, \widehat{\alpha}, \beta, r, D)$

Algorithm 3 Update $\widehat{A}_\pm, T_\pm, \widehat{p}$ given a data $(\mathbf{x}_t, y_t)$

$\widehat{A}_+ = \widehat{A}_+ + \mathbb{I}_{[y_t=1]}\bar{\mathbf{x}}_t$
$\widehat{A}_- = \widehat{A}_- + \mathbb{I}_{[y_t=-1]}\bar{\mathbf{x}}_t$
$T_+ = T_+ + \mathbb{I}_{[y_t=1]}$
$T_- = T_- + \mathbb{I}_{[y_t=-1]}$
$\widehat{p} = T_+/(T_+ + T_-)$

with probability at least $1-\delta$,

$$L(\widehat{\mathbf{w}}_m) - \min_{\|\mathbf{w}\|_1 \leq R} L(\mathbf{w}) \leq \tilde{O}\left(\frac{\ln(\frac{1}{\delta})}{n}\right).$$

4. Convergence Analysis

This section is devoted to convergence analysis. Due to limitation of space, we present detailed analysis for a key lemma. More details about the proof of main Theorem can be found in the supplement.

Before analysis, we first give some notations that will be frequently used in this section.

$$\begin{aligned}\mathbf{v} &= (\mathbf{w}^\top, a, b)^\top \in \mathbb{R}^{d+2} \\ \mathbf{u} &= (\mathbf{v}^\top, \alpha)^\top \in \mathbb{R}^{d+3} \\ G(\mathbf{u}, \mathbf{z}) &= (\nabla_{\mathbf{v}} F(\mathbf{v}, \alpha; \mathbf{z}), -\nabla_\alpha F(\mathbf{v}, \alpha; \mathbf{z})) \\ \widehat{G}_t(\mathbf{u}, \mathbf{z}_t) &= (\nabla_{\mathbf{v}} \widehat{F}_t(\mathbf{v}, \alpha; \mathbf{z}_t), -\nabla_\alpha \widehat{F}_t(\mathbf{v}, \alpha; \mathbf{z}_t)) \\ g(\mathbf{u}) &= (\nabla_{\mathbf{v}} f(\mathbf{v}, \alpha), -\nabla_\alpha f(\mathbf{v}, \alpha))\end{aligned} \tag{3}$$

Lemma 1. *For each call of Algorithm 2, we update D, β according to*

$$D = 2\sqrt{2}\kappa r + \frac{4\sqrt{2}\kappa\left(2 + \sqrt{2\ln(\frac{12}{\delta})}\right)(1+2\kappa)R}{\sqrt{(\min(\widehat{p}, 1-\widehat{p})T - \sqrt{2T\ln(\frac{12}{\delta})})}},$$

and

$$\beta = (1+8\kappa^2) + \frac{32\kappa^2(1+2\kappa)^2\left(2+\sqrt{2\ln(\frac{12}{\delta})}\right)^2}{\min(\widehat{p},1-\widehat{p}) - \sqrt{2\ln(\frac{12}{\delta})/(T)}},$$

where $\widehat{p}$ is an estimation of p. Suppose $\|\mathbf{v}_1 - \mathbf{v}_*\|_2 \leq r$, where $\mathbf{v}_* \in \Omega_1$ is the optimal solution closest to $\mathbf{v}_1$, and $T > \max\left(\frac{R^2}{r^2}, \frac{32\ln(\frac{12}{\delta})}{(\min(p,1-p))^2}\right)$, then

$$\max_{\alpha\in\Omega_2} f(\bar{\mathbf{v}}_T,\alpha) - \min_{\mathbf{v}\in\Omega_1}\max_{\alpha\in\Omega_2} f(\mathbf{v},\alpha) \leq \frac{\left(2\sqrt{3\gamma_1} + \sqrt{\ln(\frac{6T}{\delta})\gamma_2}\right) rG}{\sqrt{T}}, \quad (4)$$

where $\gamma_1 = (1+8\kappa^2) + \frac{32\kappa^2(1+2\kappa)^2\left(2+\sqrt{2\ln(\frac{12}{\delta})}\right)^2}{\min(p,1-p)/2}$, $\gamma_2 = 16\left(1 + 2\sqrt{2}\kappa + \frac{8\kappa\left(2+\sqrt{2\ln(\frac{12}{\delta})}\right)(1+2\kappa)}{\sqrt{\min(p,1-p)/2}}\right)$.

