Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

Stochastic extra-gradient based alternating direction methods for graph-guided regularized minimization[*]

Qiang LAN[1,2], Lin-bo QIAO[‡1,2], Yi-jie WANG[1,2]

[1]*College of Computer, National University of Defense Technology, Changsha 410073, China*

[2]*National Laboratory for Parallel and Distributed Processing, National University of Defense Technology, Changsha 410073, China*

E-mail: lanqiang_nudt@163.com; qiao.linbo@nudt.edu.cn; wwyyjj1971@vip.sina.com

Received Dec. 3, 2016; Revision accepted Apr. 17, 2017; Crosschecked June 7, 2018

Abstract: In this study, we propose and compare stochastic variants of the extra-gradient alternating direction method, named the stochastic extra-gradient alternating direction method with Lagrangian function (SEGL) and the stochastic extra-gradient alternating direction method with augmented Lagrangian function (SEGAL), to minimize the graph-guided optimization problems, which are composited with two convex objective functions in large scale. A number of important applications in machine learning follow the graph-guided optimization formulation, such as linear regression, logistic regression, Lasso, structured extensions of Lasso, and structured regularized logistic regression. We conduct experiments on fused logistic regression and graph-guided regularized regression. Experimental results on several genres of datasets demonstrate that the proposed algorithm outperforms other competing algorithms, and SEGAL has better performance than SEGL in practical use.

Key words: Stochastic optimization; Graph-guided minimization; Extra-gradient method; Fused logistic regression; Graph-guided regularized logistic regression

https://doi.org/10.1631/FITEE.1601771 **CLC number:** TP311

1 Introduction

There are many problems arising from statistics, machine learning, and genetic engineering. They can be formulated as minimization problems. One example is the logistic regression model. In the training phase of logistic regression, it can be formulated as the minimization of logistic loss function $l(\boldsymbol{x})$:

$$\min_{\boldsymbol{x}} \ \frac{1}{n}\sum_{i=1}^{n}\log(1+\exp(-b_i(\boldsymbol{a}_i^{\mathrm{T}}\boldsymbol{x}+c))), \quad (1)$$

where c is a constant, n is the number of data samples, $(\boldsymbol{a}_i, b_i)$ is the i^{th} pair of data samples, in which $\boldsymbol{a}_i \in \mathbb{R}^d$ is the feature and $b_i = \{-1, +1\}$ is the corresponding label, and d is the dimension of the features. In the testing phase of logistic regression, we calculate the conditional probability $P(b|\boldsymbol{a})$ of label b conditioned on sample $\boldsymbol{a}$ given $\boldsymbol{x} \in \mathbb{R}^d$, which is the weight vector obtained in the training process, where conditional probability $P(b|\boldsymbol{a})$ is defined as

$$P(b|\boldsymbol{a}) = \frac{1}{1+\exp(-b(\boldsymbol{a}^{\mathrm{T}}\boldsymbol{x}+c))}. \quad (2)$$

Given feature $\boldsymbol{a}$, if b has the same sign as $\boldsymbol{a}^{\mathrm{T}}\boldsymbol{x}+c$, then $P(b|\boldsymbol{a}) \geq 0.5$; otherwise, $P(b|\boldsymbol{a}) < 0.5$.

Qiao et al. (2017) reported that minimizing an objective function with a regularized term would improve the efficacy of the original model. For example, the sparse logistic regression is $\min_{\boldsymbol{x}} \ \ell(\boldsymbol{x}) + \gamma\|\boldsymbol{x}\|_1 + \lambda\sum_{j=2}^{n}|x_j - x_{j-1}|$ and the graph-guided regularized logistic regression is $\min_{\boldsymbol{x}} \ \ell(\boldsymbol{x}) + \frac{\gamma}{2}\|\boldsymbol{x}\|_2^2 +$

[‡] Corresponding author

[*] Project supported by the National Natural Science Foundation of China (No. 61303264) and the National Key Research and Development Program of China (No. 2016YFB1000401)

ORCID: Lin-bo QIAO, http://orcid.org/0000-0002-8285-2738

$\lambda \sum_{i=1}^{n} | \sum_{j=1}^{m} w_{ij} x_j |$, where $\ell(\boldsymbol{x})$ denotes the average logistic loss function and the ℓ_1 norm $\|\boldsymbol{x}\|_1$ is imposed to promote the sparsity of weight vector $\boldsymbol{x}$. Note that the last term of these models can be expressed as $r(\boldsymbol{F}\boldsymbol{x})$, where $r : \mathbb{R}^l \to \mathbb{R}$ is a convex regularization function, and $\boldsymbol{F} \in \mathbb{R}^{l\times d}$ is a penalty matrix (not necessarily diagonal) specifying the desired structured pattern in $\boldsymbol{x}$.

