软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2015,26(11):2752−2761 [doi: 10.13328/j.cnki.jos.004890]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

一种减小方差求解非光滑问题的随机优化算法*

朱小辉, 陶 卿, 邵言剑, 储德军

(中国人民解放军陆军军官学院 十一系,安徽 合肥 230031)

通讯作者: 朱小辉, E-mail: xiaohuiaigw@163.com

摘 要: 随机优化算法是求解大规模机器学习问题的高效方法之一.随机学习算法使用随机抽取的单个样本梯度代替全梯度,有效节省了计算量,但却会导致较大的方差.近期的研究结果表明:在光滑损失优化问题中使用减小方差策略,能够有效提高随机梯度算法的收敛速率.考虑求解非光滑损失问题随机优化算法 COMID(composite objective mirror descent)的方差减小问题.首先证明了 COMID 具有方差形式的 $O\left(1/\sqrt{T}+\sigma^2/\sqrt{T}\right)$ 收敛速率,其中,T 是迭代步数,σ^2 是方差.该收敛速率保证了减小方差的有效性,进而在 COMID 中引入减小方差的策略,得到一种随机优化算法 α-MDVR(mirror descent with variance reduction).不同于 Prox-SVRG(proximal stochastic variance reduced gradient),α-MDVR 收敛速率不依赖于样本数目,每次迭代只使用部分样本来修正梯度.对比实验验证了 α-MDVR 既减小了方差,又节省了计算时间.

关键词: 机器学习;随机算法;非光滑;方差;composite objective mirror descent(COMID)

中图法分类号: TP181

中文引用格式: 朱小辉,陶卿,邵言剑,储德军.一种减小方差求解非光滑问题的随机优化算法.软件学报,2015,26(11):2752−2761. http://www.jos.org.cn/1000-9825/4890.htm

英文引用格式: Zhu XH, Tao Q, Shao YJ, Chu DJ. Stochastic optimization algorithm with variance reduction for solving non-smooth problems. Ruan Jian Xue Bao/Journal of Software, 2015,26(11):2752−2761 (in Chinese). http://www.jos.org.cn/1000-9825/4890.htm

Stochastic Optimization Algorithm with Variance Reduction for Solving Non-Smooth Problems

ZHU Xiao-Hui, TAO Qing, SHAO Yan-Jian, CHU De-Jun

(11st Department, Army Officer Academy of PLA, Hefei 230031, China)

Abstract: Stochastic optimization is one of the efficient methods for solving large-scale machine learning problems. In stochastic learning, the full gradient is replaced by the gradient of loss function in terms of a randomly selected single sample to avoid high computational cost. However, large variance is usually caused. Recent studies have shown that the convergence rate of stochastic gradient methods can be effectively improved by reducing the variance. In this paper, the variance reduction issue in COMID (composite objective mirror descent) is addressed when solving non-smooth optimization problems. First a proof is provided to show that the COMID has a convergence rate $O\left(1/\sqrt{T}+\sigma^2/\sqrt{T}\right)$ in the terms of variance, where T is the iteration number and σ^2 is the variance. This convergence rate ensures the effectiveness of reducing the variance. Then, a stochastic optimization algorithm α-MDVR (mirror descent with variance reduction) is obtained by incorporating the strategy of reducing variance into COMID. Unlike Prox-SVRG (proximal stochastic variance reduced gradient), α-MDVR does not depend on the number of samples and only uses a small portion of samples to modify the gradient. The comparative experiments demonstrate that α-MDVR not only reduces the variance but also decreases the computational time.

Key words: machine learning; stochastic algorithm; non-smooth; variance; composite objective mirror descent (COMID)

* 基金项目: 国家自然科学基金(61273296); 安徽省自然科学基金(1308085QF121)
收稿时间: 2015-02-26; 修改时间: 2015-05-11, 2015-07-14; 定稿时间: 2015-08-26

机器学习正面临着数据规模日益扩大的严峻挑战.传统的机器学习批处理梯度优化方法每次迭代都要遍历所有样本[1],已经不能满足快速求解机器学习优化问题的需求.由于机器学习问题通常假设样本是独立同分布的,从而随机抽取单个样本的目标函数的梯度是整个目标函数梯度的无偏估计[2],进而可用每次迭代仅处理单个或部分样本的随机优化方法来代替批处理方法[2,3].虽然随机优化算法每一步迭代能够节省计算开销,但往往会导致较大的方差,不可避免地会影响算法的收敛速率[4].

为了减小方差对算法收敛速率的影响,Johnson 等人于 2013 年提出一种在优化算法中采用减小方差的策略,称为 SVRG(stochastic variance reduced gradient)[4].SVRG 的主要思路是:在原有单个样本的梯度计算中引入修正量,该修正量由所有样本梯度的平均值组成,与仅使用单个样本梯度的标准随机优化算法相比,修正后的梯度可以明显减小方差.对于求解光滑强凸优化问题,SVRG 算法能够达到最优的收敛速率,与当前主流优化方法 SDCA(stochastic dual coordinate ascent)[5]和 SAG(stochastic average gradient)[6]的收敛速率相同.与 SDCA 和 SAG 不同的是,SVRG 不需要存储目标函数的梯度,从而减小了内存开销.另一方面,SVRG 收敛速率的证明过程也更加简单,并可以应用于求解深度神经网络导致的非凸优化问题[4].

正是由于 SVRG 具有上述优点,Xiao 等人于 2014 年将 SVRG 由求解简单的黑箱优化问题推广到具有结构信息的正则化损失函数优化问题[7],称为 Prox-SVRG(proximal SVRG)[8].与 SVRG 不同的是,Prox-SVRG 保持正则化项不变,仅对损失项的梯度进行优化.这种处理方式与当前流行的结构优化算法 RDA(regularized dual average)[7]和 COMID(composite objective mirror descent)[9]完全相同.与 SVRG 一样,Prox-SVRG 也能达到最优的收敛速率.从随机优化算法的观点来看,SVRG 和 Prox-SVRG 所使用的梯度仍然是整个目标函数梯度的无偏估计,并没有改变随机算法的本质.但是,也正是由于 SVRG 和 Prox-SVRG 需要用全梯度来修正单个样本梯度,从而迭代过程中不得不多次遍历所有样本,虽然减小方差的效果明显,但却和批处理算法一样,耗费了大量的计算时间.值得指出的是,SVRG 和 Prox-SVRG 最优收敛速率的获得都依赖于样本数目,从而只能处理样本数目确定的优化问题,并且它们仅限于求解光滑的损失函数问题[8].