Remark: When n is sufficiently large as stated in the main theorem, then $\sqrt{\log(2/\delta)/T} < \min(\widehat{p}, 1-\widehat{p})$ by Hoeffding's inequality since $\widehat{p}$ is estimated using n_0 examples. The above result implies that the proposed algorithm has at least an $O(1/\sqrt{n})$ convergence rate.

Based on the above lemma, Algorithm 1 uses a geometrically decreasing radius r in order to achieve a faster convergence. By further utilizing the following property we can prove the main theorem.

Lemma 2. *$f_1(\mathbf{v}) = \max_{\alpha\in\Omega_2} f(\mathbf{v},\alpha)$ restricted on the set Ω_1 satisfies a quadratic growth condition, i.e., for any $\mathbf{v}\in\Omega_1$, there exists $c>0$ such that*

$$\|\mathbf{v} - \mathbf{v}_*\|_2 \leq c(f_1(\mathbf{v}) - \min_{\mathbf{v}\in\Omega_1} f_1(\mathbf{v}))^{1/2},$$

where $\mathbf{v}_*$ is the optimal solution to $\min_{\mathbf{v}\in\Omega_1} f_1(\mathbf{v})$ that is closest to $\mathbf{v}$.

Remark: It is notable that the proposed algorithm FSAUC does not require the value of c that is difficult to compute.

4.1. Proof of Lemma 1

To prove this lemma, we need the following lemma, whose proof is included in the supplement.

Lemma 3. *Let $\widehat{A} = \widehat{A}_+/T_+ - \widehat{A}_-/T_-$ and and $A = ((\mathbb{E}(\mathbf{x}|y=-1) - \mathbb{E}(\mathbf{x}|y=1))^\top, 0, 0)^\top$. After the k-th call of Algorithm 2 with $k\geq 1$, with probability $1-\frac{\delta}{3}$ we have*

$$\|\widehat{A} - A\|_2 \leq \frac{4\kappa\left(2+\sqrt{2\ln(\frac{12}{\delta})}\right)}{\sqrt{\xi T}},$$

where $\xi \equiv \min(\widehat{p},1-\widehat{p}) - \sqrt{\frac{2\ln(\frac{12}{\delta})}{T}}$.

Proof of Lemma 1. By the setting of G, we have

$$\max_{t,\mathbf{u}_t,\mathbf{z}_t}\left(\|g(\mathbf{u}_t)\|_2, \|G(\mathbf{u}_t,\mathbf{z}_t)\|_2, \|\widehat{G}_t(\mathbf{u}_t,\mathbf{z}_t)\|_2\right) \leq G.$$

Define

$\alpha_{*,T} = \arg\max_\alpha f(\bar{\mathbf{v}}_T,\alpha) = \bar{\mathbf{w}}_T^\top[\mathbb{E}(\mathbf{x}|y=-1) - \mathbb{E}(\mathbf{x}|y=1)]$. Following standard analysis of primal-dual update (e.g., see inequality 17 in (Ying et al., 2016)), we have

$$\begin{aligned}
&\max_{\alpha\in\Omega_2} f(\bar{\mathbf{v}}_T,\alpha) - \min_{\mathbf{v}\in\Omega_1}\max_{\alpha\in\Omega_2} f(\mathbf{v},\alpha)\\
&\leq f(\bar{\mathbf{v}}_T,\alpha_{*,T}) - f(\mathbf{v}_*,\bar{\alpha}_T)\\
&\leq \frac{\|\mathbf{v}_1-\mathbf{v}_*\|_2^2}{2\eta T} + \frac{\|\alpha_1-\alpha_{*,T}\|_2^2}{2\eta T} + \eta G^2\\
&+ \frac{\sum_{t=1}^T(\mathbf{u}_t-\mathbf{u}_*)^\top(g(\mathbf{u}_t) - G(\mathbf{u}_t,\mathbf{z}_t))}{T}\\
&+ \frac{\sum_{t=1}^T(\mathbf{u}_t-\mathbf{u}_*)^\top(G(\mathbf{u}_t,\mathbf{z}_t) - \hat{G}_t(\mathbf{u}_t,\mathbf{z}_t))}{T}\\
&= \mathbf{I} + \mathbf{II} + \mathbf{III} + \mathbf{IV} + \mathbf{V}
\end{aligned}$$

where $\mathbf{v}_*$ is the optimal solution closest to $\mathbf{v}_1$ of (2), and the first inequality follows from the fact that $\min_{\mathbf{v}\in\Omega_1}\max_{\alpha\in\Omega_2} f(\mathbf{v},\alpha) \geq f(\mathbf{v}_*,\bar{\alpha}_T)$.

