基于随机谱梯度的在线学习

薛 伟[1] 张文生[1,2] 任俊宏[3]
(南京理工大学计算机科学与工程学院 南京 210094)[1] (中国科学院自动化研究所 北京 100190)[2]
(北京航空航天大学软件学院 北京 100191)[3]

摘 要 考虑一类学习问题,问题的目标函数可表示为大量组函数的平均,并且假设每一个组件函数都是光滑的。在众多机器学习方法中,在线学习操作流程简洁、收敛速度快,而且可以实现模型的自动更新,为大数据的学习提供了有利的工具。针对这类问题,提出了一种基于随机谱梯度下降(Stochastic Spectral Gradient Descent,S^2GD)的在线学习方法。该方法利用Rayleigh商收集目标函数的二阶信息来构造Hessian阵逆的近似。S^2GD方法可以看作是谱梯度方法从确定性优化到随机优化的延伸。算法每次迭代所产生的搜索方向具有下降性,且现有结论表明算法收敛。在LIBSVM数据库上的初步实验表明S^2GD方法是可行的、有效的。

关键词 在线学习,随机优化,凸优化,随机梯度,谱梯度
中图法分类号 TP301 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2016.9.008

Online Learning Based on Stochastic Spectral Gradient

XUE Wei[1] ZHANG Wen-sheng[1,2] REN Jun-hong[3]
(School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, China)[1]
(Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China)[2]
(School of Software, Beihang University, Beijing 100191, China)[3]

Abstract We considered a type of learning problems whose objective functions can be formulated as an average of a large number of component functions, and assumed that each component function is smooth. Among many machine learning methods, online learning provides a favorable tool for big data learning, not only due to its simple operation and fast convergence rate, but also for its ability to automatically update model. To solve such problems, we developed a stochastic spectral gradient descent (S^2GD) method, which employs the Rayleigh quotient to collect second-order information to construct Hessian inverse approximations. S^2GD can be viewed as an approach that extends the spectral gradient method working in deterministic setting to stochastic setting. At each iteration, the generated search direction guarantees descent property. The existing conclusion indicates that the S^2GD method is of convergence. Preliminary experimental results on LIBSVM data sets are reported to demonstrate the feasibility and effectiveness of our approach.

Keywords Online learning, Stochastic optimization, Convex optimization, Stochastic gradient, Spectral gradient

1 引言

目前,机器学习是处理大数据强有力的工具之一,其基本思想是通过数据构建统计模型,进而对数据进行预测和分析。学习的目标可以看成是在联合概率分布 $P(z)$未知、所有可用信息均包含在训练集:

$$\{z_1=(x_1,y_1),\cdots,z_n=(x_n,y_n)\} \tag{1}$$

中的情况下,寻找期望风险函数 $F(w)$的最小值[1],$F(w)$的一般形式可表示成:

$$F(w)=\int f(w;z)\mathrm{d}P(z)\left(\triangleq\mathbb{E}_z f(w;z)\right) \tag{2}$$

其中,w是优化变量(通常称为权向量),$f(w;z)$是损失函数,用来预测错误的程度。因为联合分布 $P(z)$未知,所以不能直接极小化期望风险函数。也正是因为不知道联合概率分布,所以才需要学习。正如文献[2]所言,一方面,根据期望风险最小学习模型,需要用到联合概率分布;另一方面,联合分布又是未知的,所以该学习是一个病态问题。为了在 $P(z)$未知的情况下最小化式(2),常用的处理方法是把$\mathbb{E}_z f(w;z)$替换为经验风险函数:

$$F(w)=\frac{1}{n}\sum_{i=1}^{n}f_i(w)\left(\triangleq\mathbb{E}_n f(w;z)\right) \tag{3}$$

其中,$f_i(w)\triangleq f(w;z_i)=f(w;(x_i,y_i))$,然后用使经验风险最小的 w逼近使期望风险最小的 w_0。式(3)包含许多种学习问题,如基于 l_2 范数逻辑斯特回归的二分类问题。在此类

到稿日期:2015-07-23 返修日期:2015-09-25 本文受国家自然科学基金面上项目(61472423),国家自然科学基金重点项目(U1135005,61432008),江苏省普通高校研究生科研创新计划项目(KYZZ15_0123)资助。

