OPT2020: 12th Annual Workshop on Optimization for Machine Learning

Variance Reduced Stochastic Proximal Algorithm for AUC Maximization

Soham Dan* SOHAMDAN@SEAS.UPENN.EDU
Dushyant Sahoo* SADU@SEAS.UPENN.EDU
University of Pennsylvania

Abstract

Stochastic Gradient Descent (SGD) has been widely studied with classification accuracy as a performance measure. However, these algorithms are not applicable when non-decomposable pairwise performance measures are used, such as Area under the ROC curve (AUC)—a standard performance metric when the classes are imbalanced. Recently, a Stochastic Proximal Gradient Algorithm (SPAM) has been proposed to optimize AUC. However, SPAM suffers from high variance leading to slower convergence. In this paper, we develop a Variance Reduced Stochastic Proximal algorithm for AUC Maximization (VRSPAM). We show that our algorithm converges faster than SPAM both theoretically and empirically.

1. Introduction

Class imbalance poses a challenge in several domains [5] for instance, medical diagnosis of rare diseases. Classification accuracy is not an appropriate performance metric in this setting, as predicting the majority class will give a high classification accuracy. AUC is commonly used to evaluate the performance of a binary classifier in this setting [6]. AUC measures the ability of a family of classifiers to correctly rank an example from the positive class with respect to a randomly selected example from the negative class.

From an optimization perspective, the AUC metric is non-convex and thus, hard to optimize. Instead it is attractive to optimize the convex surrogate such as, the pairwise squared surrogate [1, 8, 14]. Further, the AUC metric does not decompose over individual datapoints unlike classification accuracy —in each step, an algorithm needs to pair the current datapoint with all previously seen datapoints leading to $\mathcal{O}(td)$ space and time complexity at step t, where the dimension of the instance space is d. Recently, Ying et al. [18] reformulated the pairwise squared loss surrogate of AUC as a saddle point problem and gave an algorithm that has a convergence rate of $\mathcal{O}\left(\frac{1}{\sqrt{t}}\right)$. Natole et al. [15] improved on this by achieving a convergence rate of $\mathcal{O}\left(\frac{\log t}{t}\right)$, under strong convexity. It has per iteration complexity of $\mathcal{O}(d)$ and applies to general, non-smooth regularization terms. Both these rates are sub-optimal to the linear rate SGD achieves with classification accuracy as a performance measure The slow convergence is caused by the high variance of the gradient in each iteration in both [15],[18].

In the context of classification accuracy, techniques to reduce the variance of SGD have been proposed—SAG [16], SVRG [10] and it's proximal variant [17]. In this paper, we present Variance

*. These authors contributed equally to this work

Reduced Stochastic Proximal algorithm for AUC Maximization (VRSPAM). VRSPAM extends previous work for surrogate-AUC maximization by using the Proximal SVRG algorithm [17]. We show that VRSPAM achieves a linear convergence rate with a fixed step size which is better than the sublinear rate of exisiting algorithms [15],[18]. Numerical experiments demonstrate faster convergence of VRSPAM.

2. AUC formulation

The AUC score associated with a linear scoring function $g(x) = \mathbf{w}^T x$, is defined as the probability that the score of a randomly chosen positive example is higher than a randomly chosen negative example [4] and is denoted by AUC($\mathbf{w}$). If $z = (x, y)$ and $z' = (x', y')$ are drawn independently from an unknown distribution $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$, then

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

Since AUC($\mathbf{w}$) in the above form is not convex because of the 0-1 loss, it is a common practice to replace this by a convex surrogate loss. In this paper, we use the least square loss which is known to be consistent (minimizing the surrogate loss function, maximizes the AUC). Let $f(\mathbf{w}) = p(1-p)\,\mathbb{E}[(1 - \mathbf{w}^T(x - x'))^2 | y = 1, y' = -1]$ and Ω be the convex regularizer where $p = Pr(y = +1)$ and $1 - p = Pr(y = -1)$ are the class priors. We consider the following objective for surrogate-AUC maximization :

$$\min_{\mathbf{w} \in \mathbb{R}^d} f(\mathbf{w}) + \Omega(\mathbf{w}) \tag{1}$$

The form for $f(\mathbf{w})$ follows from the definition of AUC : expected pairwise loss between a positive instance and a negative instance. Throughout this paper we assume a) Ω is β strongly convex and b) $\exists M$ such that $\|x\| \leq M \; \forall x \in \mathcal{X}$. We use the Frobenius norm $\Omega(\mathbf{w}) = \beta\|\mathbf{w}\|^2$ and Elastic Net $\Omega(\mathbf{w}) = \beta\|\mathbf{w}\|^2 + \nu\|\mathbf{w}\|_1$ as the convex regularizers where $\beta, \nu \neq 0$ are the regularization parameters. The minimization problem in (1) can be reformulated such that stochastic gradient descent can be performed to find the optimum value. Below is an equivalent formulation from Theorem B in Natole et al. [15]-

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

where the expectation is with respect to $z = (x, y)$ and $F(\mathbf{w}, a, b, \zeta; z) = (1-p)(\mathbf{w}^T x - a)^2 \mathbb{I}_{[y=1]} + p(\mathbf{w}^T x - b)^2 \mathbb{I}_{[y=-1]} + 2(1+\zeta)\mathbf{w}^T x (p\mathbb{I}_{[y=-1]} - (1-p)\mathbb{I}_{[y=1]}) - p(1-p)\zeta^2$.
Thus, $f(\mathbf{w}) = \min_{a,b} \max_{\zeta \in R} \mathbb{E}[F(\mathbf{w}, a, b, \zeta; z)]$. Natole et al. [15] also state that the optimal choices for a, b, ζ satisfy (note all are functions of $\mathbf{w}$) :

- $a = \mathbf{w}^T\,\mathbb{E}[x | y = 1]$ • $b = \mathbf{w}^T\,\mathbb{E}[x | y = -1]$ • $\zeta = \mathbf{w}^T(\mathbb{E}[x' | y' = -1] - \mathbb{E}[x | y = 1])$

It is important to note here that we differentiate the objective function only with respect to $\mathbf{w}$ and do not compute the gradient with respect to the other parameters which themselves depend on $\mathbf{w}$. This is the reason why existing methods cannot be applied directly.