The optimization problem can be solved by the alternating direction method of multipliers (ADMM) through introducing an auxiliary variable and reformulating it as the following linearly constrained minimization problem:

$$\begin{aligned} \min_{\boldsymbol{x},\boldsymbol{y}} \quad & f(\boldsymbol{x}) + g(\boldsymbol{y}) \\ \text{s.t.} \quad & \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{y} = \boldsymbol{b}, \end{aligned} \tag{3}$$

where $\boldsymbol{x} \in \mathbb{R}^n$, $\boldsymbol{y} \in \mathbb{R}^m$, $\boldsymbol{A} \in \mathbb{R}^{p\times n}$, $\boldsymbol{B} \in \mathbb{R}^{p\times m}$, and $\boldsymbol{b} \in \mathbb{R}^p$. The above formulation covers quite a few popular models arising from statistics and machine learning under the framework of structural risk minimization (Hastie et al., 2001), such as the linear regression obtained by setting $f(\boldsymbol{x}, \xi_i) = \frac{1}{2}\left\|\boldsymbol{a}_i^{\mathrm{T}}\boldsymbol{x} - b_i\right\|^2$ and $g(\boldsymbol{y}) = 0$, the linear support vector machine (SVM) (Cortes and Vapnik, 1995) obtained by setting $f(\boldsymbol{x}, \xi_i) = \max\{0, 1 - b_i\boldsymbol{a}_i^{\mathrm{T}}\boldsymbol{x}\}$ and $g(\boldsymbol{y}) = \frac{\lambda}{2}\|\boldsymbol{y}\|_2^2$, the Lasso (Tibshirani, 1996) obtained by setting $f(\boldsymbol{x}, \xi_i) = \frac{1}{2}\left\|\boldsymbol{a}_i^{\mathrm{T}}\boldsymbol{x} - b_i\right\|^2$ and $g(\boldsymbol{y}) = \lambda\|\boldsymbol{y}\|_1$, and various structured sparse learning (Qiao et al., 2017), where $\lambda > 0$ is a parameter.

The classical ADMM (Boyd et al., 2011) can be used to solve problem (3) with the assumption that the proximal mappings for f and g, i.e., prox_f and prox_g, are easy to obtain. However, in many practical applications, computing prox_f is not easy, such as the sparse logistic regression and fused logistic regression. The closed-form solution may not exist due to linear composition. Moreover, the computation of full gradient makes these batch methods be used only to moderate-scale problems.

To handle large-scale problems, researchers have proposed several stochastic ADMM algorithms (Ouyang et al., 2013; Suzuki, 2013; Azadi and Sra, 2014; Zhao et al., 2015) by combining the stochastic optimization technique and ADMM type methods. In these stochastic algorithms, a noisy sub-gradient is computed instead of the full gradient, so that these stochastic algorithms have the ability to handle large-scale datasets. However, these studies made the assumption that $f(\cdot)$ is differentiable and ∇f is Lipschitz continuous, but this assumption cannot be guaranteed in many real-world applications.

In this study, we propose two stochastic variants of the extra-gradient alternating direction method, named stochastic extra-gradient alternating direction method with Lagrangian function (SEGL) and augmented Lagrangian function (SEGAL), which combine the advantages of the extra-gradient type alternating direction method (EGADM) and stochastic optimization methods, to address the above difficulties. SEGL and SEGAL are efficient and robust to solve linear constrained optimization problems in large scale. We conduct experiments on the fused logistic regression and graph-guided regularized regression. Experimental results on several genres of datasets demonstrate that the proposed algorithm outperforms other competing methods and SEGAL has better performance than SEGL in practical use.

2 Related work

The classical ADMM (Boyd et al., 2011) has been known efficient to solve problem (3) with linear constraints in practice, assuming that the proximal mappings for f and g, i.e., prox_f and prox_g, are easy to obtain. However, in many practical applications, the closed-form solution of prox_f is not available, or computing prox_f is not easy, such as the sparse logistic regression (Tibshirani, 1996), fused logistic regression (Tibshirani et al., 2005), and graph-guided regularized minimization (Hastie et al., 2001). Several variants of the inexact version of ADMM addressed the computational difficulty in proximal mapping and experimental results showed that the EGADM was very efficient and stable for moderate-scale problems.

To be specific, Yang and Yuan (2013) approximated the subproblem of the ADMM by linearizing the quadratic term of its objective function. Lin et al. (2015) minimized the augmented Lagrangian function plus a proximal term in the $\boldsymbol{x}$-subproblem:

$$\begin{aligned} \boldsymbol{x}^{k+1} = \operatorname*{argmin}_{\boldsymbol{x}\in\mathcal{X}} \ & f(\boldsymbol{x}) - \langle \boldsymbol{\lambda}^k, \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{y} - \boldsymbol{b} \rangle \\ & + \frac{\gamma}{2}\|\boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{y} - \boldsymbol{b}\|_{\mathrm{F}}^2 + \frac{1}{2}\|\boldsymbol{x} - \boldsymbol{x}^k\|_{\boldsymbol{H}}^2, \end{aligned} \tag{4}$$

where $\|\cdot\|_{\mathrm{F}}$ is the Frobenius norm and $\boldsymbol{H}$ is a matrix, while they minimized the Lagrangian function in other subproblems. The full gradient was adopted to solve the subproblems. Moreover, computing the full gradient made these batch methods suffer severely from poor scalability.

To address the inability to solve large-scale optimization problems, several stochastic ADMM algorithms (Ouyang et al., 2013; Suzuki, 2013; Azadi and Sra, 2014; Zhao et al., 2015) have been proposed. In these stochastic algorithms, a noisy sub-gradient was computed instead of the full gradient. In computing the noisy sub-gradient, only one sample or a mini-batch of samples were involved, so that these stochastic algorithms had the ability to handle large-scale datasets. However, these stochastic algorithms made the assumption that $f(\cdot)$ was differentiable and ∇f was Lipschitz continuous, but it was not true in many real-world applications. Moreover, drawing a noisy sub-gradient may lead to unstable numerical performance, especially on large-scale problems.