对于求解非光滑损失的结构优化问题,COMID 算法是一种较好的选择[9].本文主要目的是在 COMID 中引入方差减小策略,为此,我们首先证明了 COMID 算法具有方差形式的收敛速率$O\left(1/\sqrt{T}+\sigma^2/\sqrt{T}\right)$,其中,$T$是迭代步数,$\sigma^2$ 是方差.该收敛速率能够从理论上保证减小方差策略的有效性,进而本文得到一种随机优化算法 α-MDVR.为了避免遍历所有样本导致的计算时间多的问题,α-MDVR 每次迭代只使用部分样本来修正梯度,既减小了方差,又节省了计算时间,其中,修正梯度需要样本的数目由α决定,也具有较好的柔韧性.另外,对于求解非光滑一般凸优化问题,α-MDVR 能够得到最优收敛速率,该收敛速率不依赖于样本数目.对比实验验证了 α-MDVR 确实能够在适度减小方差的同时节省 CPU 时间,并获得了比 COMID 更快的实际收敛速率.

1 光滑损失随机优化算法的减小方差策略

为简单起见,本文仅讨论二分类问题.设样本集合:

$$S=\{(x_1,y_1),\ldots,(x_n,y_n)\}\in R^n\times\{+1,-1\},$$

其中,(x_i,y_i)是独立同分布的,正则化损失函数的优化问题可以表述为

$$\min_{w\in\Omega}\Phi(\boldsymbol{w}),\ \Phi(\boldsymbol{w})=r(\boldsymbol{w})+\frac{1}{n}\sum_{i=1}^{n}f_i(\boldsymbol{w}) \tag{1}$$

其中,$\boldsymbol{w}\in R^n$,$r(\boldsymbol{w})$是正则化项,损失函数$f_i(\boldsymbol{w})$是由样本$\boldsymbol{x}_i$造成的损失.

本节主要介绍 SVRG 和 Prox-SVRG 算法,两种算法都是求解光滑强凸优化问题,因此,本节假设损失函数是光滑的,目标函数$\Phi(\boldsymbol{w})$具有强凸性质.

很多研究者对优化问题(1)的求解进行过研究[1,2,10−13],其中,梯度下降法是最简单的一阶优化方法,即

$$\boldsymbol{w}_{t+1}=\boldsymbol{w}_t-\eta_t g(\boldsymbol{w}_t) \tag{2}$$

其中,$g(\boldsymbol{w}_t)$是关于所有样本目标函数$\Phi(\boldsymbol{w})$在$\boldsymbol{w}_t$处的全梯度,η_t是学习步长.

随机梯度下降(stochastic gradient descent,简称 SGD)[13]是梯度下降法的随机形式.与批处理方法相比,SGD

每次迭代中只计算单个样本对应目标函数的梯度,用这种无偏估计避免计算所有样本对应目标函数的梯度.SGD 的主要迭代步骤如下:

$$\boldsymbol{w}_{t+1}=\boldsymbol{w}_t-\eta_t\hat{g}_t(\boldsymbol{w}_t,\xi_t) \tag{3}$$

式中,$\hat{g}_t(\boldsymbol{w}_t,\xi_t)$是关于单个样本对应的目标函数在$\boldsymbol{w}_t$处的次梯度,其中,$\{\xi_t\}_{t>1}$表示一系列随机变量,对应的是随机抽取的单个样本.

由于样本是独立同分布的,从而$\hat{g}_t(\boldsymbol{w}_t,\xi_t)$是整个目标函数$\Phi(\boldsymbol{w}_t)$次梯度 $g(\boldsymbol{w}_t)$的无偏估计,但却存在着方差 $E[\|\hat{g}_t(\boldsymbol{w}_t,\xi_t)-g(\boldsymbol{w}_t)\|^2]$,这无疑会影响算法的收敛速率.

Johnson 等人提出一种减小方差的方法 SVRG[4],其主要迭代步骤如下:

$$\boldsymbol{w}_t=\boldsymbol{w}_{t-1}-\eta(\nabla\Phi_{i_t}(\boldsymbol{w}_{t-1})-\nabla\Phi_{i_t}(\tilde{\boldsymbol{w}})+\tilde{\mu}) \tag{4}$$

其中,$\tilde{\boldsymbol{w}}$是 SVRG 进行 m 次迭代后取的平均值,在文献[4]中,m 取 $2n$(n 为训练样本数);$\nabla\Phi_{i_t}(\boldsymbol{w}_{t-1})$是关于单个样本目标函数在 $\boldsymbol{w}_{t-1}$ 处的梯度;$\nabla\Phi_{i_t}(\tilde{\boldsymbol{w}})$是关于单个样本目标函数在 $\tilde{\boldsymbol{w}}$ 处的梯度;$\tilde{\mu}$是整个目标函数在 $\tilde{\boldsymbol{w}}$ 处的全梯度,即,$\tilde{\mu}=\frac{1}{n}\sum_{i=1}^{n}\Phi_i(\tilde{\boldsymbol{w}})$.

与 SGD 不同的是,SVRG 用$\nabla\Phi_{i_t}(\boldsymbol{w}_{t-1})-\nabla\Phi_{i_t}(\tilde{\boldsymbol{w}})+\tilde{\mu}$代替了单个样本的梯度.从形式上看,通过引入修正量,使得方差更小,即,引入修正量后的梯度导致的方差 $E[\|(\nabla\Phi_{i_t}(\boldsymbol{w}_{t-1})-\nabla\Phi_{i_t}(\tilde{\boldsymbol{w}})+\tilde{\mu})-g(\boldsymbol{w}_t)\|^2]$确实比原有单个样本梯度导致的方差 $E[\|\hat{g}_t(\boldsymbol{w}_t,\xi_t)-g(\boldsymbol{w}_t)\|^2]$要小.对于求解光滑损失优化问题,在样本数目确定的情况下,SVRG 能够得到指数阶的最优收敛速率.

SVRG 是将目标函数$\Phi(\boldsymbol{w})$看成一个整体来考虑,忽略了损失函数和正则化项有着各自特殊的含义.Xiao 等人将问题(1)中的正则化项和损失函数分开考虑,保持正则化项不变,仅对损失项的梯度进行优化.将 SVRG 由求解简单的黑箱问题推广到具有结构信息的正则化损失函数优化问题,于 2014 年提出了 Prox-SVRG[8].其主要迭代步骤如下:

$$\boldsymbol{w}_t=prox_{\eta R}(\boldsymbol{w}_{t-1}-\eta v_t),\ v_t=(\nabla f_{i_t}(\boldsymbol{w}_{t-1})-\nabla f_{i_t}(\tilde{\boldsymbol{w}}))/(q_{i_t}n)+\tilde{v} \tag{5}$$

其中,$\nabla f_{i_t}(\boldsymbol{w}_{t-1})$是关于单个样本损失函数在$\boldsymbol{w}_{t-1}$处的梯度;$\nabla f_{i_t}(\tilde{\boldsymbol{w}})$是关于单个样本损失函数在 $\tilde{\boldsymbol{w}}$ 处的梯度;$\tilde{v}$是损失函数在 $\tilde{\boldsymbol{w}}$ 处的全梯度,即,$\tilde{v}=\frac{1}{n}\sum_{i=1}^{n}f_i(\tilde{\boldsymbol{w}})$.

