Stochastic Proximal Algorithms for AUC Maximization

Michael Natole, Jr.[1], Yiming Ying[1], Siwei Lyu[2]

Department of Mathematics & Statistics[1], Department of Computer Science[2], University at Albany, SUNY

UNIVERSITY AT ALBANY
State University of New York

SGD for Massive Streaming Data

- Data $\{z_i = (x_i, y_i) \in \mathcal{Z} : i = 1 \ldots T\}$, where $\mathcal{X} \subseteq \mathbb{R}^d$ and $\mathcal{Y} = \{\pm 1\}$ (binary classification). T is continuously increasing (**streaming data**).
- The quality of a classifier $f_{\mathbf{w}}(x) = \operatorname{sign}(\mathbf{w}^\top x)$ can be measured by misclassification error (accuracy)-related loss $\phi(\mathbf{w}^\top x)$. For example, 0-1 loss $\phi(y\mathbf{w}^\top x) = \mathbb{I}_{[y\mathbf{w}^\top x \leq 0]}$ or least square loss $\phi(\mathbf{w}^\top x) = (1 - y\mathbf{w}^\top x)^2$.
- Stochastic Gradient Descent (SGD) for accuracy performance measure: assume $\{z_t = (x_t, y_t)\}$ is i.i.d., then SGD performs as follows:

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \nabla_{\mathbf{w}} \phi(\mathbf{w}_t^\top x_t)$$

- The literature on SGD for accuracy is vast *[Bach (2011); Bottou and Le Cun (2003); Nemirovski et al. (2008); Shamir and Zhang (2012); Srebro and Tewari (2010); Rakhlin et al. (2102); Ying and Pontil (2008);....]*
- **Rough summary of SGD's convergence**: time and per-iteration cost $\mathcal{O}(d)$; convergence rate is of $\mathcal{O}(1/\sqrt{t})$, and $\mathcal{O}(1/t)$ if strongly convex.

AUC Maximization

- However, accuracy is not suitable for important learning tasks such as imbalanced classification. Area under the ROC curve (AUC) is a suitable performance measure in imbalanced classification (anomaly detection and cancer diagnosis), and information retrieval. *[Hanley and McNeil (1982); Elkan (2001); Cortes and Mohri (2003); Fawcett, 2006]*
- **Definition of AUC Score:** For a linear scoring function $f(x) = \mathbf{w}^T x$, its AUC score *[Clemencon et al. (2008)]* is defined by

$$\begin{aligned} AUC(\mathbf{w}) &= \Pr(\mathbf{w}^T x \geq \mathbf{w}^T x' | y = 1, y' = -1) \\ &= 1 - \mathbb{E}[\mathbb{I}_{[\mathbf{w}^T(x - x') < 0]} | y = 1, y' = -1]. \end{aligned}$$

- Replacing the indicator function by the least square loss, maximizing AUC can be formulated as:

$$\min_{\mathbf{w}} \mathbb{E}[(1 - \mathbf{w}^T(x - x'))^2 | y = 1, y' = -1] + \Omega(\mathbf{w}) \qquad (1)$$

where $\Omega(\cdot)$ is a convex penalty term.

Question: How to design SGD like algorithms on par with the accuracy case?
Challenge: The objective function is a double integral (summation) over pairs of samples while, in practice, one DOES NOT receive pairs rather a fast-updating sequence of individual samples.

Previous Work

- Various work *[Wang et al. (2008); Zhao et al. (2012)]* developed SGD/OGD based on the local error $\mathcal{L}_t(\mathbf{w}) = \frac{1}{t-1}\sum_{j=1}^{t-1} \text{Loss}(y_t, \mathbf{w}^\top x_j)$ which have storage and per-iteration costs $\mathcal{O}(td)$ at time t
- *Gao et al. (2013)* focused on least square loss; Notice that it only needs to update the covariance matrix which have storage and time complexity $\mathcal{O}(d^2)$.
- Recent work *[Ying et al. (2016)]* showed that (1) is equivalent to the saddle point problem (SPP):

$$\min_{\mathbf{w}, a, b} \max_{\alpha \in \mathbb{R}} \{\mathbb{E}_z[F(\mathbf{w}, a, b, \alpha; z)] + \Omega(\mathbf{w})\}, \qquad (2)$$

where $F(\mathbf{w}, a, b, \alpha; z)$ is quadratic (see the paper).
- The algorithm there is based on gradient descent on primal variables $(\mathbf{w}, a, b)$ and ascent on dual variable α. It is known that such stochastic first-order algorithm has an optimal rate $\mathcal{O}(1/\sqrt{t})$.

