Adaptive Stochastic Alternating Direction Method of Multipliers

Peilin Zhao*  Jinwei Yang†  Tong Zhang‡  Ping Li‡

*Institute for Infocomm Research, A*STAR, Singapore
†Rutgers University; and University of Notre Dame, USA
‡Rutgers University, USA; and Baidu Research, China

July 9, 2015
Outline

1. Background Review
   - Problem Settings
   - Notations
   - ADMM Algorithm
   - Limitations of ADMM
   - Stochastic ADMM
   - Motivation

2. Adaptive Stochastic ADMM
   - Adaptive Stochastic ADMM
   - Ada-SADMM with Diagonal Matrix
   - Ada-SADMM with Full Matrix

3. Experiments
   - Application
   - Experimental Testbed and Setup
   - Performance Evaluation

4. Conclusion
Problem Settings

Objective of Alternating Direction Method of Multipliers (ADMM)

Objective Function:

\[
\min_{w \in W, v \in V} \ f((w^T, v^T)^T) := \mathbb{E}_\xi \ell(w, \xi) + \varphi(v),
\]

s.t. \( Aw + Bv = b, \) 

- For simplicity, \( \ell(w) = \mathbb{E}_\xi \ell(w, \xi) \)
  - \( \mathbb{E}_\xi \ell(w, \xi) \) can be \( \frac{1}{n} \sum_{i=1}^{n} \ell(w, z_i) \)

- Optimal solution: \( (w_*^T, v_*^T)^T \)

- Useful for many real-world problems
  - e.g. compressed sensing, video processing, and matrix completion.

Goal

Find \( (w^T, v^T) \), s.t.

- \( f((w^T, v^T)^T) - f((w_*^T, v_*^T)^T) \leq \epsilon \)

- \( \|Aw + Bv - b\| \leq \epsilon \)
Notations

- $\|w\|_G := \sqrt{w^\top G w}$, if $G$ is PSD.
- $\| \cdot \|$ to denote $\| \cdot \|_2$.
- $\langle \cdot , \cdot \rangle$ to denote inner product in Euclidean space.
- Proximal function: $\phi_t(w) = \frac{1}{2} \|w\|_{H_t}^2 = \frac{1}{2} \langle w, H_tw \rangle$, where $H_t$ is PSD,
- Bregman divergence of $\phi_t(w)$:

$$B_{\phi_t}(w, u) = \phi_t(w) - \phi_t(u) - \langle \nabla \phi_t(u), w - u \rangle = \frac{1}{2} \|w - u\|_{H_t}^2.$$
Augmented Lagrangian

To solve the problem (1), the **Augmented Lagrangian** is introduced:

\[
\mathcal{L}_\beta(w, v, \theta) := \ell(w) + \varphi(v) - \langle \theta, Aw + Bv - b \rangle + \frac{\beta}{2} \|Aw + Bv - b\|_2^2,
\]

where $\beta > 0$. Note: $\ell(w) = \mathbb{E}_\xi \ell(w, \xi)$.

Reasons to add $\frac{\beta}{2} \|Aw + Bv - b\|_2^2$:
- to introduce strongly convexity
- $Aw_* + Bv_* - b = 0$

### ADMM Algorithm

ADMM iteratively updates the primal $w, v$ and the dual $\theta$ by fixing the others

\[
\begin{align*}
w_{t+1} &= \arg \min_w \mathcal{L}_\beta(w, v_t, \theta_t), \\
v_{t+1} &= \arg \min_v \mathcal{L}_\beta(w_{t+1}, v, \theta_t), \\
\theta_{t+1} &= \theta_t - \beta(Aw_{t+1} + Bv_{t+1} - b).
\end{align*}
\]
Limitations of ADMM

calculating $E_{\xi} \ell(w, \xi)$ may be too expensive, since
- we may only have an unbiased estimate of $\ell(w)$ or
- the expectation $E_{\xi} \ell(w, \xi)$ is an empirical one for big data problems.
Stochastic Augmented Lagrangian

Instead of using Augmented Lagrangian,

\[ \mathcal{L}_\beta(w, v, \theta) := \ell(w) + \varphi(v) - \langle \theta, Aw + Bv - b \rangle + \frac{\beta}{2} \| Aw + Bv - b \|^2, \]

Stochastic Augmented Lagrangian is introduced as

\[ \hat{\mathcal{L}}_\beta(w, v, \theta) = \langle g_t, w \rangle + \varphi(v) - \langle \theta, Aw + Bv - b \rangle + \frac{\beta}{2} \| Aw + Bv - b \|^2 + \frac{1}{\eta} \| w - w_t \|^2, \]

where \( g_t = \ell'(w_t, \xi_t) \).