但是,正是由于 SVRG 和 Prox-SVRG 需要用全梯度来修正单个样本梯度,从而不得不遍历所有样本,虽然减小方差的效果明显,但却耗费了大量的计算时间.值得指出的是,SVRG 和 Prox-SVRG 最优收敛速率的界都与样本数目有关,从而只能处理样本数目确定的优化问题,并且它们仅限于求解光滑的损失函数问题.

2 非光滑损失问题的α-MDVR 算法

与问题(1)对应的随机优化问题是:

$$\min_{w\in\Omega}\Phi(\boldsymbol{w})=r(\boldsymbol{w})+E_{\xi}[f(\boldsymbol{w},\boldsymbol{x})] \tag{6}$$

其中,$r(\boldsymbol{w})$是正则化项;$f(\boldsymbol{w},\boldsymbol{x})$是损失函数;$\xi$表示随机变量,对应的是随机抽取的单个样本.

为了方便地进行理论分析和实验验证,本文仅考虑最简单但最典型的非光滑稀疏学习问题“L1 正则化+Hinge 损失”,即

$$r(\boldsymbol{w})=\|\boldsymbol{w}\|_1, f(\boldsymbol{w},\boldsymbol{x})=\max\{0,1-y\boldsymbol{w}^T\boldsymbol{x}\}.$$

与问题(1)不同的是,问题(6)不需要固定样本数,可以描述任意规模的机器学习优化问题.当样本数目充分大时,问题(1)可以看成是问题(6)的一种随机逼近.

对于求解正则化非光滑损失问题(6),如果样本维数足够大,MD(mirror descent)[14]被认为是最优的一阶方法,随机 MD 的主要迭代步骤如下:

$$\boldsymbol{w}_{t+1}=\arg\min_{w\in\Omega}\{\eta_t\langle\boldsymbol{g}_t,\boldsymbol{w}-\boldsymbol{w}_t\rangle+B_{\varphi}(\boldsymbol{w},\boldsymbol{w}_t)\} \tag{7}$$

其中,函数 $B_\varphi(\boldsymbol{w},\boldsymbol{w}_t)$表示$\varphi$函数的 Bregman Divergence[15];$\boldsymbol{g}_t$是关于单个样本目标函数$\Phi(\boldsymbol{w})$在 $\boldsymbol{w}_t$处的次梯度;$\langle\cdot,\cdot\rangle$表示向量内积.

MD 将目标函数$\Phi(\boldsymbol{w})$当成一个整体进行处理,属于黑箱方法.

2010 年,Duchi 等人[9]对经典 MD 算法进行了突破性的改进,将黑箱方法 MD 扩展到结构方法 COMID.该算法在处理优化问题的过程中保持正则化项不动,仅对损失函数进行近似展开,此时的子问题可以解析求解,主要迭代步骤如下:

$$\boldsymbol{w}_{t+1}=\arg\min_{w\in\Omega}\{\eta_t\langle\boldsymbol{g}_t,\boldsymbol{w}\rangle+\eta_t r(\boldsymbol{w})+B_\varphi(\boldsymbol{w},\boldsymbol{w}_t)\} \tag{8}$$

与 MD 不同的是,$\boldsymbol{g}_t$仅为损失函数$f(\boldsymbol{w}_t,\boldsymbol{x})$在 $\boldsymbol{w}_t$处的次梯度.COMID 的主要执行过程如算法 1 所示.

算法 1.

1. Input: initialize $\boldsymbol{w}_1=0$;
2. For t=1 to T
3. Compute $\boldsymbol{g}_t\in\partial f(\boldsymbol{w}_t,\boldsymbol{x})$
4. Compute $\boldsymbol{w}_{t+1}$ via Eq.(8)
5. End for
6. Output: $\boldsymbol{w}_T$ or $\bar{\boldsymbol{w}}_T=(\boldsymbol{w}_1+...+\boldsymbol{w}_T)/T$.

收敛速率是评价随机算法的主要性能指标[16],是数学期望下随机算法输出解对应的目标值向原优化问题最优值收敛的速率,其数学表达式为 $E[\Phi(\boldsymbol{w})-\Phi(\boldsymbol{w}_*)]$,其中,$\boldsymbol{w}$ 是随机算法的输出解,$\boldsymbol{w}_*$是原优化问题的最优解.COMID 在求解非光滑一般凸优化问题时,有如下定理成立:

定理 1[9]. $\boldsymbol{w}_t$是算法(8)第 t 次迭代的输出.设 $r(\boldsymbol{w}_1)=0,B_\varphi(\boldsymbol{w},\boldsymbol{w}_t)=\|\boldsymbol{w}-\boldsymbol{w}_t\|^2$,步长 $\eta_t=1/\sqrt{t}$, 且存在常数 N,G 满足 $E[\|\boldsymbol{w}_1-\boldsymbol{w}_*\|]\leqslant N,\|f_t'(\boldsymbol{w}_t)\|\leqslant G$, 我们有:

$$E\left[\Phi\left(\frac{1}{T}\sum_{t=1}^{T}\boldsymbol{w}_t\right)-\Phi(\boldsymbol{w}_*)\right]\leqslant\frac{\sqrt{T}}{T}(N^2+G^2).$$

上述 COMID 收敛速率的界主要使用损失函数次梯度的上界 G 进行描述.为了更清楚地了解方差对算法收敛速率的影响,我们需要讨论 COMID 关于方差描述的收敛速率,具体见下述定理:

定理 2. 令 $\boldsymbol{w}_t$ 是算法(8)第 t 次迭代的输出.假设 $r(\boldsymbol{w}_1)=0,B_\varphi(\boldsymbol{w},\boldsymbol{w}_t)=\|\boldsymbol{w}-\boldsymbol{w}_t\|^2$,且存在常数 R,M,σ满足 $E[\|\boldsymbol{w}_1-\boldsymbol{w}_*\|]\leqslant R,E[\|\nabla\boldsymbol{F}(\boldsymbol{w})\|]\leqslant M,E_\xi[\|\boldsymbol{g}_t(\boldsymbol{w}_t,\xi)-\nabla\boldsymbol{F}(\boldsymbol{w}_t)\|^2]=\sigma_t^2\leqslant\sigma^2$, 则

$$E\left[\Phi\left(\frac{1}{T}\sum_{t=1}^{T}\boldsymbol{w}_t\right)-\Phi(\boldsymbol{w}_*)\right]\leqslant\frac{(M^2+R^2)}{\sqrt{T}}+\frac{\sigma^2}{\sqrt{T}} \tag{9}$$

具体证明见附录.