薛 伟(1986—),男,博士生,CCF学生会员,主要研究领域为机器学习,E-mail:mailweixue@163.com;**张文生**(1966—),男,博士,博士生导师,CCF会员,主要研究领域为模式识别与机器学习、大数据知识挖掘、概率图模型、深度学习、精密感知与智能控制、三维数字物理仿真、嵌入式视频图像处理等(通信作者);**任俊宏**(1989—),男,硕士生,主要研究领域为软件工程技术。

问题中，$f_i(w)=\log(1+\exp(-y_i x_i^{\mathrm{T}} w))+\lambda\|w\|^2/2$，其中$(x_i,y_i)$是与二分类问题相关联的训练样本，$\lambda>0$是正则化参数，控制正则项$\|w\|^2$的强度。在这里，正则化技术也可以理解为结构风险最小化。因为当样本容量很小时，经验风险最小化学习的效果未必会很好，可能会产生过拟合的现象。结构风险最小化策略的提出是为了防止过拟合现象的产生。不难看出，结构风险就是在经验风险的基础上加上了表示模型复杂度的正则项。结构风险小的模型往往对训练数据以及未知的数据有较好的预测效果。

随机梯度下降方法(Stochastic Gradient Decent，SGD)是解决学习问题(3)的最简单也是最流行的方法之一[3-5]，其算法流程如算法1所示。

算法1 随机梯度下降算法

输入：w_1，η_1

输出：w_t

1. For t=1,2,…,T Do
2. 从{1,…,n}中随机选择i_t
3. 根据式(4)更新w_t
4. END For

在t时刻，SGD的迭代形式可表示为：

$$w_{t+1}=w_t-\eta_t\underbrace{\nabla f_{i_t}(w_t)}_{sgd_t} \tag{4}$$

其中，η_t是学习步长，样本索引i_t是从集合$\{1,\cdots,n\}$中随机选取的。与批梯度下降(Batch Gradient Decent，BGD)方法相比[1]，由于SGD方法的迭代成本与n无关，因此该方法可以大幅度地减少每步所需的计算量。1951年，Robbins和Monro[6]证明了当η_t满足条件

$$\sum_t\eta_t=\infty,\ \sum_t\eta_t^2<\infty \tag{5}$$

时，式(3)可以收敛到最优解$w^*=\underset{w}{\operatorname{argmin}}F(w)$。一个常用的学习步长为：

$$\eta_t=\frac{\alpha\eta_0}{\alpha+t} \tag{6}$$

其中，η_0是初始步长；α是调优参数，控制步长减少的速度。该步长公式可以同时满足式(5)中的两个条件。关于η_t的其他的选择方法可以查阅文献[7]的第6节。在一定条件下，式(4)定义的SGD方法可以达到次线性的收敛速度，即满足$E[Fw_t]-F(w_*)=O(1/t)$。而具有常数步长的BGD方法可以达到线性收敛。为了提高SGD方法的收敛速度，研究者们提出了一些加速的方法[8-10]。对于光滑的强凸函数而言，这些方法可以达到线性收敛。文献[8]提出了一种随机方差减少的梯度方法(Stochastic Variance Reduced Gradient，SVRG)，该方法不要求存储梯度值。Konečný和Richtárik[9]提出了一种半随机梯度下降方法(Semi-stochastic Gradient Descent，S2GD)，并指出SVRG是S2GD的一种特殊情况。Le Roux等人[10]提出了一种开创性的方法，即随机平均梯度方法(Stochastic Average Gradient，SAG)。SAG方法可以看作是增量聚合梯度方法(Incremental Aggregated Gradient，IAG)[11]的随机形式，其迭代形式如下：

$$w_{t+1}=w_t-\frac{\eta_t}{n}\underbrace{\sum_{i=1}^{n}p_t^i}_{sag_t} \tag{7}$$

其中，$p_t^i=\begin{cases}\nabla f_i(w_t), & \text{如果 } i=i_t\\ p_{t-1}^i, & \text{其他}\end{cases}$。

可以看出，相对于SGD而言，SAG重新使用了先前迭代得到的梯度信息，这有助于加速解的收敛。以上所提方法都是一阶的SGD方法，为提升传统SGD的性能，研究者引入拟牛顿策略，提出了二阶的SGD方法[12-16]。遵循改进传统SGD方法的趋势，提出了一种基于随机谱梯度的学习方法。为了清楚地描述本文工作，下面介绍一些相关知识。