Lin et al. (2018) proposed the stochastic primal-dual proximal extra-gradient descent method (SPDPEG), to solve a class of compositely regularized minimization problems with special regularizations. Specifically, $\boldsymbol{A}$ is assumed to be diagonal in problem (3). However, the assumption was quite strong and did not hold for many compositely regularized problems. This motivates us to consider problem (3) and to develop SEGL and SEGAL.

We propose the SEGL and SEGAL algorithms, which combine the efficiency and robustness of EGADM and the ability of stochastic optimization to solve linearly constrained problem (3) in large scale.

3 Stochastic variants of extra-gradient type alternating direction method

In this section, we introduce the details of stochastic variants of extra-gradient based alternating direction method with Lagrangian function and augmented Lagrangian function, with uniformly or non-uniformly averaged iterations to solve linearly constrained problem (3) in large scale.

3.1 Stochastic extra-gradient alternating direction method with Lagrangian function (SEGL)

Algorithm 1 presents the SEGL and we present important issues as follows:

The first subproblem in Algorithm 1 minimizes the augmented Lagrangian function $\mathcal{L}_\gamma(\cdot)$ with a proximal term $\frac{1}{2}\|\boldsymbol{y}-\boldsymbol{y}^k\|_{\boldsymbol{H}}^2$ with respect to $\boldsymbol{y}$, i.e.,

$$\boldsymbol{y}^{k+1} = \underset{\boldsymbol{y}\in\mathcal{Y}}{\operatorname{argmin}}\ \mathcal{L}_\gamma(\boldsymbol{x}^k, \boldsymbol{y}; \boldsymbol{\lambda}^k) + \frac{1}{2}\|\boldsymbol{y}-\boldsymbol{y}^k\|_{\boldsymbol{H}}^2, \tag{5}$$

where the augmented Lagrangian function for problem (3) is defined as

$$\begin{aligned}\mathcal{L}_\gamma(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{\lambda}) = f(\boldsymbol{x}) - \langle \boldsymbol{\lambda}, \boldsymbol{A}\boldsymbol{x}+\boldsymbol{B}\boldsymbol{y}-\boldsymbol{b}\rangle \\ + g(\boldsymbol{y}) + \frac{\gamma}{2}\|\boldsymbol{A}\boldsymbol{x}+\boldsymbol{B}\boldsymbol{y}-\boldsymbol{b}\|_{\mathrm{F}}^2,\end{aligned} \tag{6}$$

where $\boldsymbol{\lambda} \in \mathbb{R}^p$ is the dual variable corresponding to the linear constraint.

The proximal term $\frac{1}{2}\|\boldsymbol{y}-\boldsymbol{y}^k\|_{\boldsymbol{H}}^2$ is imposed to cancel the effect of matrix $\boldsymbol{B}$ in the quadratic penalty term. One typical choice of $\boldsymbol{H}$ is $\boldsymbol{H}=\boldsymbol{0}$ when $\boldsymbol{B}$ is an identity, or $\boldsymbol{H}=\tau\boldsymbol{I}-\gamma\boldsymbol{B}^{\mathrm{T}}\boldsymbol{B}$ when $\boldsymbol{B}$ is not an identity, where $\tau > \gamma\,\delta_{\max}(\boldsymbol{B}^{\mathrm{T}}\boldsymbol{B})$. Then Eq. (5) computes the proximal mapping for $f(\cdot)$ and is known to be efficiently computable.

The SEGL algorithm shares some common features with the EGADM algorithm (Lin et al., 2015). In fact, the SEGL algorithm takes a stochastic gradient estimation, i.e.,

$$\bar{\boldsymbol{x}}^{k+1} = \boldsymbol{x}^k - c^{k+1}(\nabla f(\boldsymbol{x}^k, \xi_1^{k+1}) - \boldsymbol{A}^{\mathrm{T}}\boldsymbol{\lambda}^k), \tag{7}$$

$$\bar{\boldsymbol{\lambda}}^{k+1} = \boldsymbol{\lambda}^k - \gamma(\boldsymbol{A}\boldsymbol{x}^k + \boldsymbol{B}\boldsymbol{y}^{k+1} - \boldsymbol{b}), \tag{8}$$

$$\boldsymbol{x}^{k+1} = \boldsymbol{x}^k - c^{k+1}(\nabla f(\bar{\boldsymbol{x}}^{k+1}, \xi_2^{k+1}) - \boldsymbol{A}^{\mathrm{T}}\bar{\boldsymbol{\lambda}}^{k+1}), \tag{9}$$

$$\boldsymbol{\lambda}^{k+1} = \boldsymbol{\lambda}^k - \gamma(\boldsymbol{A}\bar{\boldsymbol{x}}^{k+1} + \boldsymbol{B}\boldsymbol{y}^{k+1} - \boldsymbol{b}). \tag{10}$$

3.2 Stochastic extra-gradient alternating direction method with augmented Lagrangian function (SEGAL)

Algorithm 2 presents the SEGAL and we present important issues as follows:

Algorithm 1 Stochastic extra-gradient alternating direction method with Lagrangian function (SEGL)

```
Input: $\boldsymbol{x}^0$, $\boldsymbol{y}^0$, and $\boldsymbol{\lambda}^0$.
Output: $\tilde{\boldsymbol{y}}^t = \sum_{k=0}^{t} \alpha^{k+1}\boldsymbol{y}^{k+1}$, $\tilde{\boldsymbol{x}}^t = \sum_{k=0}^{t} \alpha^{k+1}\bar{\boldsymbol{x}}^{k+1}$, and
        $\tilde{\boldsymbol{\lambda}}^t = \sum_{k=0}^{t} \alpha^{k+1}\bar{\boldsymbol{\lambda}}^{k+1}$.
for $k = 0, 1, 2, \ldots$ do
    Choose two data samples $\xi_1^{k+1}$ and $\xi_2^{k+1}$ randomly
    Update $\boldsymbol{y}^{k+1}$ according to Eq. (5)
    Update $\bar{\boldsymbol{x}}^{k+1}$ according to Eq. (7)
    Update $\bar{\boldsymbol{\lambda}}^{k+1}$ according to Eq. (8)
    Update $\boldsymbol{x}^{k+1}$ according to Eq. (9)
    Update $\boldsymbol{\lambda}^{k+1}$ according to Eq. (10)
end for
```

