Operations Research and Fuzziology 运筹与模糊学, 2023, 13(5), 4318-4326
Published Online October 2023 in Hans. https://www.hanspub.org/journal/orf
https://doi.org/10.12677/orf.2023.135430

Hans汉斯

带有近似最优步长的随机递归梯度算法

陈炫睿

贵州大学数学与统计学院，贵州 贵阳

收稿日期：2023年7月30日；录用日期：2023年9月19日；发布日期：2023年9月27日

摘　要

在机器学习中，我们经常考虑一个目标函数是凸函数和的最小化问题。随机递归梯度算法(SARAH)是求解上面问题的一个常用方法。它允许一个简单的递归框架来更新随机梯度估计。基于SARAH方法，本文提出利用近似最优步长(AOS)去自适应地计算SARAH的步长，并将其命名为SARAH-AOS算法。针对提出的算法，我们进行了数值试验，结果表明SARAH-AOS算法对初始步长的选择并不像SARAH那样敏感。我们的算法对SARAH算法有着显著性能的改进。

关键词

机器学习，随机递归梯度算法，近似最优步长，自适应计算

Stochastic Recursive Gradient Algorithm with Approximately Optimal Stepsize

Xuanrui Chen

School of Mathematics and Statistics, Guizhou University, Guiyang Guizhou

Received: Jul. 30th, 2023; accepted: Sep. 19th, 2023; published: Sep. 27th, 2023

Abstract

In machine learning, we often consider the problem of minimizing an objective function that is a sum of convex functions. The stochastic recursive gradient algorithm (SARAH) is a common method to solve the above problems. It allows a simple recursive framework to update stochastic gradient estimates. Based on the SARAH method, this paper, we propose to use the approximately optimal stepsize (AOS) to automatically compute stepsizes for SARAH, which leads to SARAH-AOS algorithm. For the proposed algorithm, we conduct numerical experiments, and the results show that the SARAH-AOS algorithm is not as sensitive to the selection of initial step size as SARAH. Our algorithm has a significant performance improvement over the SARAH algorithm.

文章引用: 陈炫睿. 带有近似最优步长的随机递归梯度算法[J]. 运筹与模糊学, 2023, 13(5): 4318-4326.
DOI: 10.12677/orf.2023.135430

Keywords

Machine Learning, Stochastic Recursive Gradient Algorithm, Approximately Optimal Stepsize, Adaptive Computing

1. 引言

在大规模机器学习的背景下，以下类型的优化问题被广泛考虑：

$$\min P(\omega)=\frac{1}{n}\sum_{i=1}^{n}f_i(\omega), \tag{1}$$

其中每个 f_i, $i\in\{1,2,\cdots,n\}$ 是一个具有Lipschitz连续梯度的凸函数。这类问题在监督学习应用中经常出现。$\{(x_i,y_i)\}_{i=1}^{n}$ 为一个给定的训练集，且 $x_i\in R^d, y_i\in R^d$，例如，当最小二乘回归模型写成(1)，其中有 $f_i(\omega)=(\omega^T a_i-b_i)^2+\frac{\lambda}{2}\|\omega\|^2$，其中 $\|\cdot\|$ 表示 l_2 范数。

近年来，针对问题(1)开发了许多先进的优化方法。例如梯度下降方法(GD)，特别地，GD 更新迭代如下：

$$\omega_{t+1}=\omega_t-\eta_t\nabla P(\omega_t)=\omega_t-\frac{\eta_t}{n}\sum_{i=1}^{n}\nabla f_i(\omega_t),\ t=0,1,2,\cdots \tag{2}$$

其中 η_t 代表第 t 步迭代的学习率。GD 方法虽然能达到线性收敛速度，但是当训练样本非常大时，GD 方法的计算成本非常大，所以，利用 GD 方法求解大规模问题是不切实际的。

随机梯度下降法(SGD) [1]，作为一种替代方法，已成为求解(1)的首选方法。在每一步，SGD 均匀随机选取一个索引，$i\in\{1,2,\cdots,n\}$，并将迭代更新为：

$$\omega_{t+1}=\omega_t-\eta_t\nabla f_i(\omega_t),\ t=0,1,2,\cdots \tag{3}$$

这大大降低了计算成本。SGD 的收敛速度比 GD 慢，特别是在强凸情况下，它是次线性的。然而，由于每次迭代的巨大节省和低精度解决方案足够的事实，这种权衡是有利的。