3. Method

The major issue that slows down convergence for SGD is the decay of the step size to 0 as the iteration increases. This is necessary for mitigating the effect of variance introduced by random

sampling in SGD. We apply the Prox-SVRG method Xiao and Zhang [17] on the reformulation of AUC to derive the proximal SVRG algorithm for AUC maximization presented in Algorithm 1. We store $\tilde{\mathbf{w}}$ after every m Prox-SGD iterations that is progressively closer to the optimal $\mathbf{w}$ (essentially an estimate of the optimal value of (1), $\mathbf{w}^*$). Full gradient $\tilde{\boldsymbol{\mu}} = \frac{1}{n}\sum_{i=1}^{n} G(\tilde{\mathbf{w}}, z_i)$ is computed whenever $\tilde{\mathbf{w}}$ gets updated—after every m iterations of Prox-SGD: where $G(\mathbf{w}; z) = \partial_{\mathbf{w}} F(\mathbf{w}, a(\mathbf{w}), b(\mathbf{w}), \zeta(\mathbf{w}); z)$ and $\tilde{\boldsymbol{\mu}}$ is used to update next m gradients. Next m iterations are initialized by $\mathbf{w}_0 = \tilde{\mathbf{w}}$. For each iteration, we randomly pick $i_t \in \{1, ..., n\}$ and compute $\hat{\mathbf{w}}_t = \mathbf{w}_{t-1} - \eta \mathbf{v}_{t-1}$ where $\mathbf{v}_t = G(\mathbf{w}_t, z_{i_{t-1}}) - G(\tilde{\mathbf{w}}, z_{i_{t-1}}) + \tilde{\boldsymbol{\mu}}$ and then the proximal step is taken: $\mathbf{w}_t = \text{prox}_{\eta,\Omega}(\hat{\mathbf{w}}_t)$.

Algorithm 1 Proximal SVRG for AUC maximization

INPUT Constant step size η and update frequency m
INITIALIZE $\tilde{\mathbf{w}}_0$
for $s = 1, 2, ...$ **do**
  $\tilde{\mathbf{w}} = \tilde{\mathbf{w}}_{s-1}$
  $\tilde{\boldsymbol{\mu}} = \frac{1}{n}\sum_{i=1}^{n} G(\tilde{\mathbf{w}}, z_i)$
  $\mathbf{w}_0 = \tilde{\mathbf{w}}$
  for $t = 1, 2, ..., m$ **do**
    Randomly pick $i_t \in \{1, .., n\}$ and update weight
    $\hat{\mathbf{w}}_t = \mathbf{w}_{t-1} - \eta(G(\mathbf{w}_{t-1}, z_{i_t}) - G(\tilde{\mathbf{w}}, z_{i_t}) + \tilde{\boldsymbol{\mu}})$
    $\mathbf{w}_t = \text{prox}_{\eta\Omega}(\hat{\mathbf{w}}_t)$
  end
  $\tilde{\mathbf{w}}_s = \mathbf{w}_m$
end

Notice that if we take expectation of $G(\tilde{\mathbf{w}}, z_{i_{t-1}})$ with respect to i_t we get $\mathbb{E}[G(\tilde{\mathbf{w}}, z_{i_{t-1}})] = \tilde{\boldsymbol{\mu}}$. Now if we take expectation of $\mathbf{v}_t$ with respect to i_t conditioned on $\mathbf{w}_{t-1}$, we can get the following:

$$\mathbb{E}[\mathbf{v}_t|\mathbf{w}_{t-1}] = \mathbb{E}[G(\mathbf{w}, z_{i_{t-1}})] - \mathbb{E}[G(\tilde{\mathbf{w}}, z_{i_{t-1}})] + \tilde{\boldsymbol{\mu}}$$
$$= \frac{1}{n}\sum_{i=1}^{n} G(\tilde{\mathbf{w}}_{t-1}, z_i)$$

Hence the modified direction $\mathbf{v}_t$ is stochastic gradient of G at $\mathbf{w}_{t-1}$. However, the variance $\mathbb{E}\,\|\mathbf{v}_t - \partial f(\mathbf{w}_{t-1})\|^2$ can be much smaller than $\mathbb{E}\,\|G(\mathbf{w}_{t-1}, z_{i_{t-1}}) - \partial f(\mathbf{w}_{t-1})\|^2$ which we will show in section 4. We will also show that the variance goes to 0 as the algorithm converges. Thus, this is a multi-stage scheme to explicitly reduce the variance of the modified proximal gradient.

4. Convergence Analysis

We present a lemma giving the bound on the variance of modified gradient $\mathbf{v}_t$. Proof of the lemma is given in Appendix C.

Lemma 1 *Consider* VRSPAM *(Algorithm 1), then the variance of the* $\mathbf{v}_t$ *is upper bounded as:*

$$\mathbb{E}[\|\mathbf{v}_t - \partial f(\mathbf{w}_t)]\|^2]) \leq 4(8M^2)^2\|\mathbf{w}_t - \mathbf{w}^*\|^2 + 2(8M^2)^2\|\tilde{\mathbf{w}} - \mathbf{w}^*\|^2$$

At the convergence, $\tilde{\mathbf{w}} = \mathbf{w}^*$ and $\mathbf{w}_t = \mathbf{w}^*$. Thus, the variance of the updates are bounded and go to zero as the algorithm converges. Whereas, in the case of the SPAM algorithm in [15], the variance of the gradient does not go to zero as it is a stochastic gradient descent based algorithm. The following is the main theorem of this paper stating the convergence rate of Algorithm 1 and the proof is in Appendix B.