Stochastic ADMM Algorithm

\[
\begin{align*}
    w_{t+1} &= \text{arg min}_w \hat{\mathcal{L}}_\beta(w, v_t, \theta_t), \\
    v_{t+1} &= \text{arg min}_v \hat{\mathcal{L}}_\beta(w_{t+1}, v, \theta_t), \\
    \theta_{t+1} &= \theta_t - \beta (Aw_{t+1} + Bv_{t+1} - b).
\end{align*}
\]
According to the idea of AdaGrad [Duchi et. al., JMLR 2011]

- Using $\| w - w_t \|^2$ is a suboptimal choice
- Better to use adaptive functions based on the examples
Adaptive Stochastic Augmented Lagrangian

\[ \mathcal{L}_{\beta,t}(w, v, \theta) = \langle g_t, w \rangle + \varphi(v) - \langle \theta, Aw + Bv - b \rangle + \frac{\beta}{2} \|Aw + Bv - b\|^2 + \frac{1}{\eta} B_{\phi_t}(w, w_t), \]

where \( g_t = \ell'(w_t, \xi_t) \)
- \( \phi_t = \frac{1}{2}\|w\|_2^2 \) gives Stochastic ADMM
- Better \( \phi_t = \frac{1}{2}\|w\|_{H_t}^2 \) will be specified later.


Initialize: \( w_1 = 0, u_1 = 0, \theta_1 = 0, H_1 = aI, \) and \( a > 0. \)

for \( t = 1, 2, \ldots, T \) do
- Compute \( g_t = \ell'(w_t, \xi_t); \)
- Update \( H_t \) and compute \( B_{\phi_t}; \)
- \( w_{t+1} = \arg\min_{w \in \mathcal{W}} \mathcal{L}_{\beta,t}(w, v_t, \theta_t); \)
- \( v_{t+1} = \arg\min_{v \in \mathcal{V}} \mathcal{L}_{\beta,t}(w_{t+1}, v, \theta_t); \)
- \( \theta_{t+1} = \theta_t - \beta(Aw_{t+1} + Bv_{t+1} - b); \)
end for
Theorem 1: Convergence Rate of Ada-SADMM with General $H_t$

Let $\ell(w, \xi_t)$ and $\varphi(w)$ be convex functions and $H_t$ be positive definite, for $t \geq 1$. For any $T \geq 1$ and $\rho > 0$, Ada-SADMM satisfies:

$$
\mathbb{E}[f(\bar{u}_T) - f(u_*) + \rho \|A\bar{w}_T + B\bar{v}_T - b\|] \leq 
\frac{1}{2T} \left\{ \mathbb{E} \sum_{t=1}^{T} \left[ \frac{2}{\eta} (H_{\phi_t}(w_t, w_*) - H_{\phi_t}(w_{t+1}, w_*)) + \eta \|g_t\|_{H_t^{-1}}^2 \right] + \beta D_{v_*,B}^2 \right\} + \frac{\rho^2}{\beta},
$$

where $\bar{u}_T = \left( \frac{1}{T} \sum_{t=1}^{T} w_t^T, \frac{1}{T} \sum_{t=2}^{T+1} v_t^T \right)^T$, $u_* = (w_*^T, v_*^T)^T$, and $(\bar{w}_T, \bar{v}_T) = (\frac{1}{T} \sum_{t=2}^{T+1} w_t, \frac{1}{T} \sum_{t=2}^{T+1} v_t)$, and $D_{v_*,B} = \|Bv_*\|$.

