Advances in Applied Mathematics 应用数学进展, **2022, 11(7), 4708-4717**
Published Online July 2022 in Hans. http://www.hanspub.org/journal/aam
https://doi.org/10.12677/aam.2022.117497

Hans汉斯

一种基于方差缩减的临近随机牛顿算法

杜康乐

浙江师范大学，数学与计算机科学学院，浙江 金华

收稿日期：2022年6月15日；录用日期：2022年7月12日；发布日期：2022年7月19日

摘 要

本文研究了优化问题中的一类复合优化问题。对于凸非光滑的目标函数，在临近牛顿算法的基础上，引入方差缩减的方法，提出了一种新的——基于方差缩减的随机牛顿算法($SNVR$)，并进行了收敛性分析。与$ProxSGD, ProxGD, ProxSVRG$方法相比，$SNVR$有更快的收敛速度。

关键词

复合优化问题，机器学习，方差缩减，临近随机算法，牛顿算法

A Proximal Random Newton Algorithm Based on Variance Reduction

Kangle Du

College of Mathematics and Computer Science, Zhejiang Normal University, Jinhua Zhejiang

Received: Jun. 15^{th}, 2022; accepted: Jul. 12^{th}, 2022; published: Jul. 19^{th}, 2022

Abstract

This paper studies a class of compound optimization problems in optimization problems. For convex nonsmooth objective functions, a new random Newton algorithm

文章引用: 杜康乐. 一种基于方差缩减的临近随机牛顿算法[J]. 应用数学进展, 2022, 11(7): 4708-4717.
DOI: 10.12677/aam.2022.117497

based on variance reduction (SNVR) is proposed by introducing the method of variance reduction on the basis of proximal Newton algorithm, and the convergence is analyzed. Compared with the methods of ProxSGD, ProxGD, proxSVRG , SNVR has better robustness and scalability, and has faster convergence speed.

Keywords

Compound Optimization Problems, Machine Learning, Variance Reduce, Proximal Stochastic Method, Newton-Type Method

1. 引言

复合优化问题是优化问题中一类重要的问题, 近年来, 随着机器学习的兴起, 越来越来的学者开始关注此类问题. 在机器学习中, 有很多问题可以转化为一个优化问题, 比如回归模型, 由于数据量大, 这类问题可以转化为复合优化问题. 本文研究的问题如下:

$$\min_{w\in R^d} \Phi(w) = F(w) + \Omega(w) = \frac{1}{N}\sum_{i=1}^{N} f_i(w) + \Omega(w), \tag{1.1}$$

其中, $F: R^d \to R$, 由N个光滑凸函数$f_i(w)$ $(i = 1, 2, \cdots, N)$组成, Ω 通常为非光滑凸函数. 此类问题称为经验风险极小化问题或者采样平均极小化问题. 这个问题通常是难以求解的, 一方面是因为样本N比较多(因此函数值、梯度计算代价比较高), 另一方面是因为优化问题的可行域所在空间维数d比较大.

对于该问题的求解, 最初的算法是全梯度下降法(GD), 即每次迭代过程中用所有数据点来计算梯度. 由于机器学习中往往数据量大且维数高, 所以全梯度下降法并不常用, 后来, $Bottou$ [1]等提出随机梯度下降法, 该方法在每次迭代过程中随机选取一个点, 用此点处的梯度代替全梯度, 此算法能够极大的降低计算量, 但是与全梯度法相比引入了一个方差, 降低了收敛速度. 在此基础上, 又有学者提出小批量随机梯度法. 在后来的研究中, $Byra$等[2]提出一种随机拟牛顿算法; Lee 等[3]提出了一种临近牛顿型方法, 利用牛顿法收敛更快的特性提高了收敛速度; $Xiao$等[4]提出了一种基于方差缩减的临近随机梯度法$(ProxSVRG)$; $Moritz$等[5]提出一种限制型拟牛顿法的方法$(LBFGS)$; $Defazio$ 等[6]考虑每一步的迭代的梯度, 提出了另一种减少方差的算法$(SAGA)$; Jin等[7]提出一种基于共轭算法的梯度下降法.