Algorithm 2 Stochastic extra-gradient alternating direction method with augmented Lagrangian function (SEGAL)

```
Input: $\boldsymbol{x}^0$, $\boldsymbol{y}^0$, and $\boldsymbol{\lambda}^0$.
Output: $\tilde{\boldsymbol{y}}^t = \sum_{k=0}^{t} \alpha^{k+1}\boldsymbol{y}^{k+1}$, $\tilde{\boldsymbol{x}}^t = \sum_{k=0}^{t} \alpha^{k+1}\bar{\boldsymbol{x}}^{k+1}$, and
    $\tilde{\boldsymbol{\lambda}}^t = \sum_{k=0}^{t} \alpha^{k+1}\bar{\boldsymbol{\lambda}}^{k+1}$.
for $k = 0, 1, 2, \ldots$ do
    Choose two data samples $\xi_1^{k+1}$ and $\xi_2^{k+1}$ randomly
    Update $\boldsymbol{y}^{k+1}$ according to Eq. (5)
    Update $\bar{\boldsymbol{x}}^{k+1}$ according to Eq. (11)
    Update $\bar{\boldsymbol{\lambda}}^{k+1}$ according to Eq. (8)
    Update $\boldsymbol{x}^{k+1}$ according to Eq. (12)
    Update $\boldsymbol{\lambda}^{k+1}$ according to Eq. (10)
end for
```

The SEGAL algorithm shares some common features with the EGADM algorithm (Lin et al., 2015) and the SEGL (Algorithm 1). The first subproblem in Algorithm 2 is the same as that in Algorithm 1. However, the SEGAL algorithm takes a stochastic gradient estimation, which is different from Algorithm 1, i.e.,

$$\begin{aligned}\bar{\boldsymbol{x}}^{k+1} = \boldsymbol{x}^k - c^{k+1}(\nabla f(\boldsymbol{x}^k, \xi_1^{k+1}) - \boldsymbol{A}^{\mathrm{T}}\boldsymbol{\lambda}^k \\ + \gamma\boldsymbol{A}^{\mathrm{T}}(\boldsymbol{A}\boldsymbol{x}^k + \boldsymbol{B}\boldsymbol{y}^{k+1} - \boldsymbol{b})),\end{aligned} \tag{11}$$

$$\begin{aligned}\boldsymbol{x}^{k+1} = \boldsymbol{x}^k - c^{k+1}(\nabla f(\bar{\boldsymbol{x}}^{k+1}, \xi_2^{k+1}) - \boldsymbol{A}^{\mathrm{T}}\bar{\boldsymbol{\lambda}}^{k+1} \\ + \gamma\boldsymbol{A}^{\mathrm{T}}(\boldsymbol{A}\bar{\boldsymbol{x}}^{k+1} + \boldsymbol{B}\boldsymbol{y}^{k+1} - \boldsymbol{b})).\end{aligned} \tag{12}$$

Note that compared to the SEGL (Algorithm 1), the SEGAL takes a stochastic gradient estimation on the augmented Lagrangian function in each subproblem, while the SEGL takes only a stochastic gradient estimation on the augmented Lagrangian function on the $\boldsymbol{y}$-update subproblem, and performs stochastic gradient estimation on Lagrangian function on the other subproblems.

4 Experiments

In this section, we test the performance of the proposed algorithms (SEGL and SEGAL) to solve the fused logistic regression and problem (13) with the general convex objective function and graph-guided regularized logistic regression (14) with a strongly convex objective function (Zhong and Kwok, 2013), which are formulated as follows:

$$\min_{\boldsymbol{x}} \ \ell(\boldsymbol{x}) + \gamma\|\boldsymbol{x}\|_1 + \lambda\sum_{j=2}^{n}|\boldsymbol{x}_j - \boldsymbol{x}_{j-1}|, \tag{13}$$

$$\min_{\boldsymbol{x}} \ \ell(\boldsymbol{x}) + \frac{\gamma}{2}\|\boldsymbol{x}\|_2^2 + \lambda\sum_{i=1}^{n}\left|\sum_{j=1}^{m}\boldsymbol{w}_{ij}\boldsymbol{x}_j\right|, \tag{14}$$

where $\ell(\boldsymbol{x})$ denotes the average logistic loss function, which is defined as

$$\ell(\boldsymbol{x}) = \frac{1}{m}\sum_{i=1}^{m}\log(1 + \exp(-b_i(\boldsymbol{a}_i^{\mathrm{T}}\boldsymbol{x}))), \tag{15}$$

and the ℓ_1 norm $\|\boldsymbol{x}\|_1$ is imposed to promote the sparsity of weight vector $\boldsymbol{x}$.