Remark

So, if $g_t$’s are known, $H_t = H$ should minimizes $\sum_{t=1}^{T} \|g_t\|_{H_t^{-1}}^2$. 
Advantages using Diagonal $H_t$

- the diagonal matrix will provide results easier to understand
- the computational cost is lower

Proposition 1: Optimal diagonal $H_t$, if all $g_t$ are provided in advance

For any $g_1, g_2, \ldots, g_T \in \mathbb{R}^{d_1}$, we have

$$\min_{\text{diag}(s) \succeq 0, \ 1^T s \leq c} \sum_{t=1}^{T} \|g_t\|_{\text{diag}(s)}^2 = \frac{1}{c} \left( \sum_{i=1}^{d_1} \|g_{1:T,i}\| \right)^2,$$

where $g_{1:T,i} = (g_{1,i}, \ldots, g_{T,i})^T$ and the minimum is attained at

$$s_i = c \|g_{1:T,i}\| / \sum_{j=1}^{d_1} \|g_{1:T,j}\|$$
Adaptive diagonal $H_t$

$g_t$ are sequentially received, so

$$H_t = aI + \text{diag}(s_t), \quad s_{t,i} = \|g_{1:t,i}\|, \quad a \geq 0.$$  

It is easy to verify that $H_t$ is a nearly optimal choice, specifically:

$$\sum_{t=1}^{T} \frac{\|g_t\|^2_{H_t^{-1}}}{\min_{\text{diag}(s) \geq 0, \ 1^\top s \leq \frac{1}{4}} \sum_{t=1}^{T} \|g_t\|^2_{\text{diag}(s)^{-1}}} \leq \sqrt{T \sum_{t=1}^{T} \|g_t\|^2_{H_t^{-1}}} \leq \sqrt{T \sum_{t=1}^{T} \|g_t\|^2_{\text{diag}(s)^{-1}}}.$$  

Algorithm: Ada-SADMM$_{\text{diag}}$

Initialize: $w_1 = 0, u_1 = 0, \theta_1 = 0$, and $a > 0$.

for $t = 1, 2, \ldots, T$ do

Compute $g_t = \ell'(w_t, \xi_t)$;

Update $H_t = aI + \text{diag}(s_t)$, where $s_{t,i} = \|g_{1:t,i}\|$;

$w_{t+1} = \arg\min_w L_{\beta,t}(w, v_t, \theta_t)$;

$v_{t+1} = \arg\min_{v \in \mathcal{V}} L_{\beta,t}(w_{t+1}, v, \theta_t)$;

$\theta_{t+1} = \theta_t - \beta(Aw_{t+1} + Bv_{t+1} - b)$;

end for
Ada-SADMM with Diagonal Matrix

Theorem 2: Convergence Rate for Ada-SADMM$_{diag}$

Let $\ell(w, \xi_t)$ and $\varphi(w)$ be convex. Then for any $T \geq 1$ and $\rho > 0$ if $\eta = D_w,\infty / \sqrt{2}$ where $D_w,\infty = \max_{w,w'} \|w - w'\|_{\infty}$, we have

$$
\mathbb{E}[f(\bar{u}_T) - f(u_*) + \rho \|A\bar{w}_T + B\bar{v}_T - b\|] \leq \frac{1}{T} \left( \sqrt{2\mathbb{E}[D_w,\infty \sum_{i=1}^{d_1} \|g_{1:T,i}\|]} + \frac{\beta}{2} D_{v*,B}^2 + \frac{\rho^2}{2\beta} \right).
$$

Remark: Suppose $\mathcal{W} = \{w : \|w\|_{\infty} \leq 1\}$

If $x_t \in \{-1, 0, 1\}^{d_1}$, feature $i$ appears with probability $p_i = \min(1, i^{-2})$, then

$$
\mathbb{E}[D_w,\infty \sum_{i=1}^{d_1} \|g_{1:T,i}\|] \leq 2 \ln(d_1) \sqrt{T} \leq \sqrt{d_1 T}, \quad (\sqrt{d_1 T} \text{ is counterpart for Stochastic ADMM })
$$
Advantages using Full matrix $H_t$

- full matrix may be helpful for tasks with low dimension.
- it will provide us with a more complete insight.