对于问题(1.1), 本文是在临近随机牛顿法的基础之上, 引入了方差缩减的技术, 提出了一种新的随机梯度算法——基于方差缩减的随机牛顿算法($SNVR$).

2. 基于方差缩减的临近随机牛顿算法

在随机梯度下降法(SGD)中, 每一步迭代仅仅使用了当前点的随机梯度, 而迭代计算的历史梯度则直接丢弃不再使用. 然而当迭代值接近收敛时, 上一步的随机梯度同样也是当前迭代点处梯度的一个很好的估计, 带有方差缩减的随机平均梯度法($SAGA$)就是基于这一想法构造. 在每次迭代, ($SAGA$)算法记录所有之前计算过的随机梯度, 再与当前新计算的随机梯度求平均, 最终作为下一步的梯度估计, 具体来说, ($SAGA$) 算法在内存中开辟了存储N个随机梯度的空间$[\nabla F_1(w), \nabla F_2(w), \cdots, \nabla F_N(w)]$, 分别用于记录和第$i$个样本相关的最新的随机梯度. 在第$k$步更新时, 若抽取的样本点下标为$s_k$, 则计算随机梯度后将$\nabla F_{s_k}(w_k)$的值更新为当前的随机梯度值, 而其他未抽取到的下标对应的$\nabla F_j(w_k)$保持不变. 每次($SAGA$) 算法更新使用的梯度方向是

$$v_k = \nabla F_{s_k}(w_k) - \nabla F_{s_k}(w_{k-1}) + \nabla F(w_{k-1}),$$

其中, $s_k \in \{1, 2, \cdots, N\}$是随机选取的一个数据点.

注意到给定$s_1, s_2, \cdots, s_{k-1}$ 时, w_k, w_{k-1}均为定值, 由v_k的表达式可知

$$\begin{aligned}
&E[v_k|s_1, s_2, \cdots, s_{k-1}] \\
&= E[\nabla F_{s_k}(w_k)|s_k] - E[\nabla F_{s_k}(w_{k-1}) - \nabla F(w_{k-1})|s_1, s_2, \cdots, s_{k-1}] \\
&= \nabla F(w_k) - 0 \\
&= \nabla F(w_k),
\end{aligned}$$

即v_k在条件期望意义下是$\nabla F(x_k)$的一个无偏估计.

对于非光滑的目标函数, 无法直接求梯度, 我们引用临近算子, 其定义如下:

$$prox_{g,\alpha}^{H}(v) = arg \min_{w \in R^d} \{g(w) + \frac{1}{2\alpha}||w - v||_H^2\},$$

其中, $\alpha > 0$, H 为正定矩阵, $\|w - v\|_H^2 = (w - v)^T H(w - v)$, 记为$\|w\|^2 = ||w||_{I_d}^2 = w^T I_d w = w^T w$, 其中, I_d 为d 阶单位阵.

问题(1.1)的临近牛顿型算法[3]的一般形式为:

$$w_{k+1} = prox_{\Phi,\alpha_k}^{H_k}\left(w_k - \alpha_k H_k^{-1} \nabla F\left(w_k\right)\right), \tag{2.1}$$

其中, H_k^{-1}是函数F的二阶矩阵的逆$\left(\nabla^2 F(w_k)\right)^{-1}$, $\nabla F\left(w_k\right)$是N个数据点的梯度.

在机器学习模型训练过程中, 数据量往往比较大, 维度也很高, 计算量会很大, 在每次迭代的过程中, 如果选取所有数据点计算梯度, 消耗量会很大; 如果随机选取一个数据点计算梯度, 波动性会很大, 所以采用批量梯度的方法, 即每次迭代随机选取小批量的数据点. 记$S_k \subseteq \{1, 2, \cdots, N\}$,

∇F_{S_k}表达式为

$$\nabla F_{S_k} = \frac{\sum_{i\in S_k} \nabla f_i\left(w_k\right)}{b_k},$$

其中, $b_k = |S_k|$, 表示集合S_k中元素的个数.