Now we bound the five terms respectively. It is obvious that $\mathbf{I} = \frac{\|\mathbf{v}_1-\mathbf{v}_*\|_2^2}{2\eta T} \leq \frac{r_0^2}{2\eta T}$.

Let $\widehat{A}_{k-1}, \widehat{p}_{k-1}, \xi_{k-1}$ be counterparts of that in Lemma 3 estimated from data in $(k-1)$-stage. According to the setting of α in Algorithm 1, for the first call of Algorithm 2 we have $\alpha_1 = A\mathbf{v}_1$ due to $\widehat{\mathbf{v}}_0 = 0$, and for k-th call of Algorithm 2 with $k>1$ we have $\alpha_1 = \widehat{A}_{k-1}\mathbf{v}_1$. Then for the first call of Algorithm 2, we have $\mathbf{II} = \frac{\|A\mathbf{v}_1 - A\bar{\mathbf{v}}_T\|_2^2}{2\eta T}$, and for other calls we have

$$\begin{aligned}
\mathbf{II} &= \frac{\|\widehat{A}_{k-1}\mathbf{v}_1 - A\bar{\mathbf{v}}_T\|_2^2}{2\eta T}\\
&\leq \frac{2\|\widehat{A}_{k-1}(\mathbf{v}_1-\bar{\mathbf{v}}_T)\|_2^2 + 2\|(\widehat{A}_{k-1}-A)\bar{\mathbf{v}}_T\|_2^2}{2\eta T}.
\end{aligned}$$

By Cauchy-Schwartz inequality, we have

$$\begin{aligned}
&\max\{\|A(\mathbf{v}_1-\bar{\mathbf{v}}_T)\|_2^2, \|\widehat{A}_{k-1}(\mathbf{v}_1-\bar{\mathbf{v}}_T)\|_2^2\}\\
&\leq 4\kappa^2\|\mathbf{v}_1-\bar{\mathbf{v}}_T\|_2^2,\\
&\|(\widehat{A}_{k-1}-A)\bar{\mathbf{v}}_T\|_2^2 \leq \|\widehat{A}_{k-1}-A\|_2^2\|\bar{\mathbf{v}}_T\|_2^2.
\end{aligned}$$

By combining the result in Lemma 3, we have with probability $1-\delta/3$

$$\mathbf{II} \leq \frac{8\kappa^2 r^2}{2\eta T} + \mathbb{I}_{k>1}\frac{32\kappa^2\left(2+\sqrt{2\ln(\frac{12}{\delta})}\right)^2(1+2\kappa)^2R^2}{2\xi_{k-1}\eta T^2}$$

Similarly, we can bound $|\alpha_1 - \alpha_{*,T}|$ by $2\sqrt{2}\kappa r$ for the first call, and

$$|\alpha_1 - \alpha_{*,T}| \leq \frac{4\sqrt{2}\kappa\left(2+\sqrt{2\ln(\frac{12}{\delta})}\right)(1+2\kappa)r}{\sqrt{(\min(\widehat{p}_{k-1}, 1-\widehat{p}_{k-1})n_0 - \sqrt{2n_0\ln(\frac{12}{\delta})})}} + 2\sqrt{2}\kappa r$$

for k-th call with $k > 1$. According to initial value of D and r for each call, we have $|\alpha_1 - \alpha_{*,T}| \leq D$.