Theorem 2 *Consider* VRSPAM *(Algorithm 1) and let* $\mathbf{w}^* = \arg\min_{\mathbf{w}} f(\mathbf{w}) + \Omega(\mathbf{w})$*; if* $\eta < \frac{\beta}{128M^4}$*, then there exists* $\alpha < 1$ *and we have the geometric convergence in expectation:*

$$\mathbb{E}[\|\tilde{\mathbf{w}}_s - \mathbf{w}^*\|^2] \leq \alpha^s\, \mathbb{E}[\|\mathbf{w_0} - \mathbf{w}^*\|^2]$$

Hence, we get a geometric convergence rate of α^s which is much stronger than the $\mathcal{O}(\frac{1}{t})$ convergence rate obtained in Natole et al. [15]. In the next section we derive the time complexity of the algorithm and investigate dependence of α on the problem parameters.

4.1. Complexity analysis

Using Theorem 2, the number of iterations s required is $\geq \frac{1}{\log \frac{1}{\alpha}} \log \frac{\mathbb{E}\,\|\mathbf{w}_0 - \mathbf{w}^*\|^2}{\epsilon}$ to get $\mathbb{E}\,\|\tilde{\mathbf{w}}_s - \mathbf{w}^*\|^2 \leq \epsilon$. At each stage, the number of gradient evaluations are $n + 2m$ where n is the number of samples and m is the iterations in the inner loop and the complexity is $\mathcal{O}(n+m)(\log(\frac{1}{\epsilon}))$ i.e. Algorithm 1 takes $\mathcal{O}(n+m)(\log(\frac{1}{\epsilon}))$ gradient complexity to achieve accuracy of ϵ. Here, the complexity is dependent on M and β as m itself is dependent on M and β.

Using the corollary proved in Appendix D, for any $0 < \theta < 1$ and $E = \frac{1}{(1+\frac{\theta\beta^2}{128M^4})}$, if we choose $m \approx 2\frac{\log\theta}{\log E}$ then $\alpha \approx 2\theta E^2$, which is independent of m. Thus the time complexity of the algorithm is $\mathcal{O}(n + 2\frac{\log\theta}{\log E})(\log(\frac{1}{\epsilon}))$ when $m = \Theta(\frac{\log\theta}{\log E})$. As the order has inverse dependency on $\log E = \log \frac{128M^4}{128M^4+\theta\beta^2}$, increase in M will result in increase in number of iterations i.e. as the maximum norm of training samples is increased, larger m is required to reach ϵ accuracy.

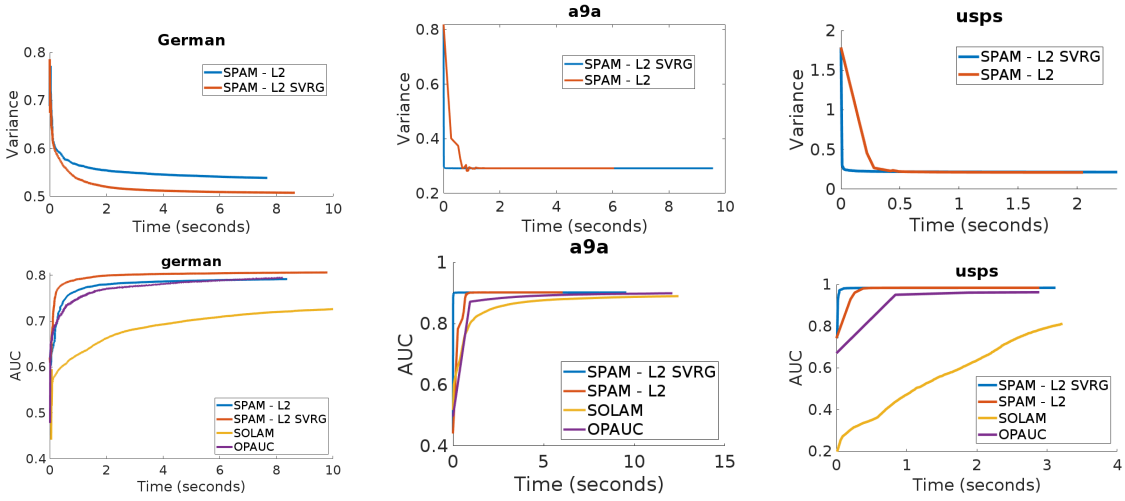

Figure 1: The top row shows that VRSPAM (SPAM-L2-SVRG) has lower variance than SPAM-L2 across different datasets. The bottom row shows VRSPAM (SPAM-L2-SVRG) converges faster and performs better than existing algorithms on AUC maximization.

Now we will compare the time complexity of our algorithm with SPAM algorithm. First, we find the time complexity of SPAM. We will use Theorem 3 from Natole et al. [15] which states that SPAM achieves the following:

$$\mathbb{E}[\|\mathbf{w}_{T+1} - \mathbf{w}^*\|^2] \leq \frac{t_0}{T}\,\mathbb{E}[\|\mathbf{w}_{t_0} - \mathbf{w}^*\|^2] + c\frac{\log T}{T}$$

where $t_0 = \max\big(2, \lceil 1 + \frac{(128M^4+\beta^2)^2}{128M^4\beta^2}\rceil\big)$, T is the number of iterations and c is a constant. Through averaging scheme developed by Lacoste-Julien et al. [11] the following can be obtained:

$$\mathbb{E}[\|\mathbf{w}_{T+1} - \mathbf{w}^*\|^2] \leq \frac{t_0}{T}\,\mathbb{E}[\|\mathbf{w}_{t_0} - \mathbf{w}^*\|^2] \tag{2}$$

