Stochastic Dual Coordinate Ascent Methods for Regularized Loss Minimization

Shai Shalev-Shwartz and Tong Zhang

School of CS and