Next, we can bound the last two terms similarly to (Ying et al., 2016). Define

$$\tilde{\mathbf{u}}_1 = \mathbf{u}_1 \in (\Omega_1 \cap \mathcal{B}(\mathbf{v}_1, r)) \times (\Omega_2 \cap \mathcal{B}(\alpha_1, D)),$$
$$\tilde{\mathbf{u}}_{t+1} = \Pi_{\substack{(\Omega_1\cap\mathcal{B}(\mathbf{v}_1,r)) \\ \times(\Omega_2\cap\mathcal{B}(\alpha_1,D))}}(\tilde{\mathbf{u}}_t - \eta(g(\mathbf{u}_t) - G(\mathbf{u}_t, \mathbf{z}_t))),$$

then we have with probability at least $1 - \frac{\delta}{3}$,

$$\begin{aligned}
&\sum_{t=1}^{T}\eta(\tilde{\mathbf{u}}_t - \mathbf{u}_*)^\top(g(\mathbf{u}_t) - G(\mathbf{u}_t, \mathbf{z}_t)) \\
&\leq \frac{\|\tilde{\mathbf{u}}_1 - \mathbf{u}_*\|_2^2}{2} + \frac{1}{2}\sum_{t=1}^{T}\eta^2\|g(\mathbf{u}_t) - G(\mathbf{u}_t, \mathbf{z}_t)\|_2^2 \\
&\leq \frac{\|\mathbf{v}_1 - \mathbf{v}_*\|_2^2 + \|\alpha_1 - \alpha_{*,T}\|_2^2}{2} + \frac{1}{2}T\eta^2 4G^2 \\
&\leq \frac{r^2 + D^2}{2} + 2\eta^2G^2T
\end{aligned}$$

Note that both $\mathbf{u}_t$ and $\tilde{\mathbf{u}}_t$ are measurable with respect to $\mathcal{F}_t = \{\mathbf{z}_1, \ldots, \mathbf{z}_{t-1}\}$, and $\{S_t : \eta(\mathbf{u}_t - \tilde{\mathbf{u}}_t)^\top(g(\mathbf{u}_t) - G(\mathbf{u}_t)) : t = 1, \ldots T\}$ is a martingale difference sequence, and for any t, we have $|\eta(\mathbf{u}_t - \tilde{\mathbf{u}}_t)^\top(g(\mathbf{u}_t) - G(\mathbf{u}_t, \mathbf{z}_t))| \leq 2\eta G\|\mathbf{u}_t - \tilde{\mathbf{u}}_t\|_2 \leq 2\eta G(2r + 2D)$. Then by Azuma-Hoeffding's inequality, we have with probability at least $1 - \frac{\delta}{3}$,

$$\begin{aligned}
&\sum_{t=1}^{T}\eta(\mathbf{u}_t - \tilde{\mathbf{u}}_t)^\top(g(\mathbf{u}_t) - G(\mathbf{u}_t, \mathbf{z}_t)) \\
&\leq 2\eta G(2r + 2D)\sqrt{2T\ln(\frac{3}{\delta})}
\end{aligned} \tag{5}$$

Hence, with probability $1 - \frac{2\delta}{3}$, we have

$$\begin{aligned}
\mathbf{IV} \leq\ & \frac{(1+8\kappa^2)r^2}{2\eta T} + \frac{16\kappa^2\left(2+\sqrt{2\ln(\frac{12}{\delta})}\right)^2(1+2\kappa)^2R^2}{\eta\xi_{k-1}T^2} \\
&+ 2\eta G^2 + \frac{4G(r+D)\sqrt{2\ln(\frac{3}{\delta})}}{\sqrt{T}}.
\end{aligned}$$

Next we bound $\mathbf{V}$. According to Lemma 3 of (Ying et al., 2016),

$$\begin{aligned}
\mathbf{V} \leq& \sum_{t=1}^{T}\left(\sup_t(\|\mathbf{u}_t - \mathbf{u}_1\|_2 + \|\mathbf{u}_1 - \mathbf{u}_*\|_2)\cdot \right. \\
&\left.\sup_{\mathbf{u}\in\Omega,\mathbf{z}\in\mathcal{Z}}\|\widehat{G}_t(\mathbf{u}, \mathbf{z}) - G(\mathbf{u}, \mathbf{z})\|_2\right)/T \\
\leq& \frac{4(r+D)\times 2\kappa(4\kappa R + 11R + 1)\sum_{t=1}^{T}\sqrt{\frac{\ln(\frac{6T}{\delta})}{t}}}{T} \\
\leq& \frac{8(r+D)G\sqrt{\ln(\frac{6T}{\delta})}}{\sqrt{T}},
\end{aligned}$$

where the last inequality holds since $\sum_{t=1}^{T}\frac{1}{\sqrt{t}} \leq 2\sqrt{T}$.