为了降低噪声的影响, 本文再采用方差缩减技术, 即

$$v_k = \nabla F_{S_k}(w_k) - \nabla F_{S_k}(w_{k-1}) + \nabla F(w_{k-1}),$$

用v_k替代$\nabla F(w_k)$, (2.1)式就变成了如下形式:

$$w_{k+1} = prox_{\Phi,\alpha_k}^{H_k}\left(w_k - \alpha_k H_k^{-1} v_k\right). \tag{2.2}$$

通过(2.2)式迭代, 本文建立基于方差缩减的随机牛顿算法($SNVR$), 如下:

步1: 选取初始点w_0, 给定 b, eps 的值, $k = 0$;

步2: 计算并存储梯度$[\nabla F_1(w_k), \nabla F_2(w_k), \cdots, \nabla F_N(w_k)]$, 求$w_0$处$Hessian$ 矩阵H, 求逆, $w_1 = prox_{\Omega,\alpha}^{H}(w_0 - \alpha H_0^{-1}\nabla F(w_0))$, $k = 1$;

步3: 从集合$\{1, 2, \cdots, N\}$中随机选取一个子集S_k, 其中元素的个数为b, 计算$\nabla F_i(w_k)$, 其中$i \in S_k$, 计算$v_k = \nabla F_{S_k}\left(w_k\right) - \nabla F_{S_k}(w_{k-1}) + \nabla F(w_{k-1})$;

步4: 计算w_k处$Hessian$矩阵H, 并求逆, 由H^{-1}计算$w_{k+1} = prox_{\Omega,\alpha_k}^{H}\left(w_k - \alpha_{k-1}H^{-1}v_{k-1}\right)$

步5: 更新梯度, 用$\nabla F_i(w_k)$替换上一步相应位置的梯度，其余梯度保持不变;

步6：若$|\Phi(w_{k+1}) - \Phi(w_k)| < eps$, 则算法停止, 否则, 转步3.

在以上算法过程中, 步4中α_k是步长, 采用常用的步长搜索方法即可; 步3是求出迭代方向, 步4是求出下一个迭代点, 步5是更新梯度.

3. 收敛性分析

假设3.1 分量函数f_i是u_i强凸的, 并且它的梯度函数满足$L_i - lipschitz$连续, 即对任意的$w, v \in R^d$, 有

$$\frac{u_i}{2}\|v - w\|^2 \le f_i(v) - f_i(w) - (v - w)^T\nabla f_i(w) \le \frac{L_i}{2}\|v - w\|^2.$$

等价的, $u_i I_d \preceq \nabla^2 f_i\left(w\right) \preceq L_i I_d$. 则$F\left(w\right) = \frac{1}{N}\sum_{i=1}^{N} f_i\left(w\right)$是强凸的, 并且它的梯度满足$L - lipschitz$连续, 此处,

$$\min\left(u_i\right) = u \le \frac{1}{N}\sum_{i}^{N} u_i, \ \ \max\left(L_i\right) = L \ge \frac{1}{N}\sum_{i}^{N} L_i.$$

在假设3.1和函数Ω凸的情况下, 可以得出问题(1.1)中目标函数Φ是$u-$强凸的.

假设3.2 对任意的非空子集$S \subseteq \{1, 2, \cdots, N\}$, 有$uI_d \preceq \nabla^2 F_S\left(w\right) \preceq LI_d$.

基于假设3.2, 存在常数$0 < \lambda_1 < \lambda_2$, 使得w_k处的$Hessian$矩阵H_k 满足:

$$\lambda_1 I \preceq H_k^{-1} \preceq \lambda_2 I.$$

设$p = prox_{\Omega}^{H}(u)$,$q = prox_{\Omega}^{H}(\bar{u})$, 因为$\Omega$为凸函数, 所以

$$||p - q|| \leq \lambda_1/\lambda_2||u - \bar{u}||. \tag{3.1}$$

引理3.1 设w_*是函数F的唯一最小值, 可以得到

$$E\|v_k - F(w_k)\|^2 \leqslant 4L[\Phi(w_k) - \Phi(w_*) + \Phi(w_{k-1}) - \Phi(w_*)]. \tag{3.2}$$