Proposition 2: Optimal Full $H_t$, if all $g_t$ are provided in advance

For any $g_1, g_2, \ldots, g_T \in \mathbb{R}^{d_t}$, we have the following equality

$$
\min_{S \succeq 0, \tr(S) \leq c} \sum_{t=1}^T \|g_t\|_2^2 S^{-1} = \frac{1}{c} \tr(G_T),
$$

where $G_T = \sum_{t=1}^T g_t g_t^\top$. and the minimizer is attained at

$$
S = cG_T^{1/2} / \tr(G_T^{1/2})
$$

If $G_T$ is not of full rank, then we can use its pseudo-inverse.
Adaptive Full $H_t$

$g_t$ are received sequentially, we propose to update the $H_t$ incrementally as

$$H_t = aI + G_t^2, \quad G_t = \sum_{i=1}^{t} g_i g_i^T, \quad t = 1, \ldots, T.$$ 

This update is nearly optimal, since

$$\sum_{t=1}^{T} \|g_t\|_{H_t^{-1}}^2 \leq 2 \text{tr}(G_T^2).$$

Algorithm: Ada-SADMM$_{\text{full}}$

**Initialize:** $w_1 = 0$, $u_1 = 0$, $\theta_1 = 0$, $G_0 = 0$, and $a > 0$

**for** $t = 1, 2, \ldots, T$ **do**

Compute $g_t = \ell'(w_t, \xi_t)$;

Update $G_t = G_{t-1} + g_t g_t^T$;

Update $H_t = al + S_t$, where $S_t = G_t^2$;

$w_{t+1} = \arg\min_w \mathcal{L}_{\beta,t}(w, v_t, \theta_t)$;

$v_{t+1} = \arg\min_v \mathcal{L}_{\beta,t}(w_{t+1}, v, \theta_t)$;

$\theta_{t+1} = \theta_t - \beta(Aw_{t+1} + Bv_{t+1} - b)$;

**end for**
Theorem 3: Convergence Rate for Ada-SADMM\textsubscript{full}

Let $\ell(w, \xi_t)$ and $\varphi(w)$ be convex. Then for any $T \geq 1$, $\rho > 0$, if $\eta = D_w,2/2$, where $D_w,2 = \max_{w_1, w_2} ||w_1 - w_2||$, then

$$\mathbb{E}[f(\bar{u}_T) - f(u_*) + \rho \|Aw_T + B\bar{v}_T - b\|]$$

$$\leq \frac{1}{T} \left( \sqrt{2\mathbb{E}[D_w,2^{\mathrm{tr}} (G_T^{1/2})]} + \frac{\beta}{2}D_{v,*}^2 + \frac{\rho^2}{2\beta} \right).$$
**Graph-Guided SVM (GGSVM)**

\[
\min_{\mathbf{w}, \mathbf{v}} \frac{1}{n} \sum_{i=1}^{n} \max(0, 1 - y_i \mathbf{x}_i^T \mathbf{w}) + \frac{\gamma}{2} \|\mathbf{w}\|^2 + \nu \|\mathbf{v}\|_1,
\]

\[s.t. \ F\mathbf{w} - \mathbf{v} = 0,\]

where the matrix \( F \) is constructed based on a graph \( G = \{ \mathcal{V}, \mathcal{E} \} \).

**Method to get \( F \):**

- Compute sparse \( \Sigma^{-1} \) using the well-known sparse inverse covariance estimation
- Assign \( \alpha_{ij} = 1 \) for \( \Sigma^{-1}_{ij} \neq 0 \).
- \( \mathcal{V} = \{ \mathbf{w}_1, \ldots, \mathbf{w}_{d_1} \} \) is a set of variables
- \( \mathcal{E} = \{ \mathbf{e}_1, \ldots, \mathbf{e}_{|\mathcal{E}|} \} \), where \( \mathbf{e}_k = \{i, j\} \) is assigned with a weight \( \alpha_{ij} \).
- \( F_{ki} = \alpha_{ij} \) and \( F_{kj} = -\alpha_{ij} \).
Datasets

Table: Several real-world datasets in our experiments.

