UNIVERSITY AT ALBANY
State University of New York

Stochastic Proximal Algorithms for AUC Maximization

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

[1]Department of Mathematics and Statistics

[2]Department of Computer Science

March 24, 2018

Outline

Motivation
- Why AUC?
- What is AUC?

AUC Maximization
- AUC Optimization
- Stochastic Proximal AUC Maximization Algorithm
- Convergence Analysis

Experiments

Conclusion

Classification

- Given data $\{z_i = (x_i, y_i) \in \mathcal{Z} : i = 1 \ldots T\}$, where $\mathcal{X} \subseteq \mathbb{R}^d$ and $\mathcal{Y} = \{\pm 1\}$, we wish to learn the following function

$$f(x_i) = \text{sign}(\mathbf{w}^T x_i) \tag{1}$$

  where $\mathbf{w} \in \mathbb{R}^d$ is the parameter to be learned.
- Evaluation by 0-1 loss is usually replaced by a convex surrogate loss $\phi : \mathbb{R} \to \mathbb{R}^+$ satisfying $\mathbb{I}_{[s<0]} \leq \phi(s)$.
  - Least Square Loss: $\phi(s) = (1 - s)^2$
  - Hinge Loss: $\phi(s) = (1 - s)_+$
- Empirical Risk Minimization (ERM)

$$\mathbf{w}^* = \arg\min_{\mathbf{w}} \frac{1}{T} \sum_{i=1}^{T} \phi(y_i \mathbf{w}^T x_i). \tag{2}$$

Stochastic Gradient Descent

Stochastic Gradient Descent

Initialize $\mathbf{w}_1$, and for any $t \geq 1$, draw sample $z_t = (x_t, y_t)$ at random, and then

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \nabla_{\mathbf{w}} \phi(y_t \mathbf{w}^T x_t) \tag{3}$$

- The idea of SGD dates back to Robbins and Monroe (1951).
- The literature on SGD is extensive [Bottou & Cunn (2004); Srebro & Tewari (2010); Moulines & Bach (2011);...].
- Most of the literature focuses on the misclassification error or accuracy.

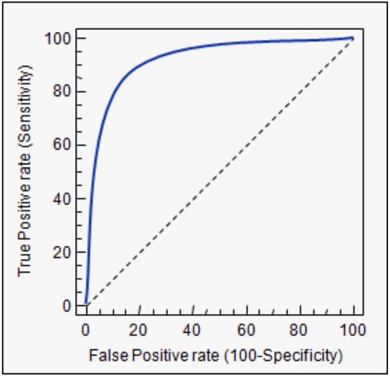
Accuracy

- Consider the case for a sample of 1000 instances with 990 "true" negative instances and 10 "true" positive instances. Suppose we obtain the following results:

	True $+1$	True -1
Predicted $+1$	1	11
Predicted -1	9	979

- The misclassification error (or classification accuracy) could be misleading for real world applications.
- This classifier has 98% accuracy, but told us very little.
- For this reason, we consider the use of AUC.

Receiver Operating Characteristic (ROC) Curve



- Given a confusion matrix, a ROC curve is a plot of the False Positive Rate (FPR) on the x-axis and the True Positive Rate (TPR) on the y-axis.

$$TPR = \frac{TP}{TP + FN}$$

$$FPR = \frac{FP}{FP + TN}$$

[Hanley & McNeil (1982); Bradley (1997); Fawcett (2006)]

Probabilistic Definition of AUC

Definition

For a linear scoring function $f(x) = \mathbf{w}^T x$, AUC is

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

where (x, y), $(x', y') \in \mathcal{Z} = \mathcal{X} \times \mathcal{Y}$ are independent.

- In imbalanced classification and information retrieval, one often uses AUC (area under the ROC curve).
- AUC is expressed as a sum of pairwise losses between instances from different classes, which is quadratic in the number of received training examples

Outline

Motivation

- Why AUC?
- What is AUC?

AUC Maximization

- AUC Optimization
- Stochastic Proximal AUC Maximization Algorithm
- Convergence Analysis

Experiments

Conclusion

AUC Maximization

- AUC maximization can be easily modified to a minimization problem:

$$\min_{\mathbf{w}} \mathbb{E}[\mathbb{I}_{[\mathbf{w}^T(x-x')<0]}|y=1, y'=-1] + \Omega(\mathbf{w})$$

  where $\Omega(\cdot)$ is a penalty function.
- Replacing the indicator function by the least square loss, AUC optimization can be formulated as:

$$\min_{\mathbf{w}} \mathbb{E}[(1-\mathbf{w}^T(x-x'))^2|y=1, y'=-1] + \Omega(\mathbf{w}) \qquad (4)$$

AUC Maximization

- When ρ is a uniform distribution over the finite data $\{z_i = (x_i, y_i) \in \mathcal{Z} : i = 1 \ldots T\}$, AUC maximization reduces to

$$\min_{\mathbf{w}} \frac{1}{n_+ n_-} \sum_{i,j=1}^{n} (1 - \mathbf{w}^T (x_i - x_j))^2 \mathbb{I}_{y_i = 1 \wedge y_j = -1} + \Omega(\mathbf{w})$$

  where n_+ and n_- denote the number of instances in the positive and negative classes, respectively.
- Key Challenges
  - What happens if the dataset is very large?
  - How to handle streaming data?