2 相关知识

2.1 牛顿法

考虑如下确定性优化问题

$$\underset{w}{\operatorname{argmin}}\ F(w) \tag{8}$$

牛顿法求解问题(8)的思想是构造函数F在w_t处的二阶近似，并将其极小化。假设F二次连续可微，在w_t附近用二次Taylor展开近似F。

$$\underbrace{F(w_t+s)\approx F(w_t)+\nabla F(w_t)^{\mathrm{T}}s+\frac{1}{2}s^{\mathrm{T}}\nabla^2F(w_t)s}_{Q_t(s)} \tag{9}$$

其中，$s=w-w_t$，$Q_t(s)$是$F(w)$的二次近似。极小化$Q_t(s)$得$w_{t+1}=w_t-[\nabla^2F(w_t)]^{-1}\nabla F(w_t)$，称为牛顿法迭代公式。如果记$H_t=\nabla^2F(w_t)$，那么$d_t=-H_t^{-1}\nabla F(w_t)$称为牛顿方向[2]。牛顿法成功的关键是利用了Hessian阵提供的曲率信息。但是，当H_t不正定时，算法产生的方向不能保证是下降方向。特别地，当H_t奇异时，牛顿方向可能不存在。此外，对于大规模优化问题，计算Hessian阵并求其逆矩阵的工作量大，甚至难以求得，这就导致只能利用目标函数一阶导数信息的方法。拟牛顿法就是这样一类方法，它利用目标函数值和一阶导数信息，构造目标函数的曲率近似，而不需要明显形成Hessian阵，在一定的条件下具有较快的收敛速度。拟牛顿法的基本思想是用H_t的某个近似B_t取代H_t。矩阵B_t应具有如下3个特点：1)在某种意义下有$B_t\approx H_t$，使算法产生的方向(拟牛顿方向)是牛顿方向的近似，以保证算法有较快的收敛速度；2)对所有的t，B_t对称正定，从而使算法产生的方向是函数F在w_t处的下降方向；3)B_t是容易计算的。

2.2 拟牛顿法

设F二次连续可微，在w_t附近的二次近似为

$$F(w)\approx F(w_t)+\nabla F(w_t)^{\mathrm{T}}(w-w_t)+\frac{1}{2}(w-w_t)^{\mathrm{T}}H_t(w-w_t)$$

对上式两边求导，得$\nabla F(w)\approx\nabla F(w_t)+H_t(w-w_t)$。为表达方便，令$w=w_t$，$s_{t-1}=w_t-w_{t-1}$，$y_{t-1}=\nabla F(w_t)-\nabla F(w_{t-1})$，于是有$H_ts_{t-1}\approx y_{t-1}$。从而要求近似矩阵$B_t$对于上面的近似式成立，即要求$B_t$满足方程$B_ts_{t-1}=y_{t-1}$，该方

---

1) 求解问题(2)的批梯度下降方法的迭代形式是：$w_{t+1}=w_t-\eta_t\nabla F(w_t)=w_t-\eta_t\sum_{i=1}^{n}\nabla f_i(w_t)$。由于计算$\nabla F_i(w_t)$需要计算$n$个函数的梯度值，因此当$n$非常大时批梯度下降类方法可能会失去原有的优势。

2) 满足$\nabla F(w_t)^{\mathrm{T}}d_t<0$的$d_t$称为下降方向。如果Hessian阵$H_t$正定，那么$d_t=-H_t^{-1}\nabla F(w_t)$是函数$F$在$w_t$处的下降方向。

程称为拟牛顿方程。常用的拟牛顿校正公式有对称秩一校正(SR1 校正)和对称秩二校正(SR2 校正)[17]，常见的两个校正公式是

$$B_t=(I-\frac{y_{t-1}s_{t-1}^{\mathrm{T}}}{y_{t-1}^{\mathrm{T}}s_{t-1}})B_{t-1}(I-\frac{s_{t-1}y_{t-1}^{\mathrm{T}}}{y_{t-1}^{\mathrm{T}}s_{t-1}})+\frac{y_{t-1}y_{t-1}^{\mathrm{T}}}{y_{t-1}^{\mathrm{T}}s_{t-1}} \tag{10}$$

$$B_t=B_{t-1}+\frac{y_{t-1}y_{t-1}^{\mathrm{T}}}{y_{t-1}^{\mathrm{T}}s_{t-1}}-\frac{B_{t-1}s_{t-1}s_{t-1}^{\mathrm{T}}B_{t-1}}{s_{t-1}^{\mathrm{T}}B_{t-1}s_{t-1}} \tag{11}$$