When $T \geq \frac{R^2}{r^2}$, by union bound, with probability at least $1 - \delta$ we have

$$\begin{aligned}
&\max_{\alpha\in\Omega_2} f(\bar{\mathbf{v}}_T, \alpha) - \min_{\mathbf{v}\in\Omega_1}\max_{\alpha\in\Omega_2} f(\mathbf{v}, \alpha) \\
&\leq \frac{\zeta_1 r^2}{\eta T} + 3\eta G^2 + \frac{\zeta_2 rG\sqrt{\ln(\frac{6T}{\delta})}}{\sqrt{T}},
\end{aligned}$$

where $\zeta_1 = (1+8\kappa^2) + \mathbb{I}_{[k>1]}\frac{32\kappa^2(1+2\kappa)^2\left(2+\sqrt{2\ln(\frac{12}{\delta})}\right)^2}{\xi_{k-1}}$,
$\zeta_2 = 16\left(1 + 2\sqrt{2}\kappa + \mathbb{I}_{[k>1]}\frac{8\kappa\left(2+\sqrt{2\ln(\frac{12}{\delta})}\right)(1+2\kappa)}{\sqrt{\xi_{k-1}}}\right)$,
$\xi_{k-1} = \min(\hat{p}_{k-1}, 1-\hat{p}_{k-1}) - \sqrt{\frac{2\ln(\frac{12}{\delta})}{T}}$.

Moreover, by choosing $\eta = \frac{\sqrt{\zeta_1}r}{\sqrt{3}G\sqrt{T}}$, we have with probability at least $1 - \delta$, we have

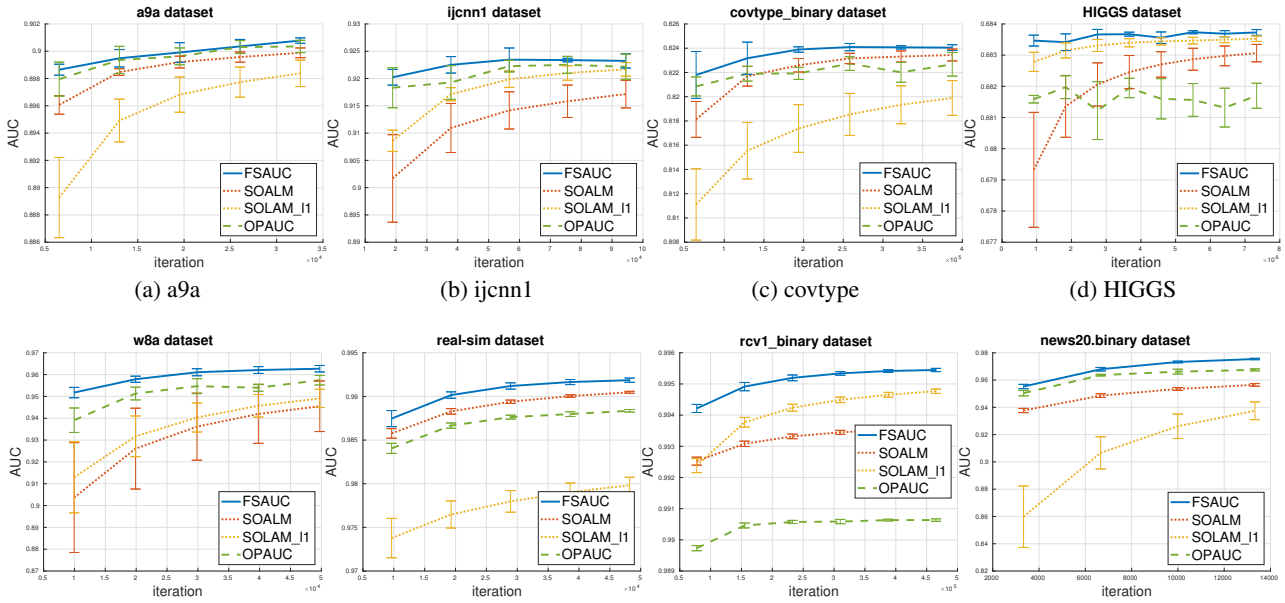
$$\begin{aligned}
&\max_{\alpha\in\Omega_2} f(\bar{\mathbf{v}}_T, \alpha) - \min_{\mathbf{v}\in\Omega_1}\max_{\alpha\in\Omega_2} f(\mathbf{v}, \alpha) \\
&\leq \frac{\left(2\sqrt{3\zeta_1} + \sqrt{\ln(\frac{6T}{\delta})}\zeta_2\right)r_0G}{\sqrt{T}}.
\end{aligned}$$