近年来，人们开发了大量提高 SGD 性能的方法。最流行的一种方法是梯度聚合算法[2]，如 SAG [3] [4]和 SAGA [5]。他们将随机梯度计算为在之前迭代中评估的随机梯度的平均值。然后它们以牺牲内存为代价来存储之前的随机梯度。随机方差衰减梯度法(SVRG) [6]有两个循环，外层循环计算一个完整的梯度，内层循环计算方差较小的随机梯度。

AdaGrad [7]和 Adam [8]根据历史梯度的平方和自适应地选择每个分量的步长。SVRG-BB 算法[9]将 BB 步长与 SVRG 算法相结合，使 SVRG 的步长被自适应计算。随机梯度递归算法(SARAH) [10]采用一种简单的递归框架来更新随机梯度估计。Liu 等人[11]将 SARAH 与 BB 算法和重要性抽样策略相结合，提出了 SARAH-I-BB 法。刘泽显等人[12] [13] [14]提出了近似最优步长梯度法用来求解严格凸二次极小化问题、大规模的无约束问题。

随着随机梯度下降算法不同版本的提出，这些改进的算法从传统随机梯度下降算法上引入了很多新

的思想。按照搜索方向和步长选取的方式不同，将随机梯度下降算法的改进策略大致分为动量、方差缩减、增量梯度和自适应学习率等四种类型。与 SAG/SAGA 相比，SARAH 不需要存储过去的梯度。所以本文基于 SARAH 算法来进行改进。

受近似最优步长和 SARAH 算法成功解决问题(1)的启发。本文将近似最优步长及其思想应用于随机方差缩减梯度算法中去，将近似最优步长与 SARAH 结合，我们称之为 SARAH-AOS 方法。

2. 近似最优步长与随机递归梯度算法

在本节中，我们回顾近似最优步长与随机梯度递归算法。本节后面的叙述中 $g(\omega)=\nabla f(\omega)$, $g_k=g(\omega_k)$。

2.1. 近似最优步长

BB 步长 $\eta_k^{BB1}=\|s_{k-1}\|^2/s_{k-1}^T y_{k-1}$ 可以通过对下面的优化问题求解获得：

$$\min_{\eta>0}\left\|\frac{s_{k-1}}{\eta}I-y_{k-1}\right\|_2^2.$$

其中 $s_{k-1}=\omega_k-\omega_{k-1}, y_{k-1}=\nabla f(\omega_k)-\nabla f(\omega_{k-1})$。

如果考虑线搜索函数 $f(\omega_k-\eta g_k)$ 的二次近似模型：

$$\phi_k(\eta)=f(\omega_k)-\eta g_k^T g_k+\frac{1}{2}\eta^2 g_k^T B_k g_k, \tag{4}$$

其中 B_k 为黑森矩阵的近似。当 $B_k=\frac{s_{k-1}^T y_{k-1}}{\|s_{k-1}\|^2}I$ 时，函数 $f(\omega_k-\eta\omega_k)$ 的近似模型为 $\phi_k(\eta)$，对 $\phi_k(\eta)$ 极小化，最后能得到 η_k^{BB1}，即：

$$\eta_k^{BB1}=\arg\min_{\eta>0}\phi(\eta).$$

因此，BB 步长 η_k^{BB1} 是与(4)中的近似模型 $\phi_k(\eta)$ 相关的近似最优步长。因为在 BB 方法中，标量近似矩阵 $\frac{s_{k-1}^T y_{k-1}}{\|s_{k-1}\|^2}I$ 要尽可能的去满足割线方程。所以(4)中二次近似模型是一个有效的近似。经过上面的分析，下面给出近似最优步长的定义。

定义 2.1：令 $\phi(\eta)$ 是函数 $f(\omega_k-\eta g_k)$ 的近似模型。一个正数 ω^* 被称为关于 $\phi(\eta)$ 的近似最优步长，如果 ω^* 满足：

$$\omega^*=\arg\min_{\eta>0}\phi(\eta).$$

近似最优步长与柯西步长 $\eta_k^{SD}=\arg\min_{\eta>0}f(\omega_k-\eta g_k)$ 不同，这将导致昂贵的计算成本。近似最优步长通常易于计算，可应用于无约束优化。