Summary of Existing Work

Algorithm	Loss	Penalty	Storage	Iteration	Rate
OAM	General	L^2	$\mathcal{O}(td)$	$\mathcal{O}(td)$	$\mathcal{O}(1/\sqrt{T})$
OLP	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	$\mathcal{O}(d)$	$\mathcal{O}(d)$	$\mathcal{O}(1/\sqrt{T})$
New Alg.	Least-Square	General	$\mathcal{O}(d)$	$\mathcal{O}(d)$	$\mathcal{O}(1/T)$

[Zhao et al. (2012); Kar et al. (2014); Gao et al (2013); Ying et al. (2016)]

Previous Work

Theorem

AUC optimization (4) in the linear case is equivalent to the following saddle point problem:

$$\min_{\mathbf{w},a,b} \max_{\alpha \in \mathbb{R}} \left\{ \mathbb{E}[F(\mathbf{w}, a, b, \alpha; z)] + \Omega(\mathbf{w}) \right\}, \tag{5}$$

where the expectation is with respect to $z = (x, y)$, and $F(\mathbf{w}, a, b, \alpha; z)$ is a quadratic function involving $p = Pr(y = 1)$.

- AUC maximization can be reduced to a single integral.

[Ying et al. (2016)]

Motivation of Key Ideas

SOLAM

Upon receiving data z_t, perform

1. Gradient descent on the primal variables $\mathbf{v} = (\mathbf{w}, a, b)$

$$\mathbf{v}_{t+1} = \mathbf{v}_t - \gamma_t \partial_{\mathbf{v}} F(\mathbf{v}_t, \alpha_t, z_t)$$

2. Gradient ascent on the dual variable α:

$$\alpha_{t+1} = \alpha_t + \gamma_t \partial_{\alpha} F(\mathbf{v}_t, \alpha_t, z_t)$$

- This has a theoretical convergence rate of $\mathcal{O}(1/\sqrt{T})$, but can we do better?

[Nemirovski et al. (2009); Ying et al. (2016)]

Our Key Ideas

- For fixed $\mathbf{w}$, it is easy to see that the optima for a, b, and α are respectively achieved at

$$a(\mathbf{w}) = \mathbf{w}^\top \mathbb{E}[x|y=1], \quad b(\mathbf{w}) = \mathbf{w}^\top \mathbb{E}[x|y=-1], \tag{6}$$

$$\alpha(\mathbf{w}) = \mathbf{w}^\top (\mathbb{E}[x|y'=-1] - \mathbb{E}[x|y=1]). \tag{7}$$

- Using the updates for a, b, and α, our new AUC optimization formulations becomes

$$\min_{\mathbf{w}} \mathbb{E}[F(\mathbf{w}, a(\mathbf{w}), b(\mathbf{w}), \alpha(\mathbf{w}); z_t)] + \Omega(\mathbf{w}) \tag{8}$$

Stochastic Proximal AUC Maximization

SPAM

Input: Step sizes $\{\eta_t > 0 : t \in \mathbb{N}\}$
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 (6) and (7).
  $\hat{\mathbf{w}}_{t+1} = \mathbf{w}_t - \eta_t \partial_1 F(\mathbf{w}_t, a(w_t), b(\mathbf{w}_t), \alpha(\mathbf{w}_t); z_t)$
  $\mathbf{w}_{t+1} = \text{prox}_{\eta_t \Omega}(\hat{\mathbf{w}}_{t+1})$
end for

- The proximal step is given by

$$\text{prox}_{\eta_t \Omega}(u) = \arg\min \left\{ \frac{1}{2} \| u - \mathbf{w} \|_2^2 + \eta_t \Omega(\mathbf{w}) \right\}$$

Advantage of SPAM

- Because of the use of the proximal operator, SPAM can accommodate for a non-smooth penalty term $\Omega(\cdot)$.
- We will consider:
  - L^2, i.e. $\Omega(\mathbf{w}) = \frac{\beta}{2}\|\mathbf{w}\|_2^2$
  - Elastic Net, i.e. $\Omega(\mathbf{w}) = \frac{\beta}{2}\|\mathbf{w}\|_2^2 + \beta_1\|\mathbf{w}\|_1^2$

[Zou & Hastie (2005)]