对于正则化光滑损失函数的凸优化问题,一些常用的结构优化算法具有方差形式描述的最优速率[7].但对于非光滑的一般凸问题,目前仅证明了黑箱形式的 MD 算法具有方差形式的最优速率[17].定理 2 实际上证明了结构优化算法 COMID 也具有方差形式的最优收敛速率.

为了更加清楚地描述每一步迭代导致的方差对算法收敛速率的影响,从定理 2 的证明过程中还可以得到如下形式的界:

$$E\left[\Phi\left(\frac{1}{T}\sum_{t=1}^{T}\boldsymbol{w}_t\right)-\Phi(\boldsymbol{w}_*)\right]\leqslant\frac{(M^2+R^2)}{\sqrt{T}}+\frac{1}{2T}\sum_{t=1}^{T}\frac{1}{\sqrt{t}}\sigma_t^2.$$

从理论分析的角度来说,如果能够在 COMID 每一步迭代过程中减少σ_t,显然会有助于提高算法的收敛速率.因此,我们在 COMID 中引入思路与 SVRG 和 Prox-SVRG 类似的减小方差策略,得到一种求解随机优化问题(6)的α-MDVR 算法,具体执行流程如下:

算法 2. α-MDVR 算法.

1. **Initialize**: $\tilde{\boldsymbol{w}}_0=0$, $\boldsymbol{w}_0=0$, $m=\alpha n$

2. **Iterate**: for s=1,2,...
3. $\tilde{\boldsymbol{w}} = \tilde{\boldsymbol{w}}_{s-1}$
4. $\tilde{\boldsymbol{v}} = \nabla F(\tilde{\boldsymbol{w}}) = \frac{1}{m}\sum_{t=1}^{m}\nabla f_t(\tilde{\boldsymbol{w}})$
5. $\boldsymbol{w}_0 = \boldsymbol{w}_{s-1}$
6. **Iterate**: for t=1,2,…,m
7. **Initialize**: $\eta_s = 1/\sqrt{(s-1)m+t}$
8. $\boldsymbol{g}_{t-1} = \nabla f_{t-1}(\boldsymbol{w}_{t-1},\xi_{t-1}) - \nabla f_{t-1}(\tilde{\boldsymbol{w}},\xi_{t-1}) + \tilde{\boldsymbol{v}}$
9. Compute $\boldsymbol{w}_t$ via Eq.(8)
10. **end**
11. set $\tilde{\boldsymbol{w}}_s = \frac{1}{m}\sum_{t=1}^{m}\boldsymbol{w}_t$
12. set $\boldsymbol{w}_s = \frac{1}{sm}\sum_{t=0}^{m}\boldsymbol{w}_t$
13. **end**

算法 2 中, $\boldsymbol{g}_t(\boldsymbol{w}_t,\xi_t) = \nabla \boldsymbol{f}_t(\boldsymbol{w}_t,\xi_t) - \nabla \boldsymbol{f}_t(\tilde{\boldsymbol{w}},\xi_t) + \nabla \boldsymbol{F}(\tilde{\boldsymbol{w}}), \nabla \boldsymbol{F}(\tilde{\boldsymbol{w}})$ 仅仅是关于 m 个样本损失函数在中间变量 $\tilde{\boldsymbol{w}}$ 处的梯度,其中,m 是算法在一个阶段内迭代的次数.

与 SVRG 和 Prox-SVRG 减小方差操作不同的是,α-MDVR 在迭代过程中梯度的修正量只取部分样本.值得指出的是,在算法 2 中,选取部分样本修正后的梯度仍然是样本集合上损失函数梯度的无偏估计,因此,算法 2 实际上是一种特殊的随机优化算法,定理 1 仍然适用.但定理 2 更进一步地解释了方差减少策略的理论依据问题.

为了方便实验和说明修正样本数目对方差及实际收敛速率的影响,α实际上取的是修正梯度所用的样本数目占总样本数目的比例,即,α取定后,修正梯度所用的样本数就是αn,其中,n 为所用数据集的样本总数.容易看出,当α=1 时,计算 $\tilde{\boldsymbol{w}}$ 处的梯度需要遍历所有样本,此时,算法减少方差的操作与 SVRG 和 Prox-SVRG 完全一致;当α=1/n 时,算法不进行梯度修正,即为标准的 COMID.综上所述,本文提出的算法从实际运行时间的角度考虑了方差减小问题,α的选取使算法具有更好的柔韧性.

3 实 验

本节对随机优化算法α-MDVR 的性能进行实验验证.实验所涉及的所有算法均在 Sun Fire X4170 M2 服务器(2.4GHz Intel(R) Xeon(R)处理器,12G 内存,Solaris 操作系统)上运行.实验中所用的 4 个标准数据库均可以从 LIBSVM 网站中获得,具体下载地址为 http://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/.表 1 给出了 4 个数据库的详细信息.

我们首先在 4 个标准数据库上验证α-MDVR 方差减少策略的有效性;其次验证不同参数α对α-MDVR 性能的影响,并同时比较算法的性能.为了公平比较,在实验中所涉及的 4 个标准数据库上都采取统一的随机方法产生样本,且各算法在 4 个数据库上均运行 10 次,将输出结果的平均值作为最终输出.本文实验中所涉及的参数,在一定的范围内,利用网格搜索的方法,选择目标函数下降最快的那一组参数为本文实验所用参数.

Table 1 Standard database description

表 1 标准数据库描述

数据库	训练样本数	测试样本数	维数
astro-physics	29 882	32 487	99 757
a9a	24 703	7 858	123
rcv1	20 242	677 399	47 236
mnist	60 000	10 000	780

3.1 减小方差验证实验

本实验的目的是比较α-MDVR 和 COMID 的实际方差减小效果.图 1 为 3 种算法的方差比较结果图,其中,横坐标表示迭代次数 s;纵坐标表示算法的方差(var),即,每次迭代所用的梯度与全梯度之间差值的平方.