where $\mathbb{E}[\|\mathbf{w}_{t_0} - \mathbf{w}^*\|^2] \leq \frac{2\sigma_*^2}{\tilde{C}^2_{\beta,M}} + \exp\big(\frac{128M^4}{\tilde{C}^2_{\beta,M}}\big) = F$, $\tilde{C}^2_{\beta,M} = \frac{\beta}{(1+\frac{\beta^2}{128M^4})^2}$ and $\mathbb{E}[\|G(\mathbf{w}^*; z) - \partial f(\mathbf{w}^*)\|^2] = \sigma_*^2$. Using Equation 2, time complexity of SPAM algorithm can be written as $\mathcal{O}(\frac{t_0 F}{\epsilon})$

i.e. SPAM algorithm takes $\mathcal{O}(\frac{t_0 F}{\epsilon})$ iterations to achieve ϵ accuracy. Thus, SPAM has lower per iteration complexity but slower convergence rate as compared to VRSPAM. Therefore, VRSPAM will take less time to get a good approximation of the solution.

Name	VRSPAM-L^2	VRSPAM-NET	SPAM-L^2	SPAM-NET	SOLAM	OPAUC
DIABETES	.8299±.0323	**.8305±.0319**	.8272±.0277	.8085±.0431	.8128±.0304	.8309±.0350
GERMAN	.7902±0386	.7845±.0398	.7942±.0388	.7937±.0386	.7778±.0373	**.7978±.0347**
SPLICE	.9640±.0156	**.9699±.0139**	.9263±.0091	.9267±.0090	.9246±.0087	.9232±.0099
USPS	**.8552±.006**	.8549±.0059	.8542±.0388	.8537±.0386	.8395±.0061	.8114±.0065
LETTER	.9834±.0023	.9804±.0032	**.9868±.0032**	.9855±.0029	.9822±.0036	.9620±.0040
A9A	**.9003±.0045**	.8981±.0046	.8998±.0046	.8980±.0047	.8966±.0043	.9002±.0047
W8A	**.9876±.0008**	.9787±.0013	.9682±.0020	.9604±.0020	.9817±.0015	.9633±.0035
MNIST	**.9465±.0014**	.9351±.0014	.9254±.0025	.9132±.0026	.9118±.0029	.9242±.0021
ACOUSTIC	.8093±.0033	.8052±.033	.8120±.0030	.8109±.0028	8099±.0036	**.8192±.0032**
IJCNN1	**.9750±.001**	.9745±.002	.9174±.0024	.9155±.0024	.9129±.0030	.9269±.0021

Table 1: AUC values (mean±std) comparison for different algorithms on test data. Best values are in bold.

5. Experiment

Here we empirically compare VRSPAM with other existing algorithms used for AUC maximization. We use the following two variants of our proposed algorithm based on the regularizer used:

- VRSPAM $- L^2 : \Omega(\mathbf{w}) = \frac{\beta}{2}\|\mathbf{w}\|^2$ (Frobenius Norm Regularizer)
- VRSPAM$-NET : \Omega(\mathbf{w}) = \frac{\beta}{2}\|\mathbf{w}\|_2^2 + \beta_1\|\mathbf{w}\|_1$ (Elastic Net Regularizer [19]). The proximal step for elastic net is given as $\arg\min_{\mathbf{w}}\{\frac{1}{2}\|\mathbf{w} - \frac{\hat{\mathbf{w}}_{t+1}}{\eta_t\beta+1}\|^2 + \frac{\eta_t\beta_1}{\eta_t\beta+1}\|\mathbf{w}\|_1\}$

VRSPAM is compared with SPAM, SOLAM [18] and one-pass AUC optimization algorithm (OPAUC) [9]. SOLAM was modified to have the Frobenius Norm Regularizer [15]. VRSPAM is compared against OPAUC with the least square loss. Details of the datasets are described Table 2 in Appendix E.

- Variance results: In the top row of Figure 1, we show the variance of the VRSPAM update ($\mathbf{v}_t$) in comparison with the variance of SPAM update ($G(\mathbf{w}_{t-1}, z_{i_{t-1}})$) . We observe that the variance of VRSPAM is lower than the variance of SPAM and decreases to the minimum value faster, which is in line with Theorem 2.
- Convergence results: In the bottom row of Figure 1, we show the performance of our algorithm compared to existing methods for AUC maximization. We observe that VRSPAM converges to the maximum AUC value faster than the other methods, and on several datasets, VRSPAM achieves better test AUC value than other methods, as seen in Table 1.

Note that, the initial weights of VRSPAM are set to be the output generated by SPAM after one iteration, which is standard practice [10]. Table 1 summarizes the AUC evaluation for different algorithms. AUC values for SPAM-L^2, SPAM-NET, SOLAM and OPAUC are from [15].

6. Conclusion

In this paper, we propose a variance reduced stochastic proximal algorithm for AUC maximization (VRSPAM). We theoretically analyze the proposed algorithm and derive a much faster convergence rate of $\mathcal{O}(\alpha^t)$ where $\alpha < 1$ (linear convergence rate), improving upon state-of-the-art methods Natole et al. [15] which have a convergence rate of $\mathcal{O}(\frac{1}{t})$ (sub-linear convergence rate), for strongly convex objective functions with per iteration complexity of one data-point. We showed theoretically and empirically VRSPAM converges faster than existing methods for AUC maximization.

References

[1] Shivani Agarwal. Surrogate regret bounds for the area under the roc curve via strongly proper losses. In *Conference on Learning Theory*, pages 338–353, 2013.

[2] Heinz H Bauschke, Patrick L Combettes, et al. *Convex analysis and monotone operator theory in Hilbert spaces*, volume 408. Springer, 2011.