其中，式(10)称为 DFP(Davidon-Fletcher-Powell)校正，式(11)为 BFGS(Broyden-Fletcher-Goldfarb-Shanno)校正，两者都属于 SR2 校正的范畴。DFP 公式与 BFGS 公式的加权线性组合可构成一类新的校正公式：

$$B_t=(1-\phi)B_t^{\mathrm{DFP}}+\phi B_t^{\mathrm{BFGS}}$$

如果 $\phi=s_{t-1}^{\mathrm{T}}y_{t-1}/(s_{t-1}H_{t-1}y_{t-1})^{\mathrm{T}}y_{t-1}$，那么该线性组合为 SR1 校正。然而，当矩阵 B_t 稠密或者问题的维度较大时，这种标准的拟牛顿方法可能会失效。换句话说，迭代过程中主要的计算量在于每一次迭代都要存储一个较大的矩阵。因此，需要一个无需矩阵存储的方法。为此，可以用一个对角矩阵来代替 B_t，即 $B_t=\beta_t I$，其中 $\beta_t>0$，I 是单位矩阵。于是，拟牛顿条件可写成 $\beta_t I s_{t-1}=y_{t-1}$，两端同时乘以 s_{t-1}^{T}，有

$$\beta_t^1=\frac{s_{t-1}^{\mathrm{T}}y_{t-1}}{\|s_{t-1}\|^2} \tag{12}$$

类似地，在方程两端同时乘以 y_{t-1}^{T}，有

$$\beta_t^2=\frac{\|y_{t-1}\|^2}{s_{t-1}^{\mathrm{T}}y_{t-1}} \tag{13}$$

如果 $s_{t-1}^{\mathrm{T}}y_{t-1}>0$，那么矩阵 $B_t=\beta_t I$ 是正定的，保证了当前方向 $d_t=-B_t^{-1}\nabla F(w_t)=-\beta_t^{-1}\nabla F(w_t)$ 是函数 F 在 w_t 处的下降方向。式(12)和式(13)最早是由 Barzilai 和 Borwein 提出[18]，其基本思想是利用迭代当前点以及前一点的信息来确定步长因子 β_t。注意到 $y_{t-1}=[\int_0^1\nabla^2F(w_{t-1}+\theta s_{t-1})\mathrm{d}\theta]s_{t-1}$，故式(12)可以看成是如下形式的 Rayleigh 商。

$$\frac{s_{t-1}^{\mathrm{T}}[\int_0^1\nabla^2F(w_{t-1}+\theta s_{t-1})\mathrm{d}\theta]s_{t-1}}{\|s_{t-1}\|^2}$$

而式(13)则是如下形式的 Rayleigh 商。

$$\frac{s_{t-1}^{\mathrm{T}}[(\int_0^1\nabla^2F(w_{t-1}+\theta s_{t-1})\mathrm{d}\theta)^{\mathrm{T}}\int_0^1\nabla^2F(w_{t-1}+\theta s_{t-1})\mathrm{d}\theta]s_{t-1}}{s_{t-1}^{\mathrm{T}}[\int_0^1\nabla^2F(w_{t-1}+\theta s_{t-1})\mathrm{d}\theta]s_{t-1}}$$

因此，该方法也称为谱梯度方法。由于具有简单、有效、存储需求少等特点，该方法已经在大规模优化领域得到了很多应用[19-22]。下面给出随机谱梯度下降方法。

3 随机谱梯度下降方法

在 t 时刻，求解问题(3)转化为求解如下优化问题：

$$\operatorname{argmin}_w f_t(w) \tag{14}$$

其中，$f_t(w)$ 表示 $f(w;(x_t,y_t))$，样本 (x_t,y_t) 是从训练集中随机选取的。根据 2.2 节的分析，将变量之差和梯度之差定义如下：