Note that this problem is hard to solve by the ADMM, because of the difficulty in computing the proximal mapping of $\ell(\boldsymbol{x})$. To handle the above problems, we equivalently reformulate them as

$$\begin{aligned}\min_{\boldsymbol{x},\boldsymbol{y}} \quad & f(\boldsymbol{x}) + \gamma\|\boldsymbol{x}\|_1 + \lambda\|\boldsymbol{y}\|_1 \\ \text{s.t.} \quad & \boldsymbol{F}\boldsymbol{x} - \boldsymbol{y} = \boldsymbol{0},\end{aligned} \tag{16}$$

$$\begin{aligned}\min_{\boldsymbol{x},\boldsymbol{y}} \quad & f(\boldsymbol{x}) + \frac{\gamma}{2}\|\boldsymbol{x}\|_2^2 + \lambda\|\boldsymbol{y}\|_1 \\ \text{s.t.} \quad & \boldsymbol{F}\boldsymbol{x} - \boldsymbol{y} = \boldsymbol{0}.\end{aligned} \tag{17}$$

Here $f(\boldsymbol{x}) = \frac{1}{N}\sum_{i=1}^{N} l(\boldsymbol{x}, \xi_i)$, where $l(\boldsymbol{x}, \xi_i)$ is the logistic loss function on ξ_i and $\lambda > 0$ is a parameter. $\boldsymbol{F}$ is the penalty matrix promoting the desired sparse structure of $\boldsymbol{x}$. Specifically, $\boldsymbol{F} \in \mathbb{R}^{(n-1)\times n}$ in problem (16) is specified as a matrix with all ones on the diagonal, negative ones on the super-diagonal area, and zeros elsewhere. $\boldsymbol{F}$ in problem (17) is generated by sparse inverse covariance selection (Hsieh et al., 2013).

4.1 SEGL and SEGAL for the fused logistic regression

We first present the details to solve the fused logistic regression using Algorithm 1 in this subsection. It is easy to check that the gradients of $\ell(\boldsymbol{y}, \xi)$ with respect to $\boldsymbol{y}$ can be obtained by

$$\nabla_{\boldsymbol{y}}\ell(\boldsymbol{y}, \xi_i) = -\frac{b_i\boldsymbol{a}_i}{m}\left(1 - \frac{1}{1 + \exp(-b_i\boldsymbol{a}_i^{\mathrm{T}}\boldsymbol{y})}\right), \tag{18}$$

where the division operation is element-wise.

Based on these discussions, we can summarize the SEGL to solve problem (16) as Algorithm 3, where the ℓ_1 shrinkage operator is defined as

$$\operatorname{shrink}(z, \tau) = \operatorname{sign}(z) \circ \max\{|z| - \tau, 0\}, \tag{19}$$

where '∘' is the element-wise multiplication.

Algorithm 3 SEGL for the fused logistic regression

```
Input: Weight parameter $\alpha^0$, step size $c^0$, $\boldsymbol{x}^0$, $\boldsymbol{y}^0$, and $\boldsymbol{\lambda}^0$.
Output: $\tilde{\boldsymbol{y}}^t = \sum_{k=0}^{t} \alpha^{k+1}\boldsymbol{y}^{k+1}$, $\tilde{\boldsymbol{x}}^t = \sum_{k=0}^{t} \alpha^{k+1}\bar{\boldsymbol{x}}^{k+1}$, and $\tilde{\boldsymbol{\lambda}}^t = \sum_{k=0}^{t} \alpha^{k+1}\bar{\boldsymbol{\lambda}}^{k+1}$.
for $k = 0, 1, 2, \ldots$ do
    $\boldsymbol{y}^{k+1} = \underset{\boldsymbol{y}\in\mathcal{Y}}{\operatorname{argmin}}\ \mathcal{L}_\gamma(\boldsymbol{x}^k, \boldsymbol{y}; \boldsymbol{\lambda}^k) + \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{y}^k\|_{\boldsymbol{H}}^2$
    $\bar{\boldsymbol{x}}^{k+1} = \boldsymbol{x}^k - c^{k+1}(\nabla\ell(\boldsymbol{x}^k, \xi_1^{k+1}) - \boldsymbol{A}^{\mathrm{T}}\boldsymbol{\lambda}^k)$
    $\bar{\boldsymbol{\lambda}}^{k+1} = \boldsymbol{\lambda}^k - \gamma(\boldsymbol{A}\boldsymbol{x}^k + \boldsymbol{B}\boldsymbol{y}^{k+1} - \boldsymbol{b})$
    $\boldsymbol{x}^{k+1} = \boldsymbol{x}^k - c^{k+1}(\nabla\ell(\bar{\boldsymbol{x}}^{k+1}, \xi_2^{k+1}) - \boldsymbol{A}^{\mathrm{T}}\bar{\boldsymbol{\lambda}}^{k+1})$
    $\boldsymbol{\lambda}^{k+1} = \boldsymbol{\lambda}^k - \gamma(\boldsymbol{A}\bar{\boldsymbol{x}}^{k+1} + \boldsymbol{B}\boldsymbol{y}^{k+1} - \boldsymbol{b})$
    if the stop criterion is satisfied then
        Break
    end if
end for
```

We omit the details of the SEGAL algorithm to solve fused logistic regression problem, since it is easy to obtain in a similar procedure.

4.2 Numerical results

In this subsection, we present the experimental results of all these compared algorithms to solve fused logistic regression problem (16) and graph-guided minimization problem (17). All compared algorithms were implemented in Matlab 2016a and executed on a laptop with an Intel® Core™ i7-4710-MQ CPU @ 2.5 GHz and 16 GB memory.