[3] Chih-Chung Chang and Chih-Jen Lin. Libsvm: A library for support vector machines. *ACM transactions on intelligent systems and technology (TIST)*, 2(3):27, 2011.

[4] Stéphan Clémençon, Gábor Lugosi, Nicolas Vayatis, et al. Ranking and empirical minimization of u-statistics. *The Annals of Statistics*, 36(2):844–874, 2008.

[5] Charles Elkan. The foundations of cost-sensitive learning. In *International joint conference on artificial intelligence*, volume 17, pages 973–978. Lawrence Erlbaum Associates Ltd, 2001.

[6] Tom Fawcett. An introduction to roc analysis. *Pattern recognition letters*, 27(8):861–874, 2006.

[7] Andrew Frank and Arthur Asuncion. Uci machine learning repository [http://archive. ics. uci. edu/ml]. irvine, ca: University of california. *School of information and computer science*, 213: 2–2, 2010.

[8] Wei Gao and Zhi-Hua Zhou. On the consistency of auc pairwise optimization. In *Twenty-Fourth International Joint Conference on Artificial Intelligence*, 2015.

[9] Wei Gao, Rong Jin, Shenghuo Zhu, and Zhi-Hua Zhou. One-pass auc optimization. In *International Conference on Machine Learning*, pages 906–914, 2013.

[10] Rie Johnson and Tong Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In *Advances in neural information processing systems*, pages 315–323, 2013.

[11] Simon Lacoste-Julien, Mark Schmidt, and Francis Bach. A simpler approach to obtaining an o (1/t) convergence rate for the projected stochastic subgradient method. *arXiv preprint arXiv:1212.2002*, 2012.

[12] Mingrui Liu, Xiaoxuan Zhang, Zaiyi Chen, Xiaoyu Wang, and Tianbao Yang. Fast stochastic auc maximization with o (1/n)-convergence rate. In *International Conference on Machine Learning*, pages 3195–3203, 2018.

[13] Mingrui Liu, Zhuoning Yuan, Yiming Ying, and Tianbao Yang. Stochastic auc maximization with deep neural networks. *arXiv preprint arXiv:1908.10831*, 2019.

[14] Harikrishna Narasimhan and Shivani Agarwal. Support vector algorithms for optimizing the partial area under the roc curve. *Neural computation*, 29(7):1919–1963, 2017.

[15] Michael Natole, Yiming Ying, and Siwei Lyu. Stochastic proximal algorithms for auc maximization. In *International Conference on Machine Learning*, pages 3707–3716, 2018.

[16] Nicolas L Roux, Mark Schmidt, and Francis R Bach. A stochastic gradient method with an exponential convergence _rate for finite training sets. In *Advances in neural information processing systems*, pages 2663–2671, 2012.

[17] Lin Xiao and Tong Zhang. A proximal stochastic gradient method with progressive variance reduction. *SIAM Journal on Optimization*, 24(4):2057–2075, 2014.

[18] Yiming Ying, Longyin Wen, and Siwei Lyu. Stochastic online auc maximization. In *Advances in neural information processing systems*, pages 451–459, 2016.

[19] Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net. *Journal of the royal statistical society: series B (statistical methodology)*, 67(2):301–320, 2005.

Appendix A. Helper lemmas

We first define some lemmas which will be used for proving the Theorem 2 which is the main theorem proving the geometric convergence of Algorithm 1. First is the Lemma 3 from Natole et al. [15] which states that $\partial_{\mathbf{w}} F(\mathbf{w}_t, a(\mathbf{w}_t), b(\mathbf{w}_t), \alpha(\mathbf{w}_t); z_t)$ is an unbiased estimator of the true gradient. As we are not calculating the true gradient in VRSPAM, we need the following Lemma to prove the convergence result.

Lemma 3 ([15]) *Let* $\mathbf{w}_t$ *be given by* VRSPAM *in Algorithm 1. Then, we have*

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

This Lemma is directly applicable in VRSPAM since the proof of the Lemma hinges on the objective function formulation and not on the algorithm specifics.

The next lemma provides an upper bound on the norm of difference of gradients at different time steps.

Lemma 4 ([15]) *Let* $\mathbf{w}_t$ *be described in the main paper. Then, we have*

$$\|G(\mathbf{w}_{t'}; z_t) - G(\mathbf{w}_t; z_t)\| \leq 8M^2 \|\mathbf{w}_{t'} - \mathbf{w}_t\|$$

Proof

$$\begin{aligned}\|G(\mathbf{w}_{t'}; z_t) - G(\mathbf{w}_t; z_t)\| &\leq 4M^2 p \|\mathbf{w}_{t'} - \mathbf{w}_t\| \mathbb{1}_{[y_t=-1]} + 4M^2(1-p)\|\mathbf{w}_{t'} - \mathbf{w}_t\| \mathbb{1}_{[y_t=1]} \\ &\quad + 4M^2 p \|\mathbf{w}_{t'} - \mathbf{w}_t\| \mathbb{1}_{[y_t=-1]} + 4M^2 |p - \mathbb{1}_{[y_t=1]}| \|\mathbf{w}_{t'} - \mathbf{w}_t\| \\ &\leq 8M^2 \|\mathbf{w}_{t'} - \mathbf{w}_t\|\end{aligned}$$

The proof directly follows by writing out the difference and using the second assumption on the boundedness of $\|x\|$. ■

We now present and prove a result that will be necessary in showing convergence in Theorem 2

Lemma 5 *Let* $C = \frac{1+128M^4\eta^2}{(1+\eta\beta)^2}$ *and* $D = \frac{128M^4\eta^2}{(1+\eta\beta)^2}$*; if* $\eta \leq \frac{\beta}{128M^4}$ *then* $C^m + DC\frac{C^m - 1}{C-1} \leq 1$ *holds true.*