2.2. 随机递归梯度算法(SARAH)

该算法的关键步骤是随机梯度估计的递归更新(SARAH 更新)。

$$v_t=\nabla f_{i_t}(\omega_t)-\nabla f_{i_t}(\omega_{t-1})+v_{t-1}, \tag{5}$$

然后是迭代更新：

$$\omega_{t+1}=\omega_t-\eta v_t. \tag{6}$$

为了比较，SVRG 更新可以用类似的方式编写：

$$v_t=\nabla f_{i_t}\left(\omega_t\right)-\nabla f_{i_t}\left(\omega_0\right)+v_0. \tag{7}$$

可以观察到，在 SVRG 中， v_t 是梯度的无偏估计量，而对于 SARAH 则不是。因为：

$$\mathbb{E}\left[v_t\middle|\mathcal{F}_t\right]=\nabla P\left(\omega_t\right)-\nabla P\left(\omega_{t-1}\right)+v_{t-1}\neq\nabla P\left(\omega_t\right). \tag{8}$$

因此，SARAH 不同于 SGD 和 SVRG 类型的方法，但以下总期望成立，$\mathbb{E}\left[v_t\right]=\mathbb{E}\left[\nabla P\left(\omega_t\right)\right]$。SARAH 算法与 SVRG 算法类似，因为这两个算法都包含两个循环。在每一个外循环中两者都需要计算一个完整的梯度，不同的地方就在于在内循环，其中 SARAH 算法通过在(5)中与之前的 v_{t-1} 之间加减分量梯度递归地更新随机步长方向 v_t。

3. 带有近似最优步长的随机递归梯度算法

3.1. B_k 的选择以及步长的确定

为了解决问题(1)，我们考虑 $f\left(\omega_k-\eta g_k\right)$ 的二次逼近模型：

$$\phi_k\left(\eta\right)=f\left(\omega_k\right)-\eta g_k^Tg_k+\frac{1}{2}\eta^2g_k^TB_kg_k.$$

显然，通过最小化上述二次模型，梯度法的近似最优步长为：

$$\eta_k^{AOS}=\frac{\left\|g_k\right\|^2}{g_k^TB_kg_k}. \tag{9}$$

由上式可以得出 B_k 选择的好坏会直接影响到二次逼近模型的近似效果，所以如何构造出合适的 B_k 非常重要。

本文利用 BFGS 公式对标量矩阵 $\lambda_kI=\frac{s_{k-1}^Ty_{k-1}}{\left\|s_{k-1}\right\|^2}I$ 进行更新，从而得到 B_k：

$$B_k=\lambda_kI-\frac{\lambda_kIs_{k-1}s_{k-1}^T\lambda_kI}{s_{k-1}^T\lambda_kIs_{k-1}}+\frac{y_{k-1}y_{k-1}^T}{s_{k-1}^Ty_{k-1}}, \tag{10}$$

将(10)代入(9)可以得到：

$$\tilde{\eta}_k^{AOS}=\frac{\left\|g_k\right\|^2}{\lambda_k\left(\left\|g_k\right\|^2-\frac{\left(g_k^Ts_{k-1}\right)}{\left\|s_{k-1}\right\|^2}\right)+\frac{\left(g_k^Ts_{k-1}\right)^2}{s_{k-1}^Ty_{k-1}}}. \tag{11}$$

通过数值实验观察，我们可以发现 $\tilde{\eta}_k^{AOS}$ 位于区间 $\left[\eta_k^{BB2},\eta_k^{BB1}\right]$ 中。因此我们采用如下截断形式：

$$\eta_k^{AOS}=\min\left\{\eta_k^{BB1},\max\left\{\eta_k^{BB2},\tilde{\eta}_k^{AOS}\right\}\right\}. \tag{12}$$

3.2. SARAH-AOS 算法

当用 SARAH 求解问题(1)时，采用固定的步长 η，并且 η 需要用户指定。并且 SARAH 算法对于步长 η 的选取非常敏感。我们提出了 SARAH-AOS 方法，该方法使用近似最优步长来计算步长 η_k。我们的 SARAH-AOS 算法在算法 1 中描述。SARAH 和 SARAH-AOS 之间的区别是，后者使用近似最优步长来