证明 对于任意给定的小批量S_k, 考虑以下函数

$$h_{S_k}(w) = F_{S_k}(w) - F_{S_k}(w_*) - \nabla F_{S_k}(w_*)^T(w - w_*),$$

因为$\nabla h_{S_k}(w_*) = 0$, 所以$h_{S_k}(w_*) = \min_w h_{S_k}(w)$, 则

$$\begin{aligned}
0 = h_{S_k}(w_*) &\leq \min_{\alpha} h_{S_k}(w - \alpha\nabla h_{S_k}(w)) \\
&\leq \min_{\alpha}(h_{S_k}(w) - \alpha\|\nabla h_{S_k}(w)\|^2 + \frac{1}{2}L\alpha^2\|\nabla h_{S_k}(w)\|^2) \\
&= h_{S_k}(w) - \frac{1}{2L}\|\nabla h_{S_k}(w)\|^2,
\end{aligned}$$

可得

$$\|\nabla F_{S_k}(w) - \nabla F_{S_k}(w_*)\|^2 \leq 2L[F_{S_k}(w) - F_{S_k}(w_*) - \nabla F_{S_k}(w_*)^T(w - w_*)].$$

通过对n个样本随机选取的小批量S_k求和, 且由$\nabla F(w_*) = 0$, 可得

$$\binom{N}{b}^{-1} \sum_{|S_k|=b} \|\nabla F_{S_k}(w) - \nabla F_{S_k}(w_*)\|^2 \leq 2L[F(w) - F(w_*)].$$

考虑w_k, w_{k-1}, 对于S_k求期望, 可得

$$\begin{aligned}
&E\|v_k - \nabla F(w_k)\|^2 \leq E\|v_k\|^2 \\
&= E\|\nabla F_{S_k}(w_k) - \nabla F_{S_k}(w_*) + \nabla F_{S_k}(w_*) - \nabla F_{S_k}(w_{k-1}) + \nabla F(w_{k-1})\|^2 \\
&\leq 2E\|\nabla F_{S_k}(w_k) - \nabla F_{S_k}(w_*)\|^2 + 2E\|\nabla F_{S_k}(w_*) - \nabla F_{S_k}(w_{k-1}) + \nabla F(w_{k-1})\|^2 \\
&\leq 2E\|\nabla F_{S_k}(w_k) - \nabla F_{S_k}(w_*)\|^2 + 2E\|\nabla F_{S_k}(w_*) - \nabla F_{S_k}(w_{k-1})\|^2,
\end{aligned}$$

第一个不等式和第二个不等式用了$E\|\xi - E\xi\|^2 \leq E\|\xi\|^2$, 由上式, 可得

$$E\|v_k - F(w_k)\|^2 \leqslant 4L[\Phi(w_k) - \Phi(w_*) + \Phi(w_{k-1}) - \Phi(w_*)],$$

得证.

引理3.2 [4] 若$\Phi(w)=F(w)+\Omega(w)$, 且$\nabla F(w)$满足$L-lipschitz$连续, $F(w)$和$\Omega(w)$分别满足u_F和u_Ω强凸, 则对任意的$w,v\in R^d$, 令$w_+=prox^H_{\Omega,\alpha}(w-\alpha H^{-1}v), g=\dfrac{(w-w_+)}{\alpha}$,$\Delta=v-\nabla F(w)$, 其中, α为步长, 且满足$0<\alpha\leq 1/L$, 有

$$\Phi(u)\geq\Phi(w_+)+g^TH^T(u-w)+\Delta^T(w_+-u)+\frac{\alpha}{2}\|g\|^2_H. \tag{3.3}$$

基于上面的两个假设和引理3.1, 得到以下收敛性分析.

定理3.1 设$w_*=argmin\Phi(w), 0<\alpha<\dfrac{\lambda_1^2}{16L\lambda_2}$, 在假设1和2的基础上, 有

$$E[\Phi(w_{k+1})-\Phi(w_*)]\leqslant\rho^k(\Phi(w_0)-\Phi(w_*)). \tag{3.4}$$

其中, $\rho=(\dfrac{1}{u\alpha}+\dfrac{8L\alpha\lambda_2}{{\lambda_1}^2})^{1/2}<1$.