Proof We start with:

$$\begin{aligned}&\eta \leq \frac{\beta}{128M^4} \\ \Rightarrow &128M^4\eta^2 \leq \eta\beta \\ \Rightarrow &128M^4\eta^2(2 + 128M^4\eta^2) \leq \eta\beta(2 + 1\eta\beta) \\ \Rightarrow &128M^4\eta^2 + (128M^4\eta^2)^2 \leq (\eta\beta)^2 + 2\eta\beta - 128M^4\eta^2 \\ \Rightarrow &128M^4\eta^2 \leq \frac{(1+\eta\beta)^2 - 1 - 128M^4\eta^2}{1 + 128M^4\eta^2} \\ \Rightarrow &128M^4\eta^2 \leq \frac{1 - \frac{1+128M^4\eta^2}{(1+\eta\beta)^2}}{\frac{1+128M^4\eta^2}{(1+\eta\beta)^2}}\end{aligned}$$

Substituting values of C and D and using the condition that $D \leq 128M^4\eta^2$, we get

$$\begin{aligned}
&\Rightarrow D \leq \frac{1-C}{C} \\
&\Rightarrow DC\frac{C^m-1}{C-1} \leq 1-C^m \\
&\Rightarrow C^m + DC\frac{C^m-1}{C-1} \leq 1
\end{aligned}$$

■

Appendix B. Proof of Theorem 1

From the first order optimality condition, we can directly write

$$\mathbf{w}^* = \text{prox}_{\eta\Omega}(\mathbf{w}^* - \eta\partial f(\mathbf{w}^*))$$

Using the above we can write

$$\|\mathbf{w}_{t+1} - \mathbf{w}^*\|^2 = \|\,\text{prox}_{\eta\Omega}(\hat{\mathbf{w}}_{t+1}) - \text{prox}_{\eta\Omega}(\mathbf{w}^* - \eta\partial f(\mathbf{w}^*))\|^2$$

Using Proposition 23.11 from Bauschke et al. [2], we have $\text{prox}_{\eta\Omega}$ is $(1+\eta\beta)$-cocoercieve and for any $\mathbf{u}$ and $\mathbf{w}$ using Cauchy Schwartz we can get the following inequality

$$\|\,\text{prox}_{\eta\Omega}(\mathbf{u}) - \text{prox}_{\eta\Omega}(\mathbf{w})\| \leq \frac{1}{1+\eta\beta}\|\mathbf{u}-\mathbf{w}\|$$

From above we get

$$\begin{aligned}
\|\mathbf{w}_{t+1} - \mathbf{w}^*\|^2 &\leq \frac{1}{(1+\eta\beta)^2}\|(\hat{\mathbf{w}}_{t+1}) - (\mathbf{w}^* - \eta\partial f(\mathbf{w}^*))\|^2 \\
&\leq \frac{1}{(1+\eta\beta)^2}\|(\mathbf{w}_t - \mathbf{w}^*) - \eta(G(\mathbf{w}_t, z_{i_t}) - G(\tilde{\mathbf{w}}, z_{i_t}) + \tilde{\boldsymbol{\mu}} - \partial f(\mathbf{w}^*))\|^2
\end{aligned}$$

Taking expectation on both sides we get

$$\begin{aligned}
\mathbb{E}\,\|\mathbf{w}_{t+1} - \mathbf{w}^*\|^2 \leq \frac{1}{(1+\eta\beta)^2}\big(&\eta^2\,\mathbb{E}[\|G(\mathbf{w}_t, z_{i_t}) - G(\tilde{\mathbf{w}}, z_{i_t}) + \tilde{\boldsymbol{\mu}} - \partial f(\mathbf{w}^*))\|^2] \\
&+ \mathbb{E}[\|\mathbf{w}_t - \mathbf{w}^*\|^2] - 2\eta\,\mathbb{E}[\langle \mathbf{w}_t - \mathbf{w}^*, G(\mathbf{w}_t, z_{i_t}) - G(\tilde{\mathbf{w}}, z_{i_t}) + \tilde{\boldsymbol{\mu}} - \partial f(\mathbf{w}^*)\rangle]\big)
\end{aligned} \tag{3}$$

Now, we first bound the last term $T = \mathbb{E}[\langle \mathbf{w}_t - \mathbf{w}^*, G(\mathbf{w}_t, z_{i_t}) - G(\tilde{\mathbf{w}}, z_{i_t}) + \tilde{\boldsymbol{\mu}} - \partial f(\mathbf{w}^*)\rangle]$ in equation 3. Using Lemma 5 we can write

$$\begin{aligned}
T &= \mathbb{E}[\langle \mathbf{w}_t - \mathbf{w}^*, \mathbb{E}_{z_t}[G(\mathbf{w}_t, z_{i_t})] - \mathbb{E}_{z_t}[G(\tilde{\mathbf{w}}, z_{i_t})] + \tilde{\boldsymbol{\mu}} - \partial f(\mathbf{w}^*)\rangle] \\
&= \mathbb{E}[\langle \mathbf{w}_t - \mathbf{w}^*, \mathbb{E}_{z_t}[G(\mathbf{w}_t, z_{i_t})] - \partial f(\mathbf{w}^*)\rangle] \\
&= \mathbb{E}[\langle \mathbf{w}_t - \mathbf{w}^*, \partial f(\mathbf{w}_t) - \partial f(\mathbf{w}^*)\rangle] \\
&\geq 0
\end{aligned}$$