计算步长 η_k，而不是像 SARAH 那样使用前面的 η。

算法 1 带有近似最优步长的随机递归梯度下降算法(SARAH-AOS)

初始参数 更新频率 m，初始步长 η_0，初始点 $\tilde{\omega}_0$

for $k=0,1,\cdots$ **do**

$\omega_0=\tilde{\omega}_{k-1}$

$v_0^k=\frac{1}{n}\sum_{i=1}^{n}\nabla f_i(\omega_0)$

if $k>1$ **then**

$\eta_k=\frac{1}{m}\eta_k^{AOS}$,

其中 η_k^{AOS} 由(12)给出。

end if

$\omega_1=\omega_0-\eta_k v_0^k$

for $t=1,\cdots,m-1$ **do**

随机选取 $i_t\in\{1,\cdots,n\}$.

$v_t=\left(\nabla f_{i_t}(\omega_k)-\nabla f_{i_t}(\omega_{k-1})\right)/\left(nq_{i_t}\right)+v_{t-1}$

$\omega_{t+1}=\omega_t-\eta_k v_t$

end for

$\tilde{\omega}_k=\omega_m$

end for

4. 数值实验

为了支持理论分析和见解，我们提出了我们的实验验证。我们在 w8a、ijcnn1、a9a 三个数据集上对我们的算法进行数值实验，数据集的具体信息如表 1 所示。并且用 SARAH-AOS 算法去解决下面的两个问题。带有 l_2 范数正则化的逻辑回归：

$$\min_x F(x)=\frac{1}{n}\sum_{i=1}^{n}\log\left[1+e^{(-b_i a_i^T x)}\right]+\frac{\lambda}{2}\|x\|_2^2, \tag{13}$$

和带有 l_2 范数正则化的平方铰链损失支持向量机：

$$\min_x F(x)=\frac{1}{n}\sum_{i=1}^{n}\left(\left[1-b_i a_i^T x\right]_+\right)^2+\frac{\lambda}{2}\|x\|_2^2. \tag{14}$$

Table 1. Data and model information of the experiments
表 1. 实验数据和模型信息

数据集	n	d	模型	λ
w8a	49749	300	LR	10^{-4}
a9a	37561	123	LR	10^{-3}
ijcnn1	49990	22	SVM	10^{-4}

新算法与 SARAH 的效果对比如图 1~6 所示。在图 1~6 中，横坐标代表迭代的次数。在图 1~3 中，纵坐标代表算法的次优性 $F(\tilde{x}_k)-F(x^*)$。在图 4~6 中，纵坐标代表步长 η_k 的变化。

在下面所有图中，虚线对应采用固定步长的 SARAH 算法。红色虚线表示最具调优步长的 SARAH

算法。蓝色虚线使用比最优步长稍大的步长，黑色虚线使用较小的固定步长。*实线对应的是具有不同初始步长的 SARAH-AOS 算法。例如，图 1 和图 4 中的实线分别对应于初始步长 $\eta_0 = 0.09$、0.15、0.24 的 SARAH-AOS 算法。