$$\Delta w_t=w_t-w_{t-1} \tag{15}$$

$$\Delta g_t=\nabla f_t(w_t)-\nabla f_{t-1}(w_{t-1}) \tag{16}$$

受文献[10]中平均梯度思想的启发，给出一个有限平均权向量的策略，即通过先前的迭代来计算变量之差 ∇w_t。具体而言，保持 ∇g_t 的形式不变，将 ∇w_t 重新定义如下：

$$\Delta w_t=\bar{w}_t-\bar{w}_{t-1} \tag{17}$$

其中，$\bar{w}_t=\frac{(\sum_{a_t}^{t}w_{a_t})}{M}$，$\alpha_t=t-M+1$，$M$ 为记忆参数，且 $1\leqslant M\leqslant t$。如果 $M=1$，式(17)将退化为式(15)。从而，S^{2}GD 的迭代形式可以表示为：

$$w_{t+1}=w_t-B_t\hat{s}_t \tag{18}$$

其中，$B_t=\beta_t I$，$\beta_t=\frac{\|\Delta w_t\|^2}{\Delta w_t^{\mathrm{T}}\Delta g_t}$(或 $\frac{\Delta w_t^{\mathrm{T}}\Delta g_t}{\|\Delta g_t\|^2}$)，$\hat{s}_t$ 表示在 t 时刻得到的随机梯度，可以由式(4)中的 $\overline{sgd_t}$ 或者式(7)中的 $\overline{sag_t}$ 计算得到。随机谱梯度下降方法的流程如算法 2 所示。

算法 2 S^{2}GD

输入：初始步长 η_0，参数 τ，M，索引 $i_t\sim U[1,\cdots,n]$，步长计算函数 S($\Delta w_t,\Delta g_t$)。置 t←1

1. $B_t\leftarrow\frac{\tau\beta_0}{\tau+t}I$
2. $g_t\leftarrow\overline{sgd_t}$
3. For t=2,…,T Do
4. $w\leftarrow w-B_t*g_t$
5. $g_t\leftarrow\overline{sgd_t}$
6. 由式(17)和式(16)计算 Δw_t 和 Δg_t
7. If $\Delta w_t^{\mathrm{T}}\Delta g_t>0$ Then
8. $B_t\leftarrow S(\Delta w_t,\Delta g_t)$
9. Else
10. $B_t\leftarrow\frac{\tau\beta_0}{\tau+t}I$
11. End If
12. End For

步长计算函数 $S(\Delta w_t,\Delta g_t)$

输入：$\Delta w_t,\Delta g_t$，阈值参数 $0<\delta<1$，$\beta_{\max}>\beta_{\min}>0$

Function 1

1. $\beta_t\leftarrow\frac{\|\Delta w_t\|^2}{\Delta w_t^{\mathrm{T}}\Delta g_t}$

Function 2

2. $\beta_t\leftarrow\frac{\Delta w_t^{\mathrm{T}}\Delta g_t}{\|\Delta g_t\|^2}$

Function 3

3. $\beta_1=\frac{\|\Delta w_t\|^2}{\Delta w_t^{\mathrm{T}}\Delta g_t}$，$\beta_2=\frac{\Delta w_t^{\mathrm{T}}\Delta g_t}{\|\Delta g_t\|^2}$
4. If mod(t,2)≠0 or $\frac{\Delta w_t^{\mathrm{T}}\Delta g_t}{\|\Delta w_t\|\,\|\Delta g_t\|}>\delta$ Then
5. $\beta_t\leftarrow\beta_1$
6. Else
7. $\beta_t\leftarrow\beta_2$
8. End If
9. $\beta_t\leftarrow\min(\beta_{\max},\max(\beta_t,\beta_{\min}))$
10. $\beta_t\leftarrow\frac{\tau\beta_t}{\tau+t}$
11. $B_t\leftarrow\beta_t I$

注 1：如果 $\Delta w_t^{\mathrm{T}}\Delta g_t>0$，那么算法产生的方向 $-B_t\hat{s}_t$ 是下降方向。然而，并不能保证这个条件一直成立，克服该缺点的一个方法就是限制 β_t 的最终取值 $\min\{\beta_{\max},\max\{\beta,\beta_{\min}\}\}$，其中 $\beta_{\max}>0$ 足够大，而 $\beta_{\min}>0$ 足够小。该策略可以保证 $-B_t\hat{s}_t$ 在每一步都是下降方向。