证明 设$w=w_k,w_+=w_{k+1},v=v_k,g=g_k,u=w_*,H=H_k^{-1},\Delta_k=v_k-\nabla F(w_k)$, 由引理3.1, 可得

$$\Phi(w_*)\geq\Phi(w_{k+1})+g_k^T(H_k^{-1})^T(w_*-w_k)+\Delta_k^T(w_{k+1}-w_*)+\frac{\alpha}{2}\|g_k\|^2_{H_k}.$$

接下来分析在每次迭代中w_{k+1}与最优值w_*之间的距离关系,

$$\begin{aligned}||w_{k+1}-w_*||^2_{H_k^{-1}}&=||w_k-\alpha g_k-w_*||^2_{H_k^{-1}}\\&=||w_k-w_*||^2_{H_k^{-1}}-2\alpha g_k^T(H_k^{-1})^T(w_k-w_*)+\alpha^2||g_k||^2_{H_k^{-1}}\\&\leq||w_k-w_*||^2_{H_k^{-1}}-2\alpha[\Phi(w_{k+1})-\Phi(w_*)]-2\alpha\Delta_k^T(w_{k+1}-w_*),\end{aligned} \tag{3.5}$$

然后, 寻找$-2\alpha\Delta_k^T(w_{k+1}-w_*)$的一个上界. 定义临近全梯度校正为:

$$\widetilde{w}_{k+1}=prox^{H_k^{-1}}_{\Omega,\alpha}\left(w_k-\alpha H_k^{-1}\nabla F(w_k)\right),$$

则有

$$\begin{aligned}&-2\alpha\Delta_k^T(w_{k+1}-w_*)\\&=-2\alpha\Delta_k^T(w_{k+1}-\widetilde{w}_{k+1})-2\alpha\Delta_k^T(\widetilde{w}_{k+1}-w_*)\\&\leq2\alpha\|\Delta_k\|\|w_{k+1}-\widetilde{w}_{k+1}\|-2\alpha\Delta_k^T(\widetilde{w}_{k+1}-w_*)\\&\leq\frac{2\alpha\lambda_2}{\lambda_1}\|\Delta_k\|\|(w_k-\alpha H_k^{-1}v_k)-(w_k-\alpha H_k^{-1}\nabla(w_k))\|-2\alpha\Delta_k^T(\widetilde{w}_{k+1}-w_*)\\&=\frac{2\alpha^2\lambda_2}{\lambda_1}\|\Delta_k\|\|H_k^{-1}\Delta_k\|-2\alpha\Delta_k^T(\widetilde{w}_{k+1}-w_*)\\&\leq\frac{2\alpha^2\lambda_2}{\lambda_1^2}\|\Delta_k\|^2-2\alpha\Delta_k^T(\widetilde{w}_{k+1}-w_*),\end{aligned}$$

第一个等号运用了柯西不等式, 第二个不等式运用(3.1)式. 再结合(3.5)式, 可得

$$\begin{aligned}&\|w_{k+1}-w_*\|^2_{H_k^{-1}}\\&\leq \|w_k-w_*\|^2_{H_k^{-1}}-2\alpha[\Phi(w_{k+1})-\Phi(w_*)]+\frac{2\alpha^2\lambda_2}{\lambda_1^2}\|\Delta_k\|^2-2\alpha\Delta_k^T(\widetilde{w}_{k+1}-w_*),\end{aligned}$$

注意到, w_{k+1}和w_*关于随机变量S_k是相互独立的, 且$E\Delta_k=0$, 所以

$$E[\Delta_k^T(\widetilde{w}_{k+1}-w_*)]=0.$$

此外, 由(3.2)式可以得到$E\|\Delta_k\|^2$的上界, 可得

$$\begin{aligned}&E\|w_{k+1}-w_*\|^2_{H_k^{-1}}\\&\leq \|w_k-w_*\|^2_{H_k^{-1}}-2\alpha[E\Phi(w_{k+1})-\Phi(w_*)]+\frac{8L\alpha^2\lambda_2}{{\lambda_1}^2}[\Phi(w_k)-\Phi(w_*)+\Phi(w_{k-1})-\Phi(w_*)],\end{aligned}$$