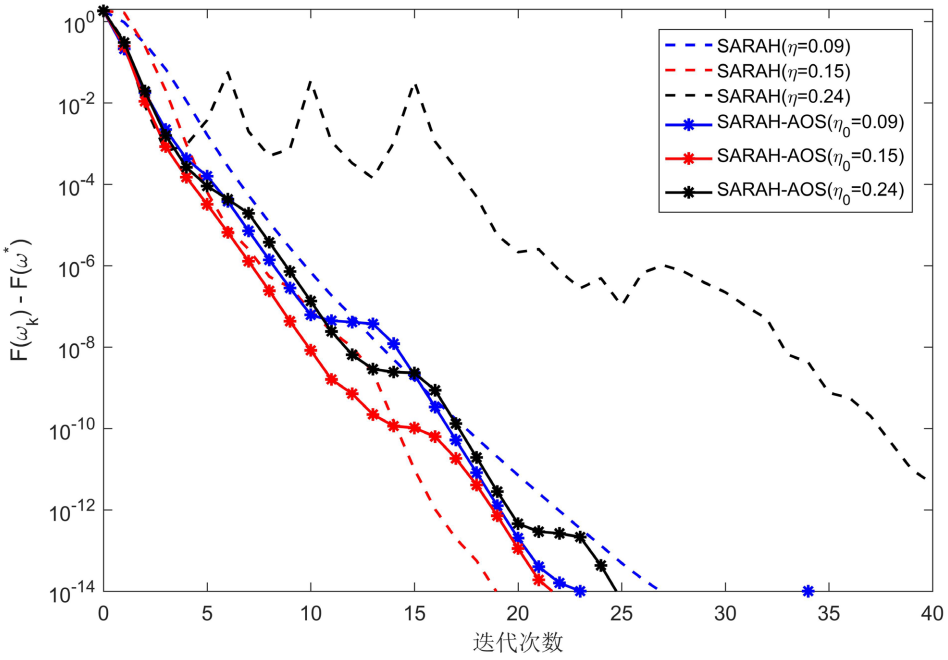

Figure 1. Sub-optimality on w8a
图 1. w8a 上的次优性比较

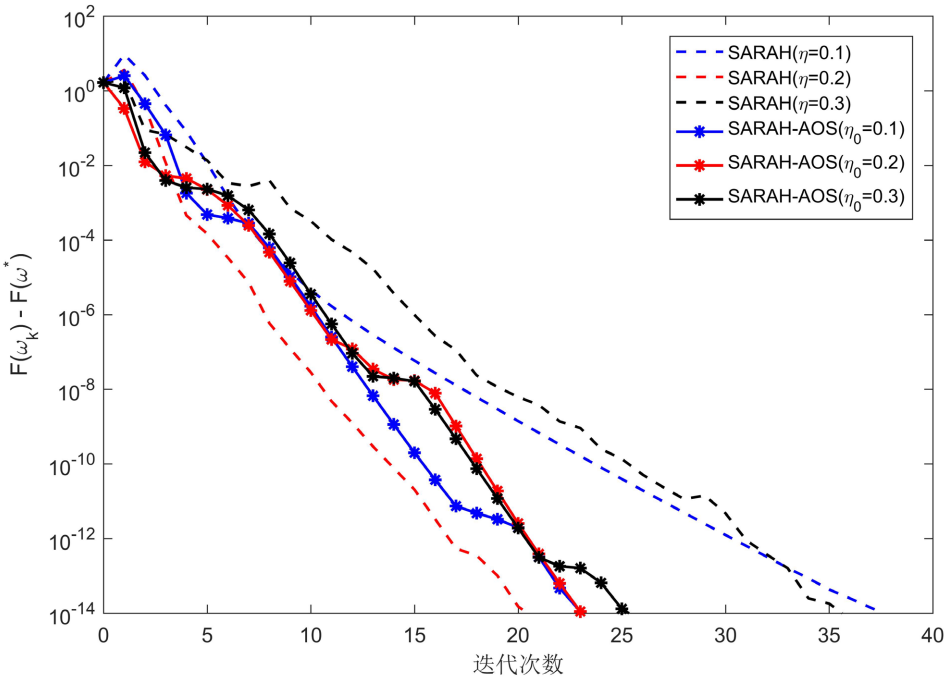

Figure 2. Sub-optimality on a9a
图 2. a9a 上的次优性比较

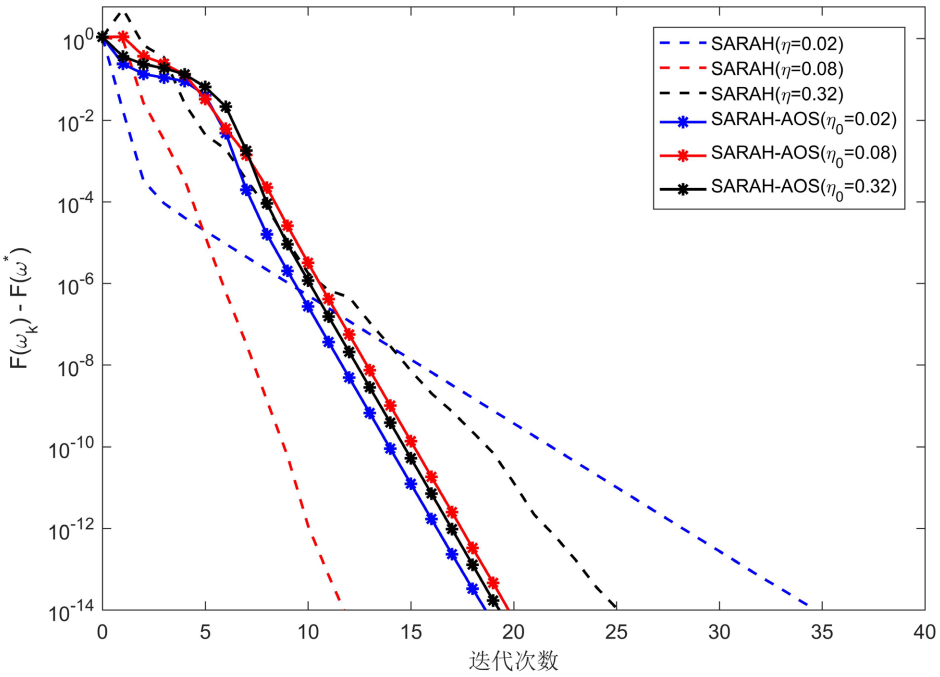

Figure 3. Sub-optimality on ijcnn1
图 3. ijcnn1 上的次优性比较

从图 1~3 我们可以看到 SARAH-AOS 算法始终可以达到与 SARAH 相同的次优性水平，且是最佳调优的步长。尽管 SARAH-AOS 比具有最佳调优步长的 SARAH 需要更多的迭代次数，但它明显优于具有其他两种步长选择的 SARAH。此外，SARAH-AOS 算法性能受初始步长的选择影响要明显小于 SARAH 算法。