Algorithm	Loss	Ω	Storage	Iteration	Rate
OAM	General	L^2	$\mathcal{O}(td)$	$\mathcal{O}(td)$	$\mathcal{O}(1/\sqrt{T})$
OPAUC	Least-Square	L^2	$\mathcal{O}(d^2)$	$\mathcal{O}(d^2)$	$\mathcal{O}(1/\sqrt{T})$
SOLAM	Least-Square	L^2	$\mathcal{O}(d)$	$\mathcal{O}(d)$	$\mathcal{O}(1/\sqrt{T})$
SPAM (this paper)	Least-Square	General	$\mathcal{O}(d)$	$\mathcal{O}(d)$	$\mathcal{O}(1/T)$

Our Algorithm

- Key observation: for fixed $\mathbf{w}$,

$$\mathbb{E}[(1 - \mathbf{w}_t^\top(x - x'))^2 | y = 1, y' = -1] = \min_{a,b} \max_{\alpha} \mathbb{E}_z[F(\mathbf{w}, a, b, \alpha; z)].$$

In particular, the optima are achieved at

$$a(\mathbf{w}) = \mathbf{w}^\top \mathbb{E}[x | y = 1], \quad b(\mathbf{w}) = \mathbf{w}^\top \mathbb{E}[x | y = -1], \qquad (3)$$

$$\alpha(\mathbf{w}) = \mathbf{w}^\top(\mathbb{E}[x | y' = -1] - \mathbb{E}[x | y = 1]). \qquad (4)$$

- New algorithm: update $\mathbf{w}$ using (proximal) gradient descent while a, b, α are given by (3) and (4)

Algorithm 1 : SPAM

Initialize $\mathbf{w}_1 \in \mathbb{R}^d$.
for $t = 1$ **to** T **do**
  Receive sample $z_t = (x_t, y_t)$
  Compute $a(\mathbf{w}_t)$, $b(\mathbf{w}_t)$, and $\alpha(\mathbf{w}_t)$ according to (3) and (4).
  $\hat{\mathbf{w}}_{t+1} = \mathbf{w}_t - \eta_t \partial_1 F(\mathbf{w}_t, a(\mathbf{w}_t), b(\mathbf{w}_t), \alpha(\mathbf{w}_t); z_t)$
  $\mathbf{w}_{t+1} = \operatorname{prox}_{\eta_t \Omega}(\hat{\mathbf{w}}_{t+1})$
end for

- $\partial_1 F$ denotes the partial derivative of F wrt the first argument.
- Proximal Mapping: $\operatorname{prox}_{\eta_t \Omega}(u) = \arg\min_{\mathbf{w}} \left\{\frac{1}{2}\|u - \mathbf{w}\|_2^2 + \eta_t \Omega(\mathbf{w})\right\}$.
- Our algorithm SPAM shares the spirit as the online forward-backward splitting *[Duchi and Singer (2009)]* and stochastic proximal gradient methods *[Rosasco et al. (2014)]*. However, there are two main differences between ours and previous proximal splitting algorithms:
  - They focused on the accuracy performance where the objective function is a single summation/integral over individual samples.
  - The convergence proofs there critically depend on boundedness assumptions: the boundedness of the iterates and/or the stochastic gradients stochastic gradient; our proof for SPAM does not need these boundedness assumptions.

Convergence Analysis

- (A1) Assume that $\Omega(\cdot)$ is β-strongly convex.
- (A2) There exists an $M > 0$ such that $\|x\| \leq M$ for any $x \in \mathcal{X}$.
- Let $C_{\beta,M} := \frac{\beta}{128M^4}$, $\widetilde{C}_{\beta,M} = \beta/(1 + \frac{\beta^2}{128M^4})^2$, and $\mathbf{w}^*$ denote the optimal solution of AUC maximization formulation (1).