移项并整理, 可得

$$\begin{aligned}&E\|w_{k+1}-w_*\|^2_{H_k^{-1}}+2\alpha[E\Phi(w_{k+1})-\Phi(w_*)]\\&\leq \|w_k-w_*\|^2_{H_k^{-1}}+\frac{8L\alpha^2\lambda_2}{{\lambda_1}^2}[\Phi(w_k)-\Phi(w_*)+\Phi(w_{k-1})-\Phi(w_*)]\\&\leq |w_k-w_*\|^2_{H_k^{-1}}+\frac{16L\alpha^2\lambda_2}{{\lambda_1}^2}[\Phi(w_{k-1})-\Phi(w_*)].\end{aligned}$$

由目标函数Φ强凸性可以得出

$$\|w_k-w_*\|^2_{H_k^{-1}}\leq 2[\Phi(w_k)-\Phi(w_*)]/u,$$

所以

$$\begin{aligned}&2\alpha[E\Phi(w_{k+1})-E\Phi(w_*)]\\&\leq \frac{2}{u}[\Phi(w_k)-\Phi(w_*)]+\frac{16L\alpha^2\lambda_2}{{\lambda_1}^2}[\Phi(w_{k-1})-\Phi(w_*)]\\&\leq (\frac{2}{u}+\frac{16L\alpha^2\lambda_2}{{\lambda_1}^2})[\Phi(w_{k-1})-\Phi(w_*)],\end{aligned}$$

可得

$$\begin{aligned}&E\Phi(w_{k+1})-E\Phi(w_*)\\&\leq (\frac{1}{u\alpha}+\frac{8L\alpha\lambda_2}{{\lambda_1}^2})[\Phi(w_{k-1})-\Phi(w_*)]\\&\leq (\frac{1}{u\alpha}+\frac{8L\alpha\lambda_2}{{\lambda_1}^2})\frac{k}{2}[\Phi(w_0)-\Phi(w_*)],\end{aligned}$$

令

$$\rho = (\frac{1}{u\alpha} + \frac{8L\alpha\lambda_2}{{\lambda_1}^2})\frac{1}{2},$$

当$0 < \alpha < \lambda_1^2/(16\lambda_2 L)$, 有$\rho < 1$, 所以

$$E[\Phi(w_{k+1}) - \Phi(w_*)] \leqslant \rho^k E((\Phi(w_0) - \Phi(w_*))),$$

定理得证.

4. 数值算例

前文给出算法$SNVR$的收敛性分析, 证明了收敛性. 考虑下面的问题, 对于给定的数据集$\{(a_1, b_1), \cdots, (a_N, b_N)\}$, 其中, $a_i \in R^d, b_i \in \{-1, 1\}$, 求下列优化问题:

$$\min_{w \in R^d} \frac{1}{N} \sum_{i=1}^{N} (w^T a_i - y_i)^2 + \lambda_1 \|w\|_1,$$

其中, λ_1 为正则化参数.本文所用的数据集为$heart$, 此数据集中$N = 600, d = 13$. 取$\lambda_1 = 10^{-5}, M = 20, b = 20, eps = 10^{-6}$, 用$matlab2020b$ 编程, 在算法过程中, 又添加最大迭代次数, 得计算结果如图1所示:

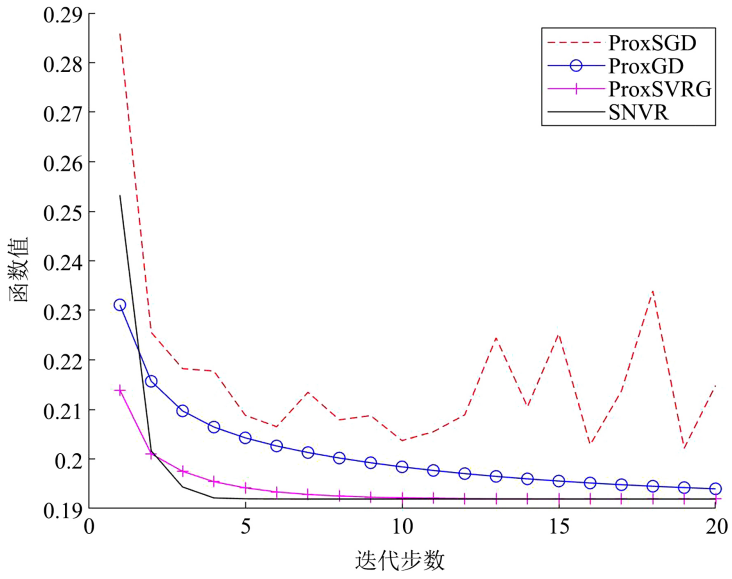

Figure 1. Variation of iteration steps and function values

图 1. 迭代步数和函数值的变化

通过上图可以看出, $SNVR$算法算出的目标函数值不断下降, 在迭代5次之后达到最优, 此时最优值0.1919.

Table 1. Variation of iteration number and function value
表 1. 迭代次数与目标值的变化

	迭代次数	目标函数值
ProxGD	20	0.2148
proxSGD	20	0.1940
ProxSVRG	14	0.1919
SNVR	5	0.1919

通过图 1和表 1可以看出, 与$ProxSGD, ProxGD, ProxSVRG$ 相比, $SNVR$算法还具有收敛性能好, 并且收敛速度更快的优点.

5. 总结

本文对于机器学习中一类复合优化问题, 在目标函数为凸非光滑的条件下, 提出了一种新的随机梯度算法——基于方差缩减的随机近似牛顿法($SNVR$). $SNVR$算法在随机梯度算法的基础上, 结合了方差缩减的方法, 该算法具有计算量小、运算速度快的特点. 本文先给出了$SNVR$算法的迭代步骤, 然后进行了收敛性分析. 最后又进行实例验证, 通过结果分析可以看出目标函数值逐渐下降, 最终收敛于一处. 与$ProxGD$, $SProxGD$, $ProxSVRG$等方法做了对比, 通过对比可以看出, $SNVR$算法能够更快的收敛到最优值.

参考文献

[1] Bottou, L. (2010) Large-Scale Machine Learning with Stochastic Gradient Descent. In: Lechevallier, Y. and Saporta, G., Eds., *Proceedings of COMPSTAT'2010*, Physica-Verlag HD, 177-186. https://doi.org/10.1007/978-3-7908-2604-3_16

[2] Byrd, R.H., Hansen, S.L., Nocedal, J. and Singer, Y. (2016) A Stochastic Quasi-Newton Method for Large-Scale Optimization. *SIAM Journal on Optimization*, **26**, 1008-1031. https://doi.org/10.1137/140954362

[3] Lee, J.D., Sun, Y. and Saunders, M.A. (2012) Proximal Newton-Type Methods for Minimizing Composite Functions. *SIAM Journal on Optimization*, **24**, 1420-1443. https://doi.org/10.1137/130921428

[4] Xiao, L. and Zhang, T. (2014) A Proximal Stochastic Gradient Method with Progressive Variance Reduction. *SIAM Journal on Optimization*, **24**, 2057-2075. https://doi.org/10.1137/140961791

[5] Moritz, P., Nishihara, R. and Jordan, M.I. (2016) A Linearly-Convergent Stochastic L-BFGS Algorithm. *19th International Conference on Artificial Intelligence and Statistics (AISTATS)*, **51**, 249-258.

[6] Defazio, A., Bach, F. and Lacoste-Julien, S. (2014) SAGA: A Fast Incremental Gradient Method with Support for Non-Strongly Convex Composite Objectives. *Proceedings of the 27th International Conference on Neural Information Processing Systems*, **1**, 1646-1654.

[7] Jin, X.B., Zhang, X.Y., Huang, K., *et al.* (2019) Stochastic Conjugate Gradient Algorithm with Variance Reduction. *IEEE Transactions on Neural Networks and Learning Systems*, **30**, 1360-1369. https://doi.org/10.1109/TNNLS.2018.2868835