By Hoeffding inequality,

$$\begin{aligned}
pT &\geq \widehat{p}_{k-1}T - \sqrt{\frac{T\ln(\frac{12}{\delta})}{2}}, \\
(1-p)T &\geq (1-\widehat{p}_{k-1})T - \sqrt{\frac{T\ln(\frac{12}{\delta})}{2}}.
\end{aligned}$$

When $T \geq \frac{32\ln(\frac{12}{\delta})}{(\min(p,1-p))^2}$, we can bound $\zeta_1 \leq \gamma_1$ and $\zeta_2 \leq \gamma_2$ due to $T \geq \frac{32\ln(\frac{12}{\delta})}{(\min(p,1-p))^2}$, then the conclusion follows.

□



(a) a9a (b) ijcnn1 (c) covtype (d) HIGGS

(e) w8a (f) real-sim (g) rcv1_binary (h) news20.binary

Figure 1. AUC-Iteration curves of FSAUC and the baselines

5. Experiments

In this section, we conduct experiments to compare our FSAUC algorithm with two state-of-the-art stochastic/online AUC optimization methods, namely SOLAM (Ying et al., 2016) and OPAUC (Gao et al., 2013). To make the comparison fair, we implement two variations of SOLAM by using two different norm constraints on $\mathbf{w}$: SOLAM with ℓ_2 norm constraint (the original one) referred to as SOLAM, and SOLAM with ℓ_1 norm constraint referred to as SOLAM-l1.

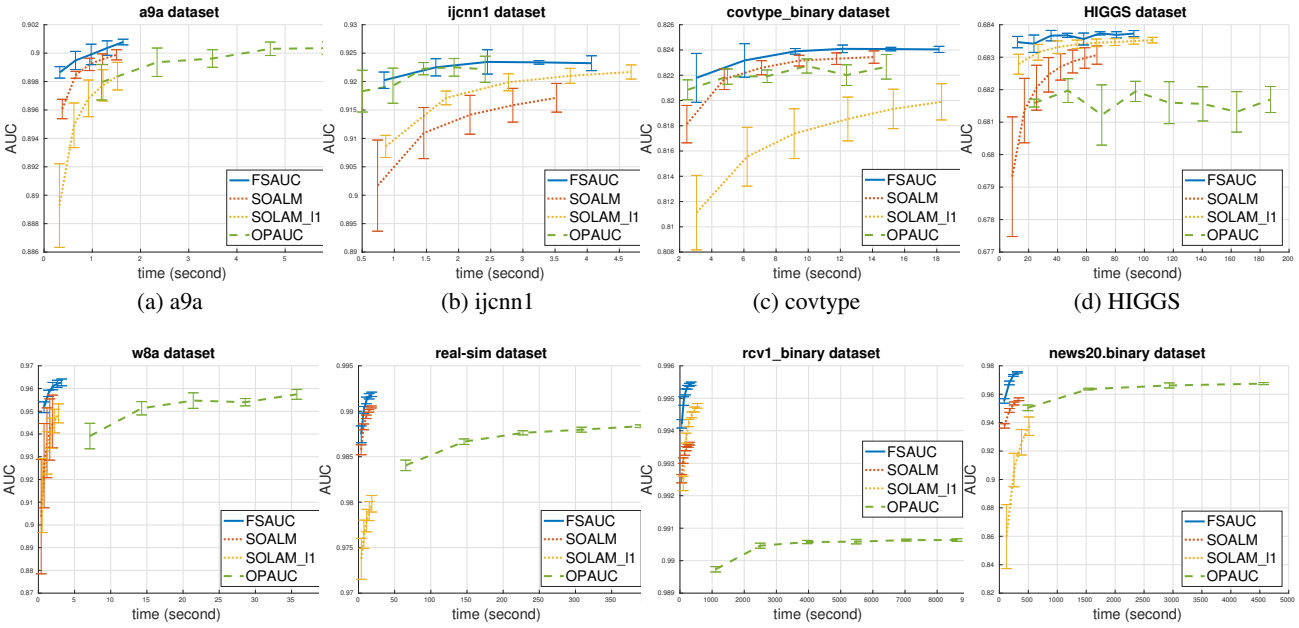
We use eight large-scale benchmark datasets from libsvm website[3], ranging from high-dimensional to low-dimensional, from balanced class distribution to imbalanced class distribution. The statistics of these datasets are summarized in Table 1. We randomly divide each dataset into three sets, respectively training, validation, and testing. For a9a and w8a datasets, we randomly split the given testing set into half validation and half testing. For the datasets that do not explicitly provide a testing set, we randomly split the entire data into 4:1:1 for training, validation, and testing. For the dataset ijcnn1 and rcv1_binary, despite that the test set is given, the size of the training set is relatively small. Thus we first combine the training and the test sets and then follow the above procedure to split it.