<table>
<thead>
<tr>
<th>Dataset</th>
<th># examples</th>
<th># features</th>
</tr>
</thead>
<tbody>
<tr>
<td>a9a</td>
<td>48,842</td>
<td>123</td>
</tr>
<tr>
<td>mushrooms</td>
<td>8,124</td>
<td>112</td>
</tr>
<tr>
<td>news20</td>
<td>16,242</td>
<td>100</td>
</tr>
<tr>
<td>splice</td>
<td>3,175</td>
<td>60</td>
</tr>
<tr>
<td>svmguide3</td>
<td>1,284</td>
<td>21</td>
</tr>
<tr>
<td>w8a</td>
<td>64,700</td>
<td>300</td>
</tr>
</tbody>
</table>

Each dataset is randomly divided into:
- training set with 80% of examples
- test set with the rest.
### Table: Evaluation of stochastic ADMM algorithms on the real-world data sets.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>a9a</th>
<th>mushrooms</th>
<th>news20</th>
<th>splice</th>
<th>svmguide3</th>
<th>w8a</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Objective value</td>
<td>test error rate</td>
<td>Time (s)</td>
<td>Objective value</td>
<td>test error rate</td>
<td>Time (s)</td>
</tr>
<tr>
<td>SADMM</td>
<td>2.6002 ± 0.4271</td>
<td>16.46 ± 0.75</td>
<td>56.091</td>
<td>0.7353 ± 0.2104</td>
<td>3.50 ± 1.36</td>
<td>7.662</td>
</tr>
<tr>
<td>Ada-diag</td>
<td>0.3550 ± 0.0001</td>
<td>15.01 ± 0.12</td>
<td>94.762</td>
<td>0.0096 ± 0.0005</td>
<td>0.06 ± 0.00</td>
<td>13.036</td>
</tr>
<tr>
<td>Ada-full</td>
<td>0.3545 ± 0.0001</td>
<td>14.98 ± 0.13</td>
<td>622.446</td>
<td>0.0091 ± 0.0002</td>
<td>0.02 ± 0.03</td>
<td>67.820</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SADMM</td>
<td>0.5652 ± 0.0151</td>
<td>13.33 ± 0.34</td>
<td>13.295</td>
<td>108.68 ± 20.966</td>
<td>24.54 ± 3.22</td>
<td>0.982</td>
</tr>
<tr>
<td>Ada-diag</td>
<td>0.3139 ± 0.0003</td>
<td>12.80 ± 0.15</td>
<td>22.479</td>
<td>0.3793 ± 0.0054</td>
<td>15.78 ± 0.59</td>
<td>1.367</td>
</tr>
<tr>
<td>Ada-full</td>
<td>0.3204 ± 0.0007</td>
<td>12.84 ± 0.16</td>
<td>148.524</td>
<td>0.3710 ± 0.0014</td>
<td>15.50 ± 0.79</td>
<td>7.039</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SADMM</td>
<td>1.6143 ± 0.3123</td>
<td>21.61 ± 0.52</td>
<td>0.129</td>
<td>0.3357 ± 0.0916</td>
<td>9.57 ± 0.12</td>
<td>191.754</td>
</tr>
<tr>
<td>Ada-diag</td>
<td>0.5163 ± 0.0046</td>
<td>20.56 ± 0.60</td>
<td>0.201</td>
<td>0.1526 ± 0.0010</td>
<td>9.31 ± 0.05</td>
<td>326.139</td>
</tr>
<tr>
<td>Ada-full</td>
<td>0.5230 ± 0.0044</td>
<td>20.00 ± 0.44</td>
<td>0.460</td>
<td>0.1469 ± 0.0006</td>
<td>9.29 ± 0.03</td>
<td>4027.196</td>
</tr>
</tbody>
</table>

- Adaptive stochastic ADMM algorithms converge much faster, achieve smaller test error rates than SADMM.
- Ada-SADMM$_{\text{full}}$ achieves smaller objective values than Ada-SADMM$_{\text{diag}}$.
- Ada-SADMM$_{\text{full}}$ is significantly slower while the Ada-SADMM$_{\text{diag}}$ is overall efficient compared with SADMM.
- Ada-SADMM$_{\text{diag}}$ achieves a good trade-off between efficiency and effectiveness.
Performance Evaluation

**Figure**: Left Panels: Average objective values. Middle Panels: Average test error rates. Right Panels: Average time costs (in seconds).

Similar Observations
Conclusion

- studied adaptive regularization for stochastic ADMM
- proposed two adaptive stochastic ADMM algorithms
- analyzed the theoretical convergence bounds
- validated their empirical efficacy through promising experiments
Thank you!

Any question?