Theorem

Under the assumptions (A1), (A2), and choosing step sizes with some $\theta \in (0, 1)$ in the form of $\{\eta_t = \frac{C_{\beta,M}}{t^\theta} : t \in \mathbb{N}\}$, the algorithm SPAM achieves the following:

$$\mathbb{E}[\|\mathbf{w}_{T+1} - \mathbf{w}^*\|^2] = \mathcal{O}(T^{-\theta}).$$

In particular, if we choose $\{\eta_t = [\widetilde{C}_{\beta,M}(t + 1)]^{-1} : t \in \mathbb{N}\}$, then there holds

$$\mathbb{E}[\|\mathbf{w}_{T+1} - \mathbf{w}^*\|^2] = \mathcal{O}\left(\frac{\log T}{T}\right).$$

- The convergence of SPAM can achieve $\mathcal{O}(1/T)$ up to a logarithmic term, which matches the optimal rate of standard SGD for accuracy with the same storage and per-iteration cost.
- Critical idea in the proof: the stochastic gradient $\partial_1 F(\mathbf{w}_t, a(\mathbf{w}_t), b(\mathbf{w}_t), \alpha(\mathbf{w}_t); z_t)$ is an unbiased estimator of the true gradient $\partial_{\mathbf{w}} f(\mathbf{w}_t) = \partial_{\mathbf{w}} \mathbb{E}[(1 - \mathbf{w}_t^\top(x - x'))^2 | y = 1, y' = -1]$, i.e.

$$\partial f(\mathbf{w}_t) = \mathbb{E}_{z_t}[\partial_1 F(\mathbf{w}_t, a(\mathbf{w}_t), b(\mathbf{w}_t), \alpha(\mathbf{w}_t); z_t)],$$

Experiments

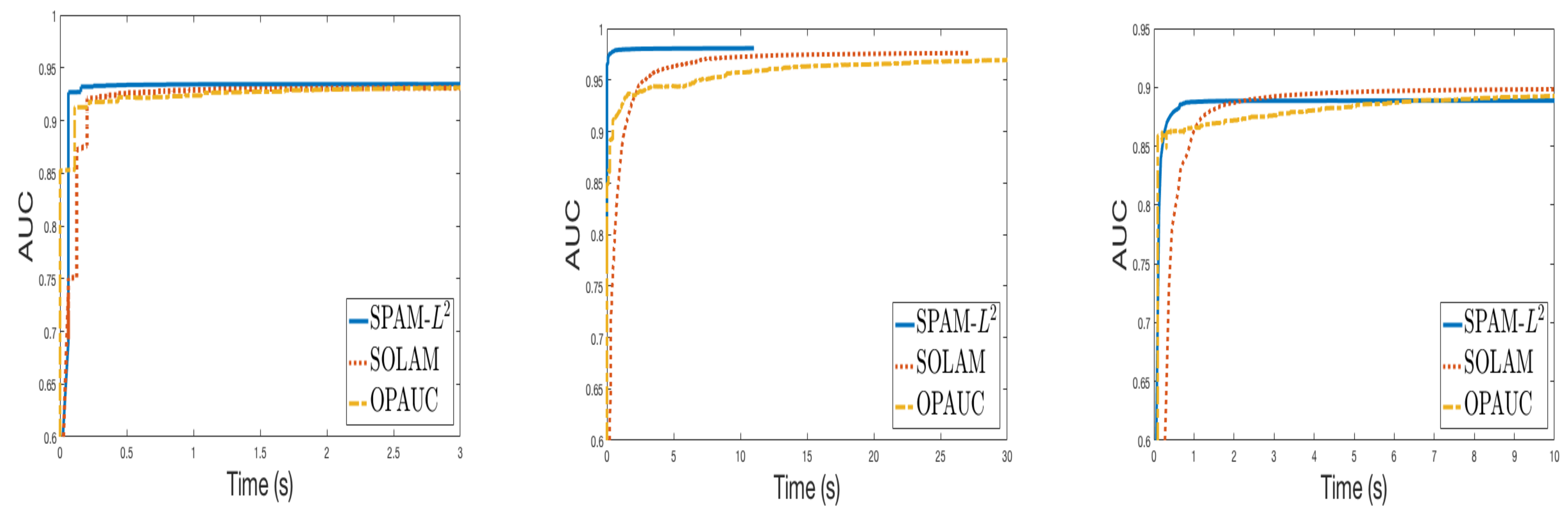
- **SPAM-L^2**: Our proposed algorithm for AUC maximization with $\Omega(\mathbf{w}) = \frac{\beta}{2}\|\mathbf{w}\|^2$.
- **SPAM-NET**: Our proposed algorithm for AUC maximization with elastic net $\Omega(\mathbf{w}) = \frac{\beta}{2}\|\mathbf{w}\|^2 + \beta_1\|\mathbf{w}\|_1$.
- **SOLAM**: Stochastic online algorithm for AUC maximization *[Ying et al. (2016)]*.
- **OPAUC**: The one-pass AUC optimization algorithm with square loss function *[Gao et al. (2013)]*
- **OAMseq**: The OAM algorithm with reservoir sampling and sequential updating method *[Zhao et al. (2011)]*.
- **B-LS-SVM**: A batch learning algorithm which optimizes the pairwise least square loss *[Joachims (2006)]*.