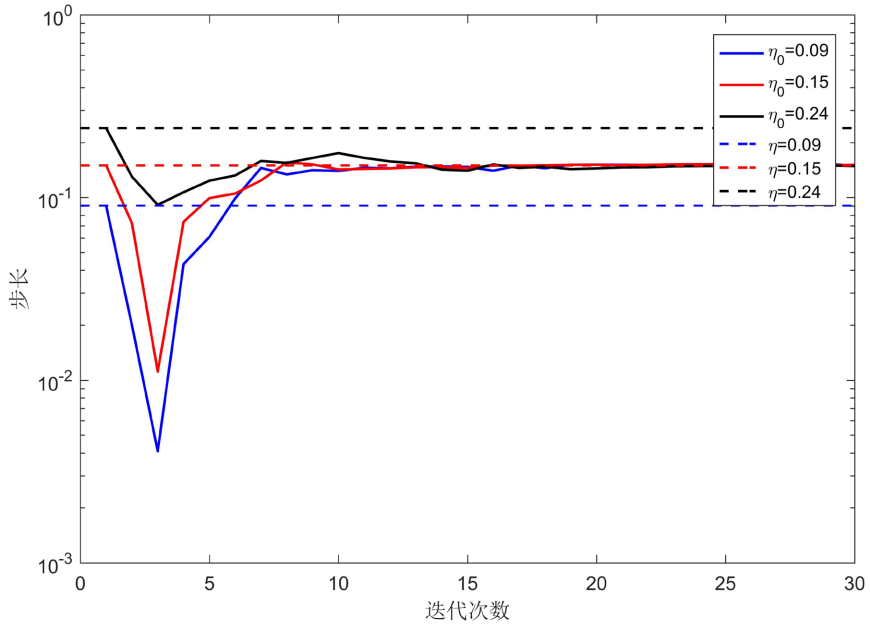

Figure 4. Stepsizes on w8a
图 4. w8a 上的步长变化

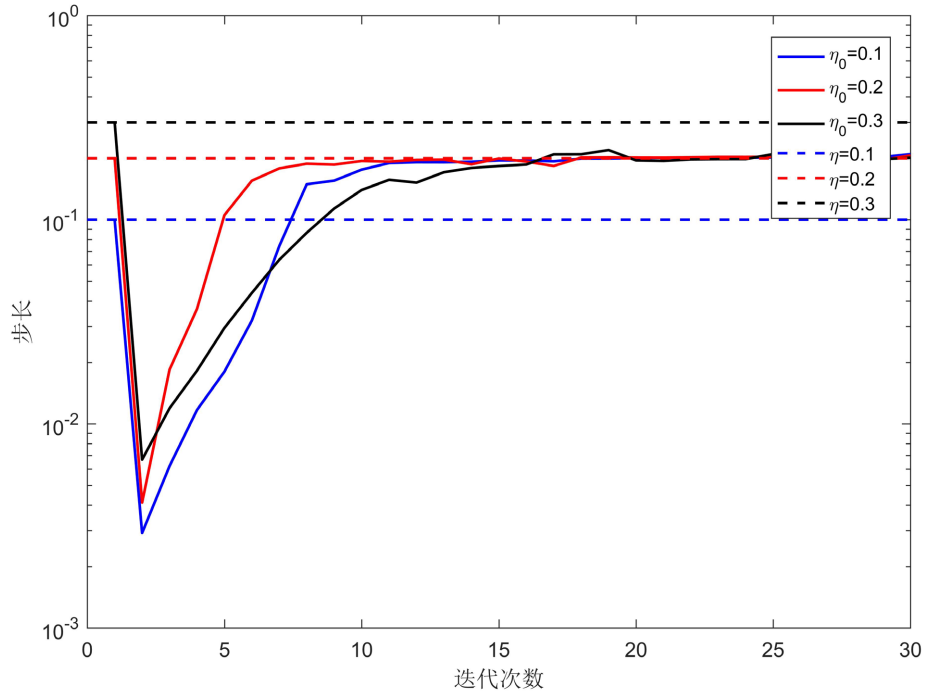

Figure 5. Stepsizes on a9a
图 5. a9a 上的步长变化

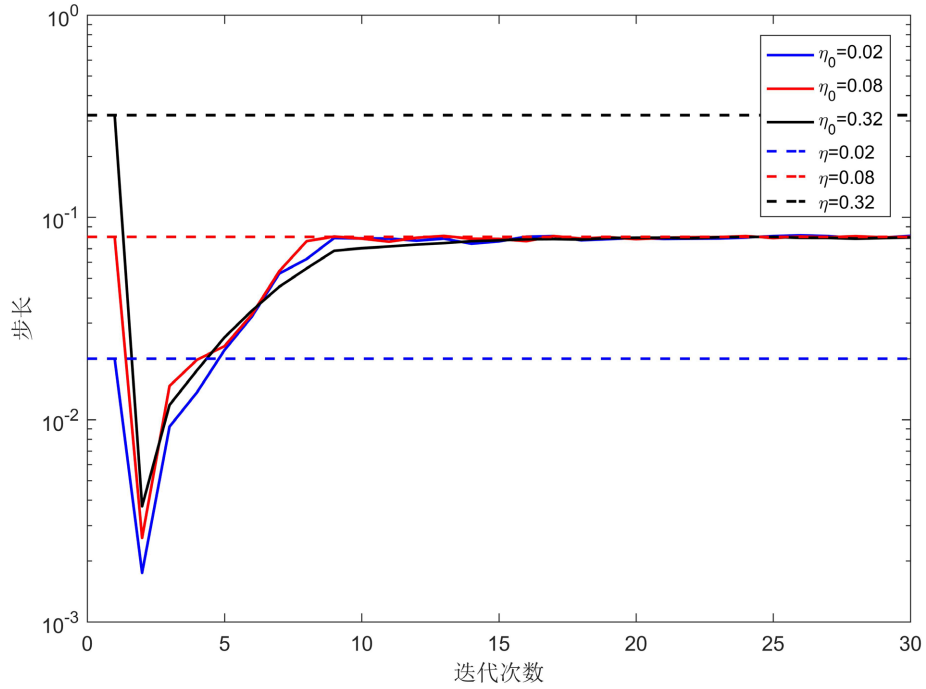

Figure 6. Stepsizes on ijcnn1
图 6. ijcnn1 上的步长变化

从图 4~6 我们观察到，在选取不同初始步长的情况下，SARAH-AOS 算法中计算的步长最终将接近 SARAH 的最调优步长。因此，SARAH-AOS 在运行算法时自动生成最佳步长，在实践中具有很大的潜力。