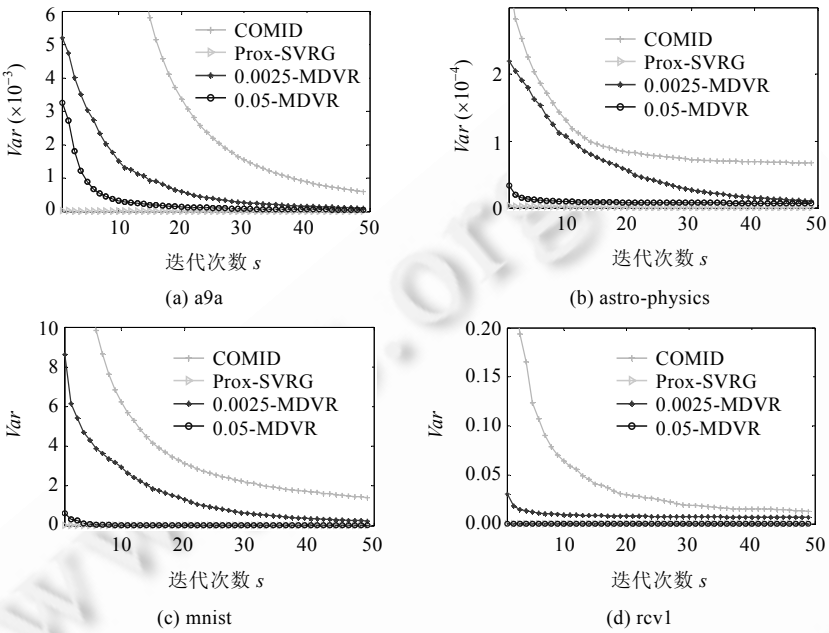


Fig.1 Variance comparison chart

图 1 方差比较图

由图 1 中可以看出,α-MDVR 方差的减小程度明显都优于 COMID,但比使用全部样本修正梯度的方差减小策略要差.即,α取值越大,方差减小的效果越好(实验中,COMID 算法的α取值为 1/n,其中,n 是样本数,Prox-SVRG 算法的α取值为 1).该实验说明,α-MDVR 具有一定的方差减小效果.

3.2 α对α-MDVR实际性能的影响

本实验主要目的是在 4 个标准数据库上考察不同参数α对α-MDVR 算法实际性能的影响.α取不同的值,表示算法在迭代过程中修正梯度时使用了不同的样本数.例如在图 2(c)中,当α=1 时,表示修正梯度时需要遍历整个训练集合,即,使用了 60 000 个样本;当α=0.05 时,表示修正梯度时使用了 3 000 个样本;当α=0.0025 时,表示修正梯度时使用了 150 个样本.为方便起见,我们记α=0 时的α-MDVR 表示 COMID 算法,即,不使用任何样本进行梯度修正.

图 2 为α取不同参数时目标函数收敛情况的比较结果图,其中,横坐标表示 CPU 时间,纵坐标表示目标函数值(object value).从图中可以看出,当α取合适的值时,α-MDVR 的收敛速度比使用全部训练样本修正梯度的算法以及不使用任何样本修正梯度的 COMID 收敛速度都要快.特别是在上述 4 个数据库中,当α取 0.05 时,α-MDVR 的性能最佳.

由于随机优化每步迭代所需要的 CPU 时间基本相同,因此,算法的时间复杂度主要体现在算法达到相同精度所需要的迭代步数上[18,19],即,收敛速率.从图 2 可以看出,α-MDVR 算法的收敛速率曲线的下降速度快于 COMID 算法,这说明α-MDVR 算法的时间复杂度更低.

综合第 3.1 节和第 3.2 节的实验,虽然α-MDVR 方差的减小效果达不到 SVRG 中使用全部样本时的效果,但当取α取合适的值时,总体计算时间却减少了很多.另一方面,虽然 COMID 不使用任何样本进行梯度修正,会节省计算时间,但导致的方差仍然会影响实际的收敛速度.总之,对于求解非光滑损失优化问题,α-MDVR 在减

少方差和缩短运行时间方面具有很好的柔韧性,比 COMID 具有更快的实际收敛速率.

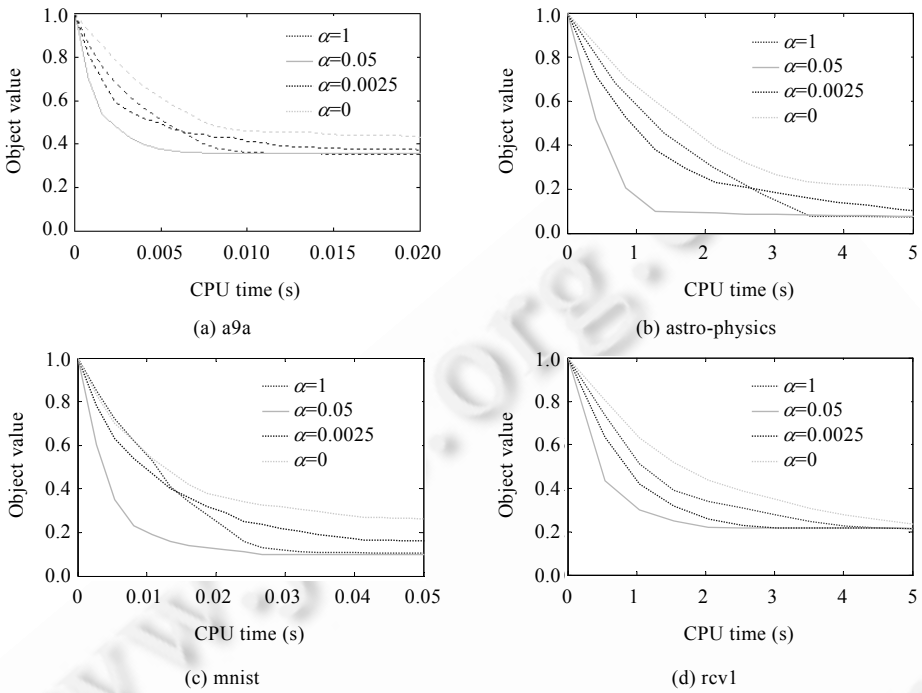


Fig.2 When α taking different values, convergence speed comparison chart of α-MDVR

图 2 α取不同值时,α-MDVR 的收敛速度比较图

4 总 结

本文针对“L1+Hinge”随机非光滑优化问题.我们首先证明了 COMID 具有最优的方差形式的收敛速率 $O\left(1/\sqrt{T}+\sigma^2/\sqrt{T}\right)$, 进而在 COMID 中引入方差减小策略,得到了一种α-MDVR 算法,并通过实验验证了α-MDVR 能够在适度减小方差的同时节省 CPU 时间,比 COMID 具有更快的实际收敛速率.

与文献[9]中 COMID 的一般性类似,本文提出的算法很容易推广到求解更一般损失函数和正则化项的优化问题,如混合范数的正则化项问题以及矩阵形式的正则化损失函数优化问题等.