注 2：为了保证算法 S^{2}GD 的收敛性，对 β_t 进行了限制，使其满足条件(5)。

注 3：S^{2}GD 采用的是最基本的谱梯度公式，为了增强算法的性能，可以采用一些改进的谱梯度公式[19,20]。

4 数值实验

本节通过实验来验证所提方法的有效性，对比方法有SGD，SAG，S²GD+F1，S²GD+F2 和 S²GD+F3[1)]。实验运行环境为 Windows 7&Matlab R2013a。

4.1 问题、数据集描述及算法参数设置

考虑二分类问题，n 表示样本的个数，d 表示特征的个数，目标函数取为 l_2-logistic 回归，于是问题(3)可写成

$$\begin{aligned}&\underset{w\in R^d}{\operatorname{argmin}}\ F(w):=\frac{1}{n}\sum_{i=1}^{n}f_i(w)\\&f_i(w)=\log(1+\exp(-y_ix_i^Tw))+\frac{\lambda}{2}\|w\|^2\end{aligned}\tag{19}$$

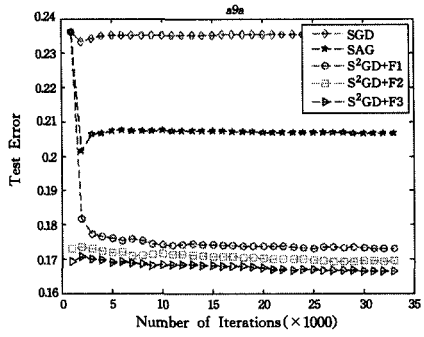
其中，$x_i\in R^d$，$y_i\in\{-1,+1\}$是对应于 x_i 的标签，$\lambda>0$ 是正则化参数。实验选取 LIBSVM 数据库中的 2 组数据进行分析，数据的具体描述如表 1 所列。S²GD 算法中的参数设置为：$\beta_0=1$，$\tau=10^3$，$M=2$，$\beta_{\min}=10^{-6}$，$\beta_{\max}=10^8$，$\delta=0.85$。

表 1 实验数据集描述

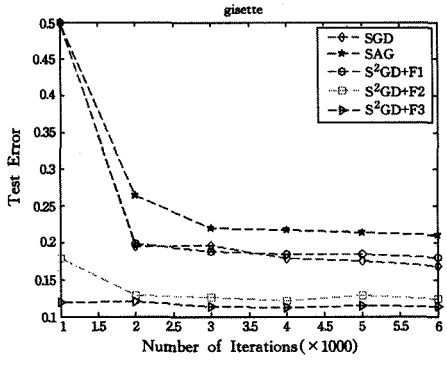
数据集 Data Sets	训练样本数 # of Training Samples	特征数 # of Features	测试样本数 # of Testing Samples
a9a	32561	123	16281
gisette	6000	5000	1000

4.2 实验方法与结果

在实验过程中，我们对每个数据集的样本采取随机抽取的方式，并给出不同正则化参数取值下的实验结果。



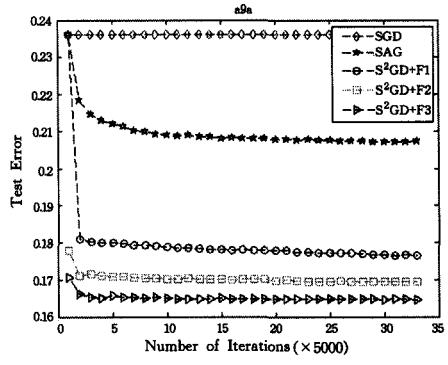
(a)



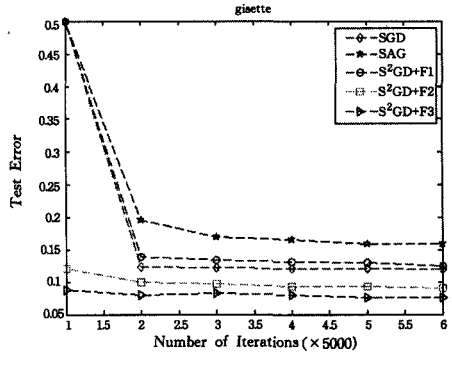
(b)