5. 结论

本文提出了一种新的算法 SARAH-AOS 算法。在 SARAH 的外循环中我们采用近似最优步长自动地计算算法的步长，与采用固定步长的传统 SARAH 算法相比，该算法有较好的数值性能。在数值实验中，我们发现新算法能够达到与带有最调优步长的随机递归梯度下降算法相同程度的次优性。并且发现 SARAH-AOS 算法对于初始步长 η_0 的选择并不敏感。并且 SARAH-AOS 算法中计算的步长最终将接近 SARAH 的最调优步长。因此新算法能自动生成最佳步长，所以它具有很大的实际应用潜力。数值计算结果也表明，它具有良好的数值性能。但是 SARAH-AOS 算法的内循环大小仍然需要用户去指定，这是该算法局限的地方。后续我们的工作将围绕这一不足点做进一步的改进。

基金项目

国家自然科学基金项目(12261019)。

参考文献

[1] Robbins, H. and Monro, S. (1951) A Stochastic Approximation Method. *The Annals of Mathematical Statistics*, **22**, 400-407. https://doi.org/10.1214/aoms/1177729586

[2] Bottou, L., Curtis, F.E. and Nocedal, J. (2018) Optimization Methods for Large-Scale Machine Learning. *SIAM Review*, **60**, 223-311. https://doi.org/10.1137/16M1080173

[3] Roux, N.L., Schmidt, M. and Bach, F.R. (2013) A Stochastic Gradient Method with an Exponential Convergence Rate for Finite Training Sets. *Neural Information Processing Systems*, Lake Tahoe, December 2013, 2663-2671.

[4] Schmidt, M.W., Roux, N.L. and Bach, F. (2017) Minimizing Finite Sums with the Stochastic Average Gradient. *Mathematical Programming*, **162**, 83-112. https://doi.org/10.1007/s10107-016-1030-6

[5] Defazio, A., Bach, F. and Lacoste-Julien, S. (2014) SAGA: A Fast Incremental Gradient Method with Support for Non-Strongly Convex Composite Objectives. *Neural Information Processing Systems*, Montreal, December 2014, 1646-1654.

[6] Johnson, R. and Zhang, T. (2013) Accelerating Stochastic Gradient Descent Using Predictive Variance Reduction. *Neural Information Processing Systems*, Lake Tahoe, December 2013, 315-323.

[7] Duchi, J.C., Hazan, E. and Singer, Y.J. (2011) Adaptive Subgradient Methods for Online Learning and Stochastic Optimization. *Journal of Machine Learning Research*, **12**, 2121-2159.

[8] Kingma, D.P. and Ba, J. (2015) Adam: A Method for Stochastic Optimization. *International Conference on Learning Representations*, San Diego, May 2015, 1-13.

[9] Tan, C., Ma, S., Dai, Y.H. and Qian, Y. (2016) Barzilai-Borwein Step Size for Stochastic Gradient Descent. *Neural Information Processing Systems*, Barcelona, December 2016, 685-693.

[10] Nguyen, L.M., Liu, J., Scheinberg, K. and Takac, M. (2017) SARAH: A Novel Method for Machine Learning Problems Using Stochastic Recursive Gradient. *Neural Information Processing Systems*, Long Beach, December 2017, 2613-2621.

[11] Liu, Y., Wang, X. and Guo, T. (2020) A Linearly Convergent Stochastic Recursive Gradient Method for Convex Optimization. *Optimization Letters*, **14**, 2265-2283. https://doi.org/10.1007/s11590-020-01550-x

[12] Liu, Z., Liu, H. and Dong, X. (2018) An Efficient Gradient Method with Approximate Optimal Stepsize for the Strictly Convex Quadratic Minimization Problem. *Optimization*, **67**, 427-440. https://doi.org/10.1080/02331934.2017.1399392

[13] Liu, Z. and Liu, H. (2018) Several Efficient Gradient Methods with Approximate Optimal Stepsizes for Large Scale Unconstrained Optimization. *Journal of Computational and Applied Mathematics*, **328**, 400-413. https://doi.org/10.1016/j.cam.2017.07.035

[14] Liu, Z. and Hongwei, L. (2018) An Efficient Gradient Method with Approximate Optimal Stepsize for Large-Scale Unconstrained Optimization. *Numerical Algorithms*, **78**, 21-39. https://doi.org/10.1007/s11075-017-0365-2