References:

[1] Tseng P. Approximation accuracy, gradient methods, and error bound for structured convex optimization. Mathematical Programming, 2010,125(2):263−295. [doi: 10.1007/s10107-010-0394-2]

[2] Nemirovski A, Juditsky A, Lan G, Shapiro A. Robust stochastic approximation approach to stochastic programming. SIAM Journal on Optimization, 2009,19(4):1574−1609. [doi: 10.1137/070704277]

[3] Shalev-Shwartz S, Tewari A. Stochastic methods for L1 regularized loss minimization. In: Proc. of the 26th Annual Int’l Conf. on Machine Learning. 2009. 929−936.

[4] Johnson R, Zhang T. Accelerating stochastic gradient descent using predictive variance reduction. In: Proc. of the Advances in Neural Information Processing Systems 26. 2013. 315−323.

[5] Shalev-Shwartz S, Zhang T. Stochastic dual coordinate ascent methods for regularized loss minimization. arXiv preprint arXiv: 1209.1873, 2012.

[6] Le Roux N, Schmidt M, Bach F. A stochastic gradient method with an exponential convergence rate for strongly convex optimization with finite training sets. arXiv preprint, arXiv: 1202.6258, 2012.

[7] Xiao L. Dual averaging methods for regularized stochastic learning and online optimization. In: Advances in Neural Information Processing Systems. 2009. 2116−2124.

[8] Xiao L, Zhang T. A proximal stochastic gradient method with progressive variance reduction. arXiv: 1403.4699v1, 2014.

[9] Duchi J, Shalev-Shwartz S, Singer Y, Tewari A. Composite objective mirror descent. In: Proc. of the 23rd Annual Workshop on Computational Learning Theory. ACM Press, 2010. 116−128.

[10] Duchi J, Shalev-Shwartz S, Singer Y. Efficient projections onto the L1-ball for learning in high dimensions. In: Proc. of the 25th Int'l Conf. on Machine Learning. 2008. 272−279.

[11] Beck A, Teboulle M. A fast iterative shrinkage thresholding algorithm for linear inverse problems. SIAM Journal on Imaging Sciences, 2009,2(1):183−202. [doi: 10.1137/080716542]

[12] Nesterov Y. A method of solving a convex programming problem with convergence rate $O(1/k^2)$. Soviet Mathematics Doklady, 1983,27(2):372−376.

[13] Bottou L. Stochastic Gradient Descent Tricks. Neural Networks: Tricks of the Trade. Berlin, Heidelberg: Springer-Verlag, 2012. 421−436. [doi: 10.1007/978-3-642-35289-8_25]

[14] Beck A, Teboulle M. Mirror descent and nonlinear projected subgradient methods for convex optimization. Operations Research Letters, 2003,31(3):167−175. [doi: 10.1016/S0167-6377(02)00231-6]

[15] Bregman LM. The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming. USSR Computational Mathematics and Mathematical Physics, 1967,7(3):200−217. [doi: 10.1016/0041-5553(67)90040-7]

[16] Hazan E, Kale S. Beyond the regret minimization barrier: An optimal algorithm for stochastic strongly convex optimization. Journal of Machine Learning Research, 2014,15(1):2489−2512.

[17] Lan G. An optimal method for stochastic composite optimization. Mathematical Programming, Series A, 2012,133(1-2):365−397. [doi: 10.1007/s10107-010-0434-y]

[18] Shalev-Shwartz S, Singer Y, Srebro N. Pegasos: Primal estimated sub-gradient solver for SVM. Mathematical Programming, 2011, 127(1):3−30. [doi: 10.1007/s10107-010-0420-4]

[19] Lin QH, Chen X, Peña J. A sparsity preserving stochastic gradient method for composite optimization. Manuscript, Carnegie Mellon University, 2011. 15213.

附录:定理 2 的证明

证明:将 $B_\varphi(\boldsymbol{w},\boldsymbol{w}_t)=\|\boldsymbol{w}-\boldsymbol{w}_t\|^2$ 代入公式(8)中,算法的迭代步骤为

$$\boldsymbol{w}_{t+1}=\text{argmin}\{\eta_t\langle \boldsymbol{g}_t(\boldsymbol{w}_t,\xi_t),\boldsymbol{w}\rangle+\eta_t\lambda\|\boldsymbol{w}\|_1+\|\boldsymbol{w}-\boldsymbol{w}_t\|^2\}.$$

令 $\boldsymbol{w}=\boldsymbol{w}_*$且 $r'(\boldsymbol{w}_t)=\partial r(\boldsymbol{w}_t)$,有如下的一阶必要条件成立:

$$\eta_t\langle \boldsymbol{g}_t(\boldsymbol{w}_t,\xi_t)+\eta_t r'(\boldsymbol{w}_{t+1})+\nabla\varphi(\boldsymbol{w}_{t+1})-\nabla\varphi(\boldsymbol{w}_t),\boldsymbol{w}_*-\boldsymbol{w}_t\rangle\geqslant 0 \tag{10}$$

根据凸函数的性质,有如下不等式成立:

$$\begin{aligned}\eta_t[f_t(\boldsymbol{w}_t)-f_t(\boldsymbol{w}_*)]+\eta_t[r(\boldsymbol{w}_{t+1})-r(\boldsymbol{w}_*)] &\leqslant \eta_t\langle \boldsymbol{w}_t-\boldsymbol{w}_*,\boldsymbol{g}_t(\boldsymbol{w}_t,\xi_t)\rangle+\eta_t\langle \boldsymbol{w}_{t+1}-\boldsymbol{w}_*,r'(\boldsymbol{w}_{t+1})\rangle\\ &=\eta_t\langle \boldsymbol{w}_t-\boldsymbol{w}_{t+1},\boldsymbol{g}_t(\boldsymbol{w}_t,\xi_t)\rangle+\eta_t\langle \boldsymbol{w}_{t+1}-\boldsymbol{w}_*,\boldsymbol{g}_t(\boldsymbol{w}_t,\xi_t)\rangle+\\ &\quad\eta_t\langle \boldsymbol{w}_{t+1}-\boldsymbol{w}_*,r'(\boldsymbol{w}_{t+1})\rangle\\ &=\eta_t\langle \boldsymbol{w}_t-\boldsymbol{w}_{t+1},\boldsymbol{g}_t(\boldsymbol{w}_t,\xi_t)\rangle-\langle \boldsymbol{w}_*-\boldsymbol{w}_{t+1},\eta_t\boldsymbol{g}_t(\boldsymbol{w}_t,\xi_t)+\eta_t r'(\boldsymbol{w}_{t+1})+\\ &\quad\nabla\varphi(\boldsymbol{w}_{t+1})-\nabla\varphi(\boldsymbol{w}_t)\rangle+\langle \boldsymbol{w}_*-\boldsymbol{w}_{t+1},\nabla\varphi(\boldsymbol{w}_{t+1})-\nabla\varphi(\boldsymbol{w}_t)\rangle\\ &\leqslant\eta_t\langle \boldsymbol{w}_t-\boldsymbol{w}_{t+1},\boldsymbol{g}_t(\boldsymbol{w}_t,\xi_t)\rangle+\langle \boldsymbol{w}_*-\boldsymbol{w}_{t+1},\nabla\varphi(\boldsymbol{w}_{t+1})-\nabla\varphi(\boldsymbol{w}_t)\rangle\\ &=\eta_t\langle \boldsymbol{w}_t-\boldsymbol{w}_{t+1},\boldsymbol{g}_t(\boldsymbol{w}_t,\xi_t)\rangle+\langle \boldsymbol{w}_{t+1}-\boldsymbol{w}_*,\eta_t\boldsymbol{g}_t(\boldsymbol{w}_t,\xi_t)+\eta_t r'(\boldsymbol{w}_{t+1})\rangle\end{aligned} \tag{11}$$