In the experiments, we compared our SEGL and SEGAL algorithms with six existing stochastic ADMM-type algorithms: SGADM (Gao et al., 2017), SADMM (Ouyang et al., 2013), OPG-ADMM (Suzuki, 2013), RDA-ADMM (Suzuki, 2013), and two adaptive SADMMs (SADMMdiag and SADMM-full) (Zhao et al., 2015). We excluded the online ADMM (Wang and Banerjee, 2013), since Suzuki (2013) showed that RDA-ADMM performs better than the online ADMM. FSADMM (Zhong and Kwok, 2013) was also excluded, since it requires the storage of all gradients, which results in impractical performance in some complex applications (Johnson and Zhang, 2013).

The experiments were conducted on five binary classification datasets: Splice, Svmguide3, Mushrooms, A9a, and W8a. Table 1 shows the details of them. For each dataset, we calculated the Lipschitz constant L as its classical upper bound $0.25\max_{1\le i\le n}\|\boldsymbol{a}_i\|^2$. The regularization parameters were $\lambda = 5\times10^{-3}$ and $\gamma = 5\times10^{-4}$ for problem (16), and $\lambda = 10^{-5}$ and $\gamma = 10^{-2}$ for problem (17). In the fused logistic regression, the step size was $c^{k+1} = 1/(\sqrt{k+1} + \tilde{L})$ and the weight of iterations was $\alpha^{k+1} = 1/(t+1)$. In the graph-guided regularized regression, the step size was $c^{k+1} = 2/(\mu\,(k+1) + 2\tilde{L})$, the weights of iterations were $\alpha^{k+1} = 1/(t+1)$ for SEGL1 and SEGAL1 (uniformly averaged) and $\alpha^{k+1} = \frac{2(k+3)}{(t+1)(t+6)}$ for SEGL2 and SEGAL2 (non-uniformly averaged). In the settings above, $\tilde{L} = \max\{8\gamma\sigma_{\max}(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A}) + \mu, \sqrt{8L^2 + \gamma\sigma_{\max}(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A})} + \mu\}$, where $\sigma_{\max}(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A})$ denotes the largest eigenvalue of $\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A}$, and $\mu = 0$ when $f(\cdot)$ is a general convex objective function.

Table 1 Statistics of datasets

Dataset	Number of samples	Dimensionality
Splice	1000	60
Svmguide3	1243	21
Mushrooms	8124	112
A9a	32 561	123
W8a	64 700	300

We used cross validation to select the parameters of other algorithms. Additionally, we used the metrics including objective value, test loss, and prediction accuracy to compare our method with other methods. The 'objective value' means the sum of the loss function and regularized terms evaluated on a training data sample, while the 'test loss' means the value of the loss function evaluated on a test data sample. Specifically, we used objective function values on training datasets, test losses (i.e., $l(\boldsymbol{x})$) on test datasets, and computational time costs on training datasets.

Fig. 1 shows the objective value, test loss, and prediction accuracy as the functions of time costs on the fused logistic regression task, where the objective function is convex but not necessarily strongly convex. We observe that our method mostly achieves the best performance, followed by six stochastic ADMM-type algorithms. We find that the prediction accuracy of the SEGL algorithm is competitive with other algorithms, which supports the usage of extragradient in the SEGL algorithm. The performance of our SEGL and SEGAL algorithms on five datasets is the most stable and effective among all methods. Furthermore, SEGAL obtains better accuracy on large-scale datasets, which recommends using SEGAL in general convex problems.