The involved parameters of each algorithm are tuned based on the validation data. FSAUC has two parameters R and G. R is decided within the range $10^{[-1:1:5]}$. G affects the stepsize of each epoch (Algorithm 1 line 4-5). Since $\eta_1 = \frac{\sqrt{\beta_0}}{\sqrt{3n_0G}}R_0$ and $\eta_{k+1} = \frac{\sqrt{\beta_k}}{2\sqrt{\beta_{k-1}}}\eta_k$, we equivalently tune $\eta_1 \in 2^{[-10:1:10]}$. As for SOLAM, following the same strategy in the original paper (Ying et al., 2016), we tune R in $10^{[-1:1:5]}$ and the learning rate in $2^{[-10:1:10]}$. OPAUC has two versions corresponding to different ways of maintaining the covariance matrix, the full version and an approximated version designed to deal with high dimensional data. On the five low-dimensional datasets (a9a, ijcnn1, covtype_binary, HIGGS, w8a), we use the full version of OPAUC, since the dimension is low enough and thus it is unnecessary to use the low rank approximation method, which is less accurate and more complicated. For the three high-dimensional datasets (real-sim, rcv1_binary, news20.binary), we set the low rank $\tau = 50$, the same as the original paper (Gao et al., 2013). There are two parameters shared by both versions of OPAUC, step size η and regularized parameter λ. Following the suggestion of the

Table 1. Statistics of Datasets

Datasets	#Training	#Testing	♯feat	% of Pos
a9a	32,561	8,141	123	24.08
ijcnn1	94,460	23,616	22	9.49
covtype_binary	387,341	96,836	54	51.19
HIGGS	7,333,333	1,833,334	28	52.98
w8a	49,749	7,476	300	2.97
real-sim	48,206	12,052	20,958	30.68
rcv1_binary	465,094	116,274	47,236	52.41
news20.binary	13,330	3,333	1,355,191	50.29

[3]http://www.csie.ntu.edu.tw/ cjlin/libsvmtools/datasets/



(a) a9a (b) ijcnn1 (c) covtype (d) HIGGS

(e) w8a (f) real-sim (g) rcv1_binary (h) news20.binary

Figure 2. AUC-Time curves of FSAUC and the baselines

Table 2. Averaged final AUC on testing data

Datasets	FSAUC	SOLAM	SOLAM_l1	OPAUC
a9a	.900782 ± .000197	.899884 ± .000358	.898382 ± .000973	.900350 ± .000450
ijcnn1	.923230 ± .001310	.917147 ± .002526	.921682 ± .001249	.922163 ± .002296
covtype_binary	.824046 ± .000242	.823440 ± .000489	.819894 ± .001434	.822669 ± .000973
HIGGS	.683727 ± .000093	.683064 ± .000280	.683526 ± .000086	.681697 ± .000401
w8a	.962703 ± .001485	.945508 ± .011601	.949083 ± .004182	.952536 ± .002712
real-sim	.991862 ± .000243	.990473 ± .000111	.979830 ± .000911	.988357 ± .000147
rcv1_binary	.995448 ± .000052	.993601 ± .000044	.994767 ± .000071	.990634 ± .000039
news20.binary	.975428 ± .000513	.956472 ± .000957	.937493 ± .006501	.967471 ± .000721

original paper (Gao et al., 2013), we tune $\eta \in 2^{[-12:1:-4]}$ and $\lambda \in 2^{[-10:1:0]}$.

Each algorithm updates the model by one pass of training data and the models at different iterations are evaluated by AUC computed on the testing data to demonstrate the (testing) convergence speed of different algorithms. We report the results on testing sets averaged over 5 random runs over the training data. The convergence curves of considered algorithms are plotted in Figure 1 and Figure 2 in terms of both the number of iterations and training time. The final AUC with the standard deviation over 5 runs is summarized in Table 2. From Figure 1, we can see that the proposed FSAUC converges faster than the two state-of-the-art methods, which is consistent with our theory. For the convergence in terms of running time shown in Figure 1, the proposed FSAUC also has the best performance.