图 1 $\lambda=10^{-5}$时算法在数据集上遍历 1 次各方法的测试误差趋势

图 1 为 $\lambda=10^{-5}$时 5 种方法在两个数据集上遍历一次得到的测试错误率比较图，横坐标表示迭代次数，纵坐标表示测试误差。图 2 为 $\lambda=10^{-5}$时 5 种方法在两个数据集上遍历 5 次得到的测试错误率比较图。从两幅图中不难看出，5 种方法的收敛速度(这里指测试误差趋于平稳的速度)都很快。在 a9a 数据集上，S²GD+F1，S²GD+F2 和 S²GD+F3 方法的正确率均高于其余两种方法，其中 S²GD+F3 表现最好；而在 gisette 数据集上，测试错误率最低的也是 S²GD+F3 方法。



(a)



(b)

图 2 $\lambda=10^{-5}$时算法在数据集上遍历 5 次各方法的测试误差趋势

表 2 给出了 $\lambda=10^{-1}$时算法在两个数据集上运行 1 次得到的结果，包括算法运行所需的时间(Time)、算法终止时的目标函数值(Fun.)和测试错误率(Err.)。表 3 列出了当 $\lambda=10^{-3}$时算法在两个数据集上运行 5 次的结果。不难看出，在数据集 a9a 上 S²GD 的 3 个方法的终止目标函数值和测试错误率均优于 SGD 和 SAG 方法。而在 gisette 数据集上的测试结果表明 S²GD+F2 和 S²GD+F3 的测试错误率也优于 SGD 和 SAG。此外，我们发现算法在数据集上遍历 1 次和 5 次所得到的结果相差不大。因此，我们猜测对于较大规模的样本，进行 1 次或少数几次随机遍历即可。

表 2 $\lambda=10^{-1}$时算法在数据集上遍历 1 次的结果

Data Sets	S²GD+F1			S²GD+F2			S²GD+F3			SGD			SGA		
	Time	Fun.	Err.	Time	Fun.	Err.	Time	Fun.	Err.	Time	Fun.	Err.	Time	Fun.	Err.
a9a	0.272	0.489	0.213	0.266	0.489	0.205	0.29	0.489	0.205	0.138	0.539	0.236	0.171	0.497	0.227
gisette	1.849	3.71	0.108	1.868	1.241	0.082	1.854	0.569	0.072	0.892	0.51	0.117	1.229	0.646	0.162

1) S²GD+F1，S²GD+F2 和 S²GD+F3 分别表示步长函数 $S(\Delta w_t,\Delta g_t)$中取 Function 1，Function 2 和 Function 3。

表 3 $\lambda=10^{-3}$时算法在数据集上遍历 5 次的结果

Data Sets	S^2GD+F1			S^2GD+F2			S^2GD+F3			SGD			SGA		
	Time	Fun.	Err.	Time	Fun.	Err.	Time	Fun.	Err.	Time	Fun.	Err.	Time	Fun.	Err.
a9a	1.287	0.426	0.193	1.282	0.4	0.181	1.274	0.389	0.175	0.69	0.507	0.236	0.881	0.449	0.225
gisette	9.501	7.506	0.229	9.374	3.974	0.147	9.331	2.928	0.115	4.523	0.47	0.16	5.961	0.649	0.228

结束语 本文给出了一个基于随机谱梯度的在线学习框架，从优化的角度设计了 3 种学习算法，其结构简单且易于实施，并通过实验验证了算法的有效性。从算法的设计角度看，所提方法可用于非光滑优化，如基于 1 范数的学习问题。

参 考 文 献

[1] Vapnik V. The Nature of Statistical Learning Theory[M]. New York: Springer, 2000

[2] 李航. 统计学习方法[M]. 北京：清华大学出版社，2012

[3] Bottou L. Stochastic Learning[M]// Advanced Lectures on Machine Learning. Springer Berlin Heidelberg, 2004: 146-168

[4] Bottou L. Large-scale machine learning with stochastic gradient descent[C]// Proceedings of Computational Statistics 2010. Physica-Verlag HD, 2010: 177-186

[5] Zhang Tong. Solving large scale linear prediction problems using stochastic gradient descent algorithms[C]// Proceedings of the 21st International Conference on Machine Learning. 2004: 919-926

[6] Robbins H, Monro S. A stochastic approximation method[J]. The Annals of Mathematical Statistics, 1951, 22(3): 400-407

[7] Powell W B. Approximate Dynamic Programming: Solving the Curses of Dimensionality[M]. John Wiley & Sons, 2007