最后一个不等式可由不等式(10)得到,此时,不等式(11)变为

$$\begin{aligned}\eta_t[f_t(\boldsymbol{w}_t)-f_t(\boldsymbol{w}_*)]+\eta_t[r(\boldsymbol{w}_{t+1})-r(\boldsymbol{w}_*)] &\leqslant \eta_t\langle \boldsymbol{w}_t-\boldsymbol{w}_{t+1},\boldsymbol{g}_t(\boldsymbol{w}_t,\xi_t)\rangle+\langle \boldsymbol{w}_*-\boldsymbol{w}_{t+1},\nabla\varphi(\boldsymbol{w}_{t+1})-\nabla\varphi(\boldsymbol{w}_t)\rangle\\ &=\eta_t\langle \boldsymbol{w}_t-\boldsymbol{w}_{t+1},\boldsymbol{g}_t(\boldsymbol{w}_t,\xi_t)-\nabla\boldsymbol{F}(\boldsymbol{w}_t)\rangle+\eta_t\langle \boldsymbol{w}_t-\boldsymbol{w}_{t+1},\nabla\boldsymbol{F}(\boldsymbol{w}_t)\rangle+\\ &\quad B_\varphi(\boldsymbol{w}_*-\boldsymbol{w}_t)-B_\varphi(\boldsymbol{w}_*-\boldsymbol{w}_{t+1})-B_\varphi(\boldsymbol{w}_{t+1}-\boldsymbol{w}_t)\\ &=\eta_t\left\langle\sqrt{\frac{1}{\eta_t}}(\boldsymbol{w}_t-\boldsymbol{w}_{t+1}),\sqrt{\eta_t}(\boldsymbol{g}_t(\boldsymbol{w}_t,\xi_t)-\nabla\boldsymbol{F}(\boldsymbol{w}_t))\right\rangle+\\ &\quad \eta_t\left\langle\sqrt{\frac{1}{\eta_t}}(\boldsymbol{w}_t-\boldsymbol{w}_{t+1}),\sqrt{\eta_t}\nabla\boldsymbol{F}(\boldsymbol{w}_t)\right\rangle+\\ &\quad B_\varphi(\boldsymbol{w}_*-\boldsymbol{w}_t)-B_\varphi(\boldsymbol{w}_*-\boldsymbol{w}_{t+1})-B_\varphi(\boldsymbol{w}_{t+1}-\boldsymbol{w}_t)\\ &\leqslant \eta_t\left(\frac{\frac{1}{\eta_t}\|\boldsymbol{w}_t-\boldsymbol{w}_{t+1}\|^2+\eta_t\|\boldsymbol{g}_t(\boldsymbol{w}_t,\xi_t)-\boldsymbol{F}F(\boldsymbol{w}_t)\|^2}{2}\right)+\\ &\quad \eta_t\left(\frac{\frac{1}{\eta_t}\|\boldsymbol{w}_t-\boldsymbol{w}_{t+1}\|^2+\eta_t\|\nabla\boldsymbol{F}(\boldsymbol{w}_t)\|^2}{2}\right)+\\ &\quad B_\varphi(\boldsymbol{w}_*-\boldsymbol{w}_t)-B_\varphi(\boldsymbol{w}_*-\boldsymbol{w}_{t+1})-B_\varphi(\boldsymbol{w}_{t+1}-\boldsymbol{w}_t)\\ &=\frac{1}{2}\|\boldsymbol{w}_t-\boldsymbol{w}_{t+1}\|^2+\frac{\eta_t^2}{2}\|\boldsymbol{g}_t(\boldsymbol{w}_t,\xi_t)-\nabla\boldsymbol{F}(\boldsymbol{w}_t)\|^2+\frac{1}{2}\|\boldsymbol{w}_t-\boldsymbol{w}_{t+1}\|^2+\\ &\quad \frac{\eta_t^2}{2}\|\nabla\boldsymbol{F}(\boldsymbol{w}_t)\|^2+\|\boldsymbol{w}_*-\boldsymbol{w}_t\|^2-\|\boldsymbol{w}_*-\boldsymbol{w}_{t+1}\|^2-\|\boldsymbol{w}_{t+1}-\boldsymbol{w}_t\|^2\\ &=B_\varphi(\boldsymbol{w}_*-\boldsymbol{w}_t)-B_\varphi(\boldsymbol{w}_*-\boldsymbol{w}_{t+1})+\frac{\eta_t^2}{2}\|\boldsymbol{g}_t(\boldsymbol{w}_t,\xi_t)-\nabla\boldsymbol{F}(\boldsymbol{w}_t)\|^2+\\ &\quad \frac{\eta_t^2}{2}\|\nabla\boldsymbol{F}(\boldsymbol{w}_t)\|^2\end{aligned}\tag{12}$$

第 2 个不等式可根据$\langle a\boldsymbol{w}_1,b\boldsymbol{w}_2\rangle\leqslant a^2\|\boldsymbol{w}_1\|^2+b^2\|\boldsymbol{w}_2\|^2$得到,将公式(12)两边同时除以 $1/\eta_t$,可得:

$$\begin{aligned}f_t(\boldsymbol{w}_t)+r(\boldsymbol{w}_t)-f_t(\boldsymbol{w}_*)-r(\boldsymbol{w}_*)\leqslant &\frac{1}{\eta_t}B_\varphi(\boldsymbol{w}_*,\boldsymbol{w}_t)-\frac{1}{\eta_t}B_\varphi(\boldsymbol{w}_*,\boldsymbol{w}_{t+1})+\frac{\eta_t}{2}\|\boldsymbol{g}_t(\boldsymbol{w}_t,\xi_t)-\nabla\boldsymbol{F}(\boldsymbol{w}_t)\|^2+\\ &\frac{\eta_t}{2}\|\nabla\boldsymbol{F}(\boldsymbol{w}_t)\|^2+\frac{1}{\eta_t}[r(\boldsymbol{w}_t)-r(\boldsymbol{w}_{t+1})]\end{aligned}\tag{13}$$