6. Conclusion

In this paper, we have proposed a novel stochastic algorithm for AUC maximization in the classical stochastic setting where one random data is received at each iteration. We theoretically analyze the proposed algorithm and derive a fast convergence rate of $\widetilde{O}(1/n)$, largely improved from the best result that the current state-of-the-art method can achieve - $\widetilde{O}(1/\sqrt{n})$. We have also verified the efficiency of our algorithm by experiments on multiple benchmark datasets, and the results show that our algorithm converges faster than two strong baseline algorithms in terms of both the number of iterations and running time. For future work, we will consider fast stochastic algorithms for AUC maximization based on different surrogate losses. One may also consider extending the proposed algorithm to learning to rank from pairwise data in which the label of each data have more than two possible values in the binary classification.

Acknowledgments

We thank the anonymous reviewers for their helpful comments. M. Liu, X. Zhang, T. Yang are partially supported by National Science Foundation (IIS-1545995).

References

Bach, F. R. and Moulines, E. Non-strongly-convex smooth stochastic approximation with convergence rate $\mathcal{O}(1/n)$. In *Advances in Neural Information Processing Systems 26 (NIPS)*, pp. 773–781, 2013.

Bauschke, H. H. and Borwein, J. M. On the convergence of von neumann's alternating projection algorithm for two sets. *Set-Valued Analysis*, 1(2):185–212, 1993.

Cortes, C. and Mohri, M. Auc optimization vs. error rate minimization. In *Advances in neural information processing systems*, pp. 313–320, 2004.

Ferri, C., Hernández-Orallo, J., and Flach, P. A. A coherent interpretation of auc as a measure of aggregated classification performance. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pp. 657–664, 2011.

Gao, W., Jin, R., Zhu, S., and Zhou, Z.-H. One-pass auc optimization. In *International Conference on Machine Learning*, pp. 906–914, 2013.

Ghadimi, S. and Lan, G. Optimal stochastic approximation algorithms for strongly convex stochastic composite optimization, ii: Shrinking procedures and optimal algorithms. *SIAM Journal on Optimization*, 23(4):20612089, 2013.

Hanley, J. A. and McNeil, B. J. The meaning and use of the area under a receiver operating characteristic (roc) curve. *Radiology*, 143(1):29–36, 1982.

Hanley, J. A. and McNeil, B. J. A method of comparing the areas under receiver operating characteristic curves derived from the same cases. *Radiology*, 148(3):839–843, 1983.

Hazan, E. and Kale, S. Beyond the regret minimization barrier: an optimal algorithm for stochastic strongly-convex optimization. *Journal of Machine Learning Research - Proceedings Track*, 19:421–436, 2011a.

Hazan, E. and Kale, S. Beyond the regret minimization barrier: an optimal algorithm for stochastic strongly-convex optimization. In *Proceedings of the 24th Annual Conference on Learning Theory (COLT)*, pp. 421–436, 2011b.

Herschtal, A. and Raskutti, B. Optimising area under the roc curve using gradient descent. In *Proceedings of the twenty-first international conference on Machine learning*, pp. 49. ACM, 2004.

Juditsky, A., Nesterov, Y., et al. Deterministic and stochastic primal-dual subgradient algorithms for uniformly convex minimization. *Stochastic Systems*, 4(1):44–80, 2014.

Metz, C. E. Basic principles of roc analysis. In *Seminars in nuclear medicine*, volume 8, pp. 283–298. Elsevier, 1978.

Srebro, N., Sridharan, K., and Tewari, A. Smoothness, low noise and fast rates. In *Advances in Neural Information Processing Systems (NIPS)*, pp. 2199–2207, 2010.

Xu, Y., Lin, Q., and Yang, T. Stochastic convex optimization: Faster local growth implies faster global convergence. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, pp. 3821–3830, 2017.

Ying, Y., Wen, L., and Lyu, S. Stochastic online auc maximization. In *Advances in Neural Information Processing Systems*, pp. 451–459, 2016.

Zhao, P., Jin, R., Yang, T., and Hoi, S. C. Online auc maximization. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pp. 233–240, 2011.