[8] Johnson R, Zhang Tong. Accelerating stochastic gradient descent using predictive variance reduction[C]// Advances in Neural Information Processing Systems. 2013: 315-323

[9] Konecn ỳ J, Richtárik P. Semi-stochastic gradient descent methods[OL]. http://arxiv.org/pdf/1312.1666v2.pdf

[10] Le Roux N, Schmidt M, Bach F. A stochastic gradient method with an exponential convergence rate for finite training [C]// Advances in Neural Information Processing Systems. 2012: 2663-2671

[11] Blatt D, Hero A O, Gauchman H. A convergent incremental gradient method with a constant step size [J]. SIAM Journal on Optimization, 2007, 18(1): 29-51

[12] Bordes A, Bottou L, Gallinari P. SGD-QN: careful quasi-Newton stochastic gradient descent[J]. Journal of Machine Learning Research, 2009, 10: 1737-1754

[13] Mokhtari A, Ribeiro A. A dual stochastic DFP algorithm for optimal resource allocation in wireless systems[C]// IEEE 14th Workshop on Signal Processing Advances in Wireless Communications. 2013: 21-25

[14] Mokhtari A. Ribeiro A. Regularized stochastic BFGS algorithm [C]// IEEE Global Conference on Signal and Information Processing. 2013: 1109-1112

[15] Schraudolph N, Yu Jin, Günter S. A stochastic quasi-newton method for online convex optimization [C]// International Conference on Artificial Intelligence and Statistics. 2007: 436-443

[16] Sopyła K, Drozda P. Stochastic gradient descent with Barzilai-Borwein update step for SVM purposes[J]. Information Sciences, 2015, 316(20): 218-233

[17] Sun Wen-yu, Yuan Ya-xiang. Optimization Theory and Methods: Nonlinear Programming[M]. Springer Science & Business Media, 2006

[18] Barzilai J, Borwein J M. Two-point step size gradient methods [J]. IMA Journal of Numerical Analysis, 1988, 8(1): 141-148

[19] Biglari F, Solimanpur M. Scaling on the spectral gradient method [J]. Journal of Optimization Theory and Applications, 2013, 158 (2): 626-635

[20] Farid M, Leong W J, Hassan M A. A new two-step gradient-type method for large-scale unconstrained optimization[J]. Computers and Mathematics with Applications, 2010, 59(10): 3301-3307

[21] Wright S J, Nowak R D, Figueiredo M A T. Sparse reconstruction by separable approximation[J]. IEEE Transactions on Signal Processing, 2009, 57(7): 2479-2493

[22] Yu Gao-hang, Guan Lu-tai, Chen Wu-fan. Spectral conjugate gradient methods with sufficient descent property for large-scale unconstrained optimization[J]. Optimization Methods and Software, 2008, 23(2): 275-293

(上接第 46 页)

[34] Uhlmann J K. Satisfying general proximity / similarity queries with metric trees[J]. Information Processing Letters, 1991, 40 (4): 175-179

[35] Krizhevsky A, Hinton G. Learning multiple layers of features from tiny images[D]. University of Toronto, 2009

[36] Oliva A, Torralba A. Modeling the shape of the scene: A holistic representation of the spatial envelope[J]. International Journal of Computer Vision, 2001, 42(3): 145-175

[37] Wang J, Yang J, Yu K, et al. Locality-constrained linear coding for image classification[C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2010: 3360-3367

[38] Lowe D G. Distinctive image features from scale-invariant keypoints[J]. International Journal of Computer Vision, 2004, 60 (2): 91-110

[39] Csurka G, Dance C, Fan L, et al. Visual categorization with bags of keypoints[J]. Proceedings of the European Conference on Computer Vision, 2004, 1(1-22): 1-2

[40] Jia Y, Shelhamer E, Donahue J, et al. Caffe: Convolutional architecture for fast feature embedding[OL]. arXiv preprint, arXiv: 1408.5093, 2014

[41] Indyk P, Motwani R. Approximate nearest neighbors: towards removing the curse of dimensionality[C]// Proceedings of the Symposium on Theory of Computing. 1998: 604-613

[42] Johnson W, Lindenstrauss J. Extensions of Lipschitz mappings into a Hilbert space[J]. Contemporary Mathematics, 1984, 26: 189-206