我们假设 $E[\|\nabla\boldsymbol{F}(\boldsymbol{w})\|]\leqslant M$, $E_\xi[\|\boldsymbol{g}_t(\boldsymbol{w}_t,\xi)-\nabla\boldsymbol{F}(\boldsymbol{w}_t)\|^2]=\sigma_t^2\leqslant\sigma^2$, 对不等式(13)两边同时取期望,并从 t=1 到 T 求和,得到:

$$\begin{aligned}\sum_{t=1}^{T}E[f_t(\boldsymbol{w}_t)+r(\boldsymbol{w}_t)-f_t(\boldsymbol{w}_*)-r(\boldsymbol{w}_*)] &\leqslant \sum_{t=1}^{T}E\left[\frac{1}{\eta_t}B_\varphi(\boldsymbol{w}_*,\boldsymbol{w}_t)-\frac{1}{\eta_t}B_\varphi(\boldsymbol{w}_*,\boldsymbol{w}_{t+1})\right]+\frac{M^2}{2}\sum_{t=1}^{T}\eta_t+\frac{1}{2}\sum_{t=1}^{T}\eta_t\sigma_t^2+\\ &\quad \sum_{t=1}^{T}[r(\boldsymbol{w}_t)-r(\boldsymbol{w}_{t+1})]\\ &=\frac{1}{\eta_1}B_\varphi(\boldsymbol{w}_*,\boldsymbol{w}_1)+\sum_{t=2}^{T}B_\varphi(\boldsymbol{w}_*,\boldsymbol{w}_t)\left(\frac{1}{\eta_t}-\frac{1}{\eta_{t-1}}\right)-\frac{1}{\eta_T}B_\varphi(\boldsymbol{w}_*,\boldsymbol{w}_{T+1})+\\ &\quad \frac{M^2}{2}\sum_{t=1}^{T}\eta_t+\frac{1}{2}\sum_{t=1}^{T}\eta_t\sigma_t^2+r(\boldsymbol{w}_1)-r(\boldsymbol{w}_{T+1})\\ &\leqslant \frac{1}{\eta_1}B_\varphi(\boldsymbol{w}_*,\boldsymbol{w}_1)+\sum_{t=2}^{T}B_\varphi(\boldsymbol{w}_*,\boldsymbol{w}_t)\left(\frac{1}{\eta_t}-\frac{1}{\eta_{t-1}}\right)+\frac{M^2}{2}\sum_{t=1}^{T}\eta_t+\frac{1}{2}\sum_{t=1}^{T}\eta_t\sigma_t^2+\\ &\quad r(\boldsymbol{w}_1)\end{aligned}\tag{14}$$

取 $B_\varphi(\boldsymbol{w},\boldsymbol{w}_t)=\|\boldsymbol{w}-\boldsymbol{w}_t\|^2, E[\|\boldsymbol{w}_1-\boldsymbol{w}_*\|]\leqslant R, r(\boldsymbol{w}_1)=0, \eta_t=1/\sqrt{t}$,因为 $\sum_{t=1}^{T}\eta_t=\sum_{t=1}^{T}\frac{1}{\sqrt{t}}\leqslant 2\sqrt{T}-1$,可得:

$$\begin{aligned}\sum_{t=1}^{T}E[f_t(\boldsymbol{w}_t)+r(\boldsymbol{w}_t)-f_t(\boldsymbol{w}_*)-r(\boldsymbol{w}_*)]&\leqslant\frac{1}{\eta_1}B_\varphi(\boldsymbol{w}_*,\boldsymbol{w}_1)+\sum_{t=2}^{T}B_\varphi(\boldsymbol{w}_*,\boldsymbol{w}_t)\left(\frac{1}{\eta_t}-\frac{1}{\eta_{t-1}}\right)+\\&\quad\frac{M^2}{2}\sum_{t=1}^{T}\eta_t+\frac{1}{2}\sum_{t=1}^{T}\eta_t\sigma_t^2+r(\boldsymbol{w}_1)\\&\leqslant(M^2+R^2)\sqrt{T}+\frac{1}{2}\sum_{t=1}^{T}\eta_t\sigma_t^2\end{aligned}\tag{15}$$

当我们取固定的方差上界σ,不等式(15)变为

$$\sum_{t=1}^{T}E[f_t(\boldsymbol{w}_t)+r(\boldsymbol{w}_t)-f_t(\boldsymbol{w}_*)-r(\boldsymbol{w}_*)]\leqslant(M^2+R^2)\sqrt{T}+\frac{1}{2}\sum_{t=1}^{T}\eta_t\sigma^2\leqslant(M^2+R^2)\sqrt{T}+\sigma^2\sqrt{T}\tag{16}$$

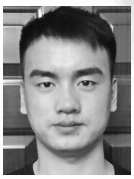
不等式(16)两边同时乘以 $1/T$:

$$\frac{1}{T}\sum_{t=1}^{T}E[f_t(\boldsymbol{w}_t)+r(\boldsymbol{w}_t)-f_t(\boldsymbol{w}_*)-r(\boldsymbol{w}_*)]=\frac{1}{T}\sum_{t=1}^{T}E[P(\boldsymbol{w}_t)-P(\boldsymbol{w}_*)]\leqslant\frac{(M^2+R^2)}{\sqrt{T}}+\frac{\sigma^2}{\sqrt{T}}.$$

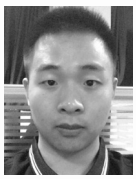
根据凸函数的性质 $\left[P\left(\frac{1}{T}\sum_{t=1}^{T}\boldsymbol{w}_t\right)\right]\leqslant\frac{1}{T}\sum_{t=1}^{T}P(\boldsymbol{w}_t)$,可得到:

$$E\left[P\left(\frac{1}{T}\sum_{t=1}^{T}\boldsymbol{w}_t\right)-P(\boldsymbol{w}_*)\right]\leqslant\frac{1}{T}\sum_{t=1}^{T}E[P(\boldsymbol{w}_t)-P(\boldsymbol{w}_*)]\leqslant\frac{(M^2+R^2)}{\sqrt{T}}+\frac{\sigma^2}{\sqrt{T}}.$$

证毕. □



朱小辉(1989－),男,安徽芜湖人,硕士,主要研究领域为模式识别,人工智能.



邵言剑(1990－),男,硕士,主要研究领域为凸优化及其在机器学习中的应用.

陶卿(1965－),男,博士,教授,博士生导师,CCF 高级会员,主要研究领域为机器学习,模式识别,应用数学.

储德军(1978－),男,博士,讲师,主要研究领域为模式识别,凸优化算法及其在机器学习中的应用.