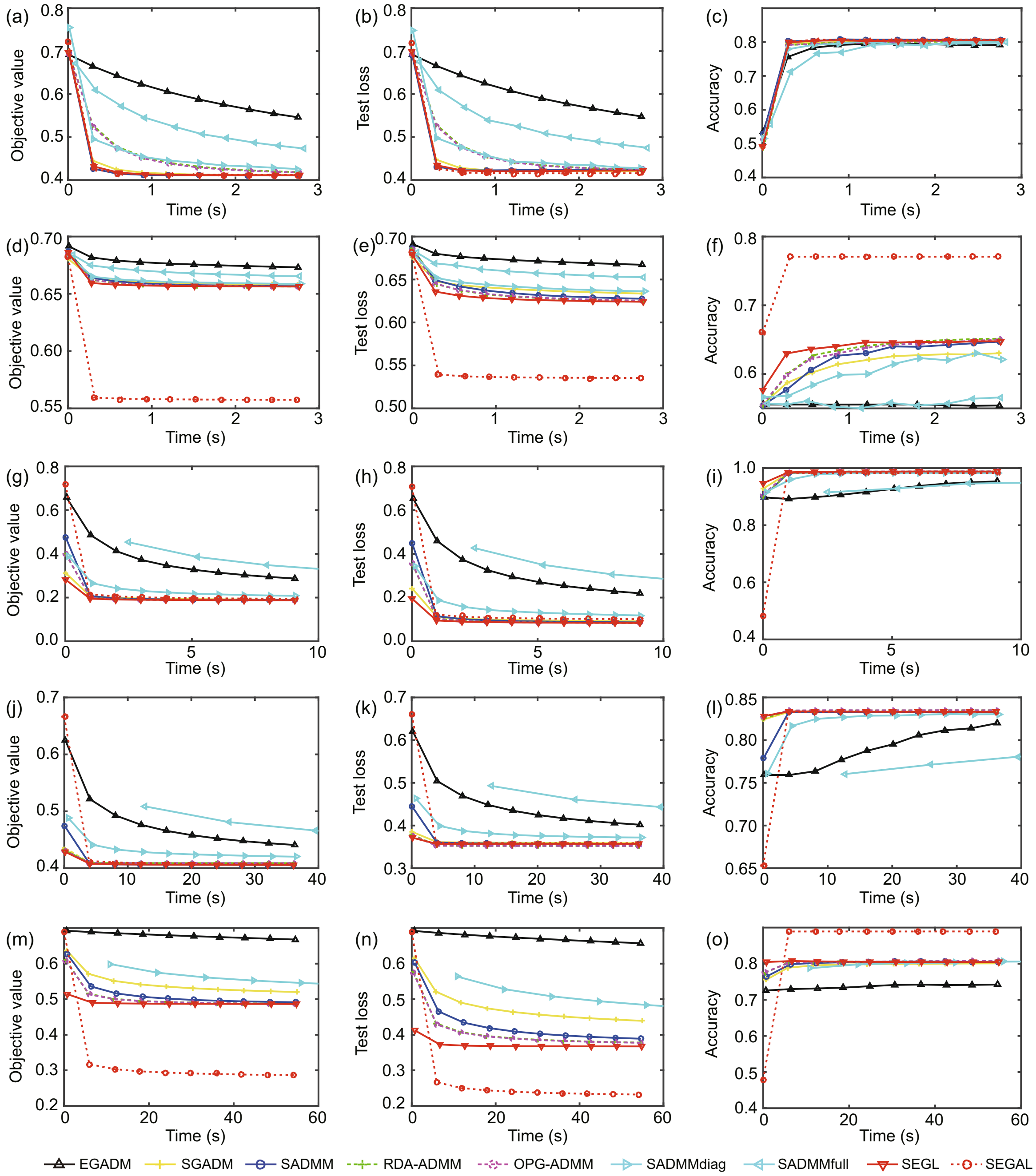

Fig. 1 Comparison of SEGL and SEGAL with EGADM, SGADM, SADMM, RDA-ADMM, OPG-ADMM, SADMMdiag, and SADMMfull on fused logistic regression tasks under five binary classification datasets: (a), (b), and (c) on Splice; (d), (e), and (f) on Svmguide3; (g), (h), and (i) on Mushrooms; (j), (k), and (l) on A9a; (m), (n), and (o) on W8a

We further compared our algorithm with other algorithms on the graph-guided regularized logistic regression task, where the objective function is strongly convex. We used both uniformly and non-uniformly averaged iterations, denoted as SEGL1, SEGAL1, SEGL2, and SEGAL2, respectively. The experimental results, as shown in Fig. 2, show that our algorithms consistently outperform other algorithms and exhibit the advantage with non-uniformly averaged iterations over the other algorithms with uniformly averaged iterations. Furthermore, SEGL achieves better accuracy on large-scale datasets, which recommends using SEGL in strongly convex problems.