Convergence Analysis: Assumptions

- The convergence results are established based on the following two assumptions:
  - (A1) Assume that $\Omega(\cdot)$ is β-strongly convex.
  - (A2) There exists an $M > 0$ such that $\|\mathbf{x}\| \leq M$ for any $x \in \mathcal{X}$.
- Furthermore, we define the following constants:

$$C_{\beta,M} := \frac{\beta}{128M^4} \qquad \widetilde{C}_{\beta,M} = \frac{\beta}{(1 + \frac{\beta^2}{128M^4})^2}$$

- We use the conventional notation that for any $T \in \mathbb{N}$, $\mathbb{N}_T = \{1, \ldots, T\}$.
- Let $\mathbf{w}^*$ denote the optimal solution of formulation (8), i.e.,

$$\mathbf{w}^* = \arg\min_{\mathbf{w} \in \mathbb{R}^d} \{\mathbb{E}[F(\mathbf{w}, a(\mathbf{w}), b(\mathbf{w}), \alpha(\mathbf{w}); z_t)] + \Omega(\mathbf{w})\}.$$

Convergence Analysis

Theorem

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

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

Convergence Analysis

Theorem

Under the assumptions of (A1), (A2), and choosing step sizes $\{\eta_t = [\widetilde{C}_{\beta,M}(t+1)]^{-1} : t \in \mathbb{N}\}$, the algorithm SPAM achieves the following:

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

- The convergence of SPAM can achieve $\mathcal{O}(1/T)$ up to a logarithmic term, which matches the optimal rate of standard stochastic gradient descent

Outline

Motivation
- Why AUC?
- What is AUC?

AUC Maximization
- AUC Optimization
- Stochastic Proximal AUC Maximization Algorithm
- Convergence Analysis

Experiments

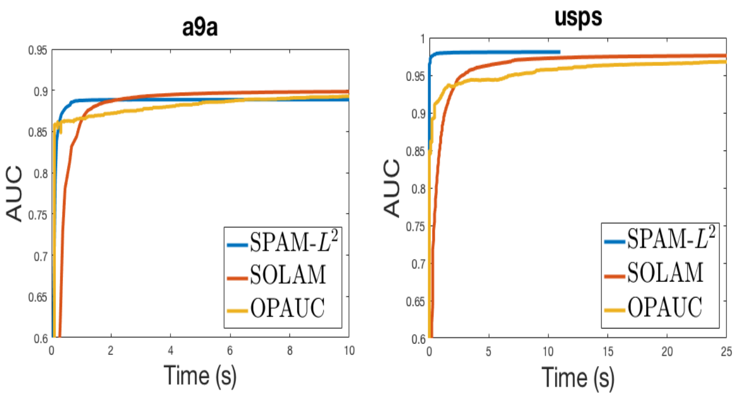
Conclusion

Evaluation on Test Data

- Online Learning: OPAUC [Gao et al. (2013)]; OAMseq and OAMgra [Zhao et al. (2011)]
- Batch Learning: B-SVM-OR and B-LS-SVM [T. Joachims (2006)]

Data	SPAM-L^2	SPAM-NET	SOLAM	OPAUC	OAM_{seq}	OAM_{gra}	B-LS-SVM
diabetes	.8272±.0277	.8085±.0431	.8128±.0304	.8309±.0350	.8264±.0367	.8262±.0338	.8325±.0329
fourclass	.8210±.0203	.8211±.0205	.8213±.0209	.8310±.0251	.8306±.0247	.8295±.0251	.8309±.0309
german	.7942±.0388	.7937±.0386	.7778±.0373	.7978±.0347	.7747±.0411	.7723±.0358	.7994±.0343
splice	.9263±.0091	.9267±.0090	.9246±.0087	.9232±.0099	.8594±.0194	.8864±.0166	.9245±.0092
usps	.9868±.0032	.9855±.0029	.9822±.0036	.9620±.0040	.9310±.0159	.9348±.0122	.9634±.0045
a9a	.8998±.0046	.8980±.0047	.8966±.0043	.9002±.0047	.8420±.0174	.8571±.0173	.8982±.0028
mnist	.9254±.0025	.9132±.0026	.9118±.0029	.9242±.0021	.8615±.0087	.8643±.0112	.9336±.0025
acoustic	.8120±.0030	.8109±.0028	.8099±.0036	.8192±.0032	.7113±.0590	.7711±.0217	.8210±.0033
ijcnn1	.9174±.0024	.9155±.0024	.9129±.0030	.9269±.0021	.9209±.0079	.9100±.0092	.9320±.0037
covtype	.9504±.0011	.9508±.0011	.9503±.0012	.8244±.0014	.7361±.0317	.7403±.0289	.8222±.0014
sector	.8768±.0126	.9077±.0104	.8767±.0129	.9292±.0081	.9163±.0087	.9043±.0100	-
news20	.8708±.0069	.8704± .0070	.8712±.0073	.8871±.0083	.8543±.0099	.8346±.0094	-

Running Time Comparison



Conclusion

- 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.

Thank you for attending!