Now, $\mathbb{E}\,\|\mathbf{w}_{t+1}-\mathbf{w}^*\|^2$ can be bounded by using above bound and Lemma 6 as below

$$\begin{aligned}\mathbb{E}\,\|\mathbf{w}_{t+1}-\mathbf{w}^*\|^2 &\leq \frac{1}{(1+\eta\beta)^2}(\mathbb{E}[\|\mathbf{w}_t-\mathbf{w}^*\|^2]+128M^4\eta^2(\mathbb{E}[\|\mathbf{w}_t-\mathbf{w}^*\|^2]+\mathbb{E}[\|\tilde{\mathbf{w}}-\mathbf{w}^*\|^2]))\\ &\leq \frac{1+128M^4\eta^2}{(1+\eta\beta)^2}\,\mathbb{E}[\|\mathbf{w}_t-\mathbf{w}^*\|^2]+\frac{128M^4\eta^2}{(1+\eta\beta)^2}\,\mathbb{E}[\|\tilde{\mathbf{w}}-\mathbf{w}^*\|^2]\end{aligned}$$

Let $C=\frac{1+128M^4\eta^2}{(1+\eta\beta)^2}$ and $D=\frac{128M^4\eta^2}{(1+\eta\beta)^2}$, then after m iterations $\mathbf{w}_t=\tilde{\mathbf{w}}_s$ and $\mathbf{w_0}=\tilde{\mathbf{w}}_{s-1}$. Substituting this in the above inequality, we get

$$\begin{aligned}&\mathbb{E}\,\|\tilde{\mathbf{w}}_s-\mathbf{w}^*\|^2\\ &\leq C^m\big(\mathbb{E}\,\|\tilde{\mathbf{w}}_{s-1}-\mathbf{w}^*\|^2+\sum_{i=0}^{m-1}\frac{D}{C^i}\,\mathbb{E}\,\|\tilde{\mathbf{w}}_{s-1}-\mathbf{w}^*\|^2\big)\\ &\leq \big(C^m+\sum_{i=0}^{m-1}\frac{DC^m}{C^i}\big)\,\mathbb{E}\,\|\tilde{\mathbf{w}}_{s-1}-\mathbf{w}^*\|^2\\ &\leq \big(C^m+DC^m\frac{1-(1/C^m)}{1-(1/C)}\big)\,\mathbb{E}\,\|\tilde{\mathbf{w}}_{s-1}-\mathbf{w}^*\|^2\\ &\leq \big(C^m+DC\frac{C^m-1}{C-1}\big)\,\mathbb{E}\,\|\tilde{\mathbf{w}}_{s-1}-\mathbf{w}^*\|^2\\ &\leq \alpha\,\mathbb{E}\,\|\tilde{\mathbf{w}}_{s-1}-\mathbf{w}^*\|^2\end{aligned}$$

where $\alpha=C^m+DC\frac{C^m-1}{C-1}$ is the decay parameter, and $\alpha<1$ by using Lemma 5. After s steps in outer loop of Algorithm 1, we get $\mathbb{E}\,\|\tilde{\mathbf{w}}_s-\mathbf{w}^*\|^2\leq\alpha^s\,\mathbb{E}\,\|\mathbf{w}_0-\mathbf{w}^*\|^2$ where $\alpha<1$.

Appendix C. Bounding the variance

First we present a lemma what will be necessary to find the bound on the variance of modified gradient $\mathbf{v}_t=G(\mathbf{w}_t,z_{i_t})-G(\tilde{\mathbf{w}},z_{i_t})+\tilde{\boldsymbol{\mu}}$

Lemma 6 *Consider* VRSPAM *(Algorithm 1), then* $\mathbb{E}[\|\mathbf{v}_t-\partial f(\mathbf{w}^*)]\|^2$ *is upper bounded as:*

$$\mathbb{E}[\|\mathbf{v}_t-\partial f(\mathbf{w}^*)]\|^2]\leq 2(8M^2)^2\|\mathbf{w}_t-\mathbf{w}^*\|^2+2(8M^2)^2\|\tilde{\mathbf{w}}-\mathbf{w}^*\|^2$$

Proof Let the variance reduced update be denoted as $\mathbf{v}_t=G(\mathbf{w}_t,z_{i_t})-G(\tilde{\mathbf{w}},z_{i_t})+\tilde{\boldsymbol{\mu}}$. As we know $\mathbb{E}[\mathbf{v}_t]=\partial f(\mathbf{w}_t)$, the variance of $\mathbf{v}_k$ can be written as below

$$\begin{aligned}\mathbb{E}[\|G(\mathbf{w}_t,z_{i_t})-G(\tilde{\mathbf{w}},z_{i_t})+\tilde{\boldsymbol{\mu}}-\partial f(\mathbf{w}^*))\|^2]&\leq 2\,\mathbb{E}[\|G(\mathbf{w}_t,z_{i_t})-G(\mathbf{w}^*,z_{i_t})\|^2]\\ &\quad+2\,\mathbb{E}[\|G(\mathbf{w}^*,z_{i_t})-G(\tilde{\mathbf{w}},z_{i_t})+\tilde{\boldsymbol{\mu}}-\partial f(\mathbf{w}^*))\|^2]\end{aligned}$$

Also, $\mathbb{E}[G(\mathbf{w}^*,z_{i_t})-G(\tilde{\mathbf{w}},z_{i_t})]=\partial f(\mathbf{w}^*)-\partial f(\tilde{\mathbf{w}})$ from Lemma 3 and using the property that $\mathbb{E}[(X-\mathbb{E}[X])^2]\leq\mathbb{E}[X^2]$ we get

$$\begin{aligned}\mathbb{E}[\|G(\mathbf{w}_t,z_{i_t})-G(\tilde{\mathbf{w}},z_{i_t})+\tilde{\boldsymbol{\mu}}-\partial f(\mathbf{w}^*))\|^2]&\leq 2\,\mathbb{E}[\|G(\mathbf{w}_t,z_{i_t})-G(\mathbf{w}^*,z_{i_t})\|^2]\\ &\quad+2\,\mathbb{E}[\|G(\mathbf{w}^*,z_{i_t})-G(\tilde{\mathbf{w}},z_{i_t})\|^2]\end{aligned}$$