Name	# Instances	#Dim	Name	# Instances	# Dim	Name	# Instances	# Dim
diabetes	768	8	mnist	60,000	780	a9a	32,561	123
fourclass	862	8	acoustic	78,823	50	news20	15,935	62,061
german	1000	24	ijcnn1	141,691	22	usps	9,298	256
splice	3175	60	covtype	581,012	54	sector	9,619	55,197

Table: Statistics about the datasets.

Data	SPAM-L^2	SPAM-NET	SOLAM	OPAUC	OAM_{seq}	B-LS-SVM
diabetes	.8272±.0277	.8085±.0431	.8128±.0304	.8309±.0350	.8264±.0367	.8325±.0329
fourclass	.8211±.0205	.8213±.0209	.8310±.0251	.8306±.0247	.8295±.0251	.8309±.0309
german	.7942±.0388	.7937±.0386	.7778±.0373	.7978±.0347	.7747±.0411	.7994±.0343
splice	.9263±.0091	.9267±.0090	.9246±.0087	.9232±.0099	.8594±.0194	.9245±.0092
usps	.9868±.0032	.9855±.0029	.9822±.0036	.9620±.0040	.9310±.0159	.9634±.0045
a9a	.8998±.0046	.8980±.0047	.8966±.0043	.9002±.0047	.8420±.0174	.8982±.0028
mnist	.9254±.0025	.9132±.0026	.9118±.0029	.9242±.0021	.8615±.0087	.9336±.0025
acoustic	.8120±.0030	.8109±.0028	.8099±.0036	.8192±.0032	.7113±.0590	.8210±.0033
ijcnn1	.9174±.0024	.9155±.0024	.9129±.0030	.9269±.0021	.9209±.0079	.9320±.0037
covtype	.9504±.0011	.9508±.0011	.9503±.0012	.8244±.0014	.7361±.0317	.8222±.0014
sector	.8768±.0126	.9077±.0104	.8767±.0129	.9292±.0081	.9163±.0087	-
news20	.8708±.0069	.8704± .0070	.8712±.0073	.8871±.0083	.8543±.0099	-

Table: Comparison of the testing AUC values (mean±std.)



(a) splice (b) usps (c) a9a

Figure: AUC vs. CPU running time

Conclusion and Future Work

- We proposed a novel stochastic proximal algorithm (SPAM) for AUC maximization with general penalty terms.
- SPAM can achieve a convergence rate of $\mathcal{O}(1/T)$ up to a logarithmic term for strongly convex objective functions.
- Future work:
  - Stochastic variance reduction algorithms for AUC maximziation
  - AUC maximization with deep neural network
  - Learning theory for AUC maximization (consistency and optimal generalization bounds)
  - Can SPAM achieve a convergence rate of $\mathcal{O}(1/T)$ without strong convexity?