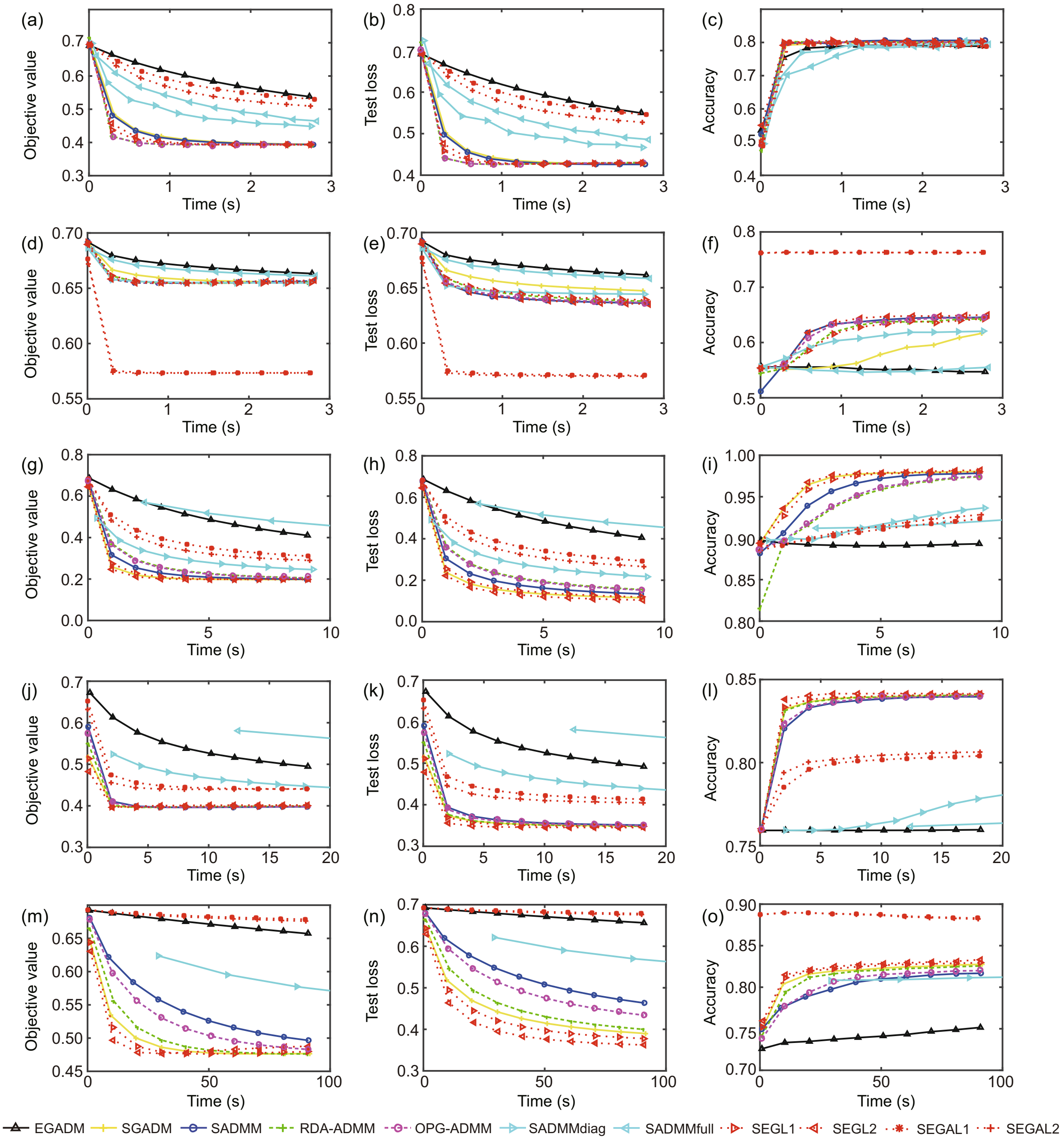

Fig. 2 Comparison of SEGL1, SEGAL1, SEGL2, and SEGAL2 with EGADM, SGADM, SADMM, RDA-ADMM, OPG-ADMM, SADMMdiag, and SADMMfull on graph-guided regularized logistic regression tasks under five binary classification datasets: (a), (b), and (c) on Splice; (d), (e), and (f) on Svmguide3; (g), (h), and (i) on Mushrooms; (j), (k), and (l) on A9a; (m), (n), and (o) on W8a

5 Conclusions

In this paper, we have proposed stochastic variants of the extra-gradient alternating direction method, named stochastic extra-gradient alternating direction method with Lagrangian function (SEGL) and augmented Lagrangian function (SEGAL) to solve linearly constrained optimization problem (3) in large scale. The proposed algorithm inherits the stability and efficiency of the extra-gradient alternating method and the ability of stochastic optimization algorithms to handle large-scale problems. In the numerical experiments conducted on fused logistic regression and graph-guided regularized logistic regression problems, we have compared our SEGL and SEGAL algorithms with six existing stochastic ADMM-type algorithms and two adaptive stochastic ADMM-type algorithms. The experiments results demonstrated the efficacy of the proposed SEGL and SEGAL beyond other competing algorithms.

References

Azadi S, Sra S, 2014. Towards an optimal stochastic alternating direction method of multipliers. Int Conf on Machine Learning, p.620-628.

Boyd S, Parikh N, Chu E, et al., 2011. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found Trends Mach Learn*, 3(1):1-122. https://doi.org/10.1561/2200000016

Cortes C, Vapnik V, 1995. Support-vector networks. *Mach Learn*, 20(3):273-297. https://doi.org/10.1023/A:1022627411411

Gao X, Jiang B, Zhang S, 2017. On the information-adaptive variants of the ADMM: an iteration complexity perspective. *J Sci Comput*, 76(1):327-363. https://doi.org/10.1007/s10915-017-0621-6

Hastie T, Tibshirani R, Friedman J, 2001. The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer-Verlag, New York, USA.

Hsieh CJ, Sustik MA, Dhillon IS, et al., 2013. BIG & QUIC: sparse inverse covariance estimation for a million variables. Advances in Neural Information Processing Systems, p.3165-3173.

Johnson R, Zhang T, 2013. Accelerating stochastic gradient descent using predictive variance reduction. Advances in Neural Information Processing Systems, p.315-323.

Lin T, Ma S, Zhang S, 2015. An extra-gradient-based alternating direction method for convex minimization. *Found Comput Math*, 17(1):35-59. https://doi.org/10.1007/s10208-015-9282-8

Lin T, Qiao L, Zhang T, et al., 2018. Stochastic primal-dual proximal extra-gradient descent for compositely regularized optimization. *Neurocomputing*, 273:516-525. https://doi.org/10.1016/j.neucom.2017.07.066

Ouyang H, He N, Tran L, et al., 2013. Stochastic alternating direction method of multipliers. Int Conf on Machine Learning, p.80-88.

Qiao LB, Zhang BF, Su JS, et al., 2017. A systematic review of structured sparse learning. *Front Inform Technol Electron Eng*, 18(4):445-463. https://doi.org/10.1631/FITEE.1601489

Suzuki T, 2013. Dual averaging and proximal gradient descent for online alternating direction multiplier method. Int Conf on Machine Learning, p.392-400.

Tibshirani R, 1996. Regression shrinkage and selection via the Lasso. *J R Stat Soc Ser B*, 1:267-288.

Tibshirani R, Saunders M, Rosset S, et al., 2005. Sparsity and smoothness via the fused Lasso. *J R Stat Soc Ser B*, 67:91-108. https://doi.org/10.1111/j.1467-9868.2005.00490.x

Wang H, Banerjee A, 2013. Online alternating direction method (longer version). arXiv Preprint, 1306.3721.

Yang JF, Yuan XM, 2013. Linearized augmented Lagrangian and alternating direction methods for nuclear norm minimization. *Math Comput*, 82(281):301-329. https://doi.org/10.1090/S0025-5718-2012-02598-1

Zhao P, Yang J, Zhang T, et al., 2015. Adaptive stochastic alternating direction method of multipliers. Int Conf on Machine Learning, p.69-77.

Zhong W, Kwok JT, 2013. Fast stochastic alternating direction method of multipliers. Int Conf on Machine Learning, p.46-54.