From Lemma 4, we have $\|G(\mathbf{w}_t, z_{i_t}) - G(\mathbf{w}^*, z_{i_t})\| \leq 8M^2\|\mathbf{w}_t - \mathbf{w}^*\|$ and $\|G(\mathbf{w}^*, z_{i_t}) - G(\tilde{\mathbf{w}}, z_{i_t})\| \leq 8M^2\|\tilde{\mathbf{w}} - \mathbf{w}^*\|$. Using this, we can upper bound the variance of gradient step as:

$$\mathbb{E}[\|G(\mathbf{w}_t, z_{i_t}) - G(\tilde{\mathbf{w}}, z_{i_t}) + \tilde{\boldsymbol{\mu}} - \partial f(\mathbf{w}^*))\|^2] \leq 2(8M^2)^2\|\mathbf{w}_t - \mathbf{w}^*\|^2 + 2(8M^2)^2\|\tilde{\mathbf{w}} - \mathbf{w}^*\|^2 \tag{4}$$

We have the desired result. ■

We now present a lemma giving the bound on the variance of modified gradient $\mathbf{v}_t$.

Lemma 7 *Consider* VRSPAM *(Algorithm 1), then the variance of the* $\mathbf{v}_t$ *is upper bounded as:*

$$\mathbb{E}[\|\mathbf{v}_t - \partial f(\mathbf{w}_t)]\|^2]) \leq 4(8M^2)^2\|\mathbf{w}_t - \mathbf{w}^*\|^2 + 2(8M^2)^2\|\tilde{\mathbf{w}} - \mathbf{w}^*\|^2$$

Proof

$$\begin{aligned}\mathbb{E}[\|\mathbf{v}_t - \partial f(\mathbf{w}_t)]\|^2] &\leq 2\,\mathbb{E}[\|\mathbf{v}_t - \partial f(\mathbf{w}^*)]\|^2] + 2\,\mathbb{E}[\|\partial f(\mathbf{w}^*) - \partial f(\mathbf{w}_t)]\|^2] \\ &\leq 2(8M^2)^2\|\mathbf{w}_t - \mathbf{w}^*\|^2 + 2(8M^2)^2\|\tilde{\mathbf{w}} - \mathbf{w}^*\|^2 + 2\,\mathbb{E}[\|G(\mathbf{w}_t, z_{i_t}) - G(\mathbf{w}^*, z_{i_t})\|^2] \\ &\leq 4(8M^2)^2\|\mathbf{w}_t - \mathbf{w}^*\|^2 + 2(8M^2)^2\|\tilde{\mathbf{w}} - \mathbf{w}^*\|^2\end{aligned}$$

where the second inequality uses Lemma 6 and last inequality uses Lemma 4. ■

Appendix D. Complexity analysis

Corollary 8 *Let* $E = \frac{1}{(1+\frac{\theta\beta^2}{128M^4})}$ *and* $0 < \theta < 1$*, if* $m \approx 2\frac{\log\theta}{\log E}$ *then* $\alpha \approx 2\theta E^2$

Proof Let $\eta = \frac{\theta\beta}{128M^4}$ where $0 < \theta < 1$, then

$$\begin{aligned}C &= \frac{1 + 128M^4\eta^2}{(1+\eta\beta)^2} = \frac{1 + \frac{\theta^2\beta^2}{128M^4}}{(1 + \frac{\theta\beta^2}{128M^4})^2} \\ &< \frac{1 + \frac{\theta\beta^2}{128M^4}}{(1 + \frac{\theta\beta^2}{128M^4})^2} \\ &= \frac{1}{(1 + \frac{\theta\beta^2}{128M^4})} \\ &= E\end{aligned}$$

therefore $D = \theta(E - E^2)$ and $DC < \theta E^2(1 - E)$, using the above equations we can simplify α as

$$\begin{aligned}\alpha &= C^m + DC\frac{1 - C^m}{1 - C} \\ &< C^m + \theta E^2(1 - E)\frac{1 - C^m}{1 - C} \\ &< C^m + \theta E^2(1 - C^m) \quad \because \frac{1 - E}{1 - C} < 1 \\ &= \theta E^2 + C^m - \theta E^2 C^m\end{aligned}$$

In the above equation, only $C^m - \theta E^2 C^m$ depends on m, if we choose m to be sufficiently large then $\alpha = \theta E^2$. An important thing to note here is that $\theta E < C < E$, now if we choose $m \approx 2\frac{\log\theta}{\log E}$ then $\alpha \approx 2\theta E^2$ which is independent of m. ■

N	Name	Instances	Features
1	DIABETES	768	8
2	GERMAN	1000	24
3	SPLICE	3,175	60
4	USPS	9,298	256
5	LETTER	20,000	16

Data	Name	Instances	Features
6	A9A	32,561	123
7	W8A	64,700	300
8	MNIST	60,000	780
9	ACOUSTIC	78,823	50
10	IJCNN1	141,691	22

Table 2: Datasets across which we evaluate our algorithm

Appendix E. Dataset

All datasets are publicly available from Chang and Lin [3] and Frank and Asuncion [7]. Table 2 provides detail about the dataset used in the experiments. Some of the datasets are multiclass, and we convert them to binary labels by numbering the classes and assigning all the even labels to one class and all the odd labels to another. The results are the mean AUC score and standard deviation of 20 runs on each dataset. All the datasets were divided into training and test data with 80% and 20% of the data. The parameters $\beta \in 10^{[-5:5]}$ and $\beta_1 \in 10^{[-5:5]}$ for VRSPAM $- L^2$ and VRSPAM $- NET$ are chosen by 5 fold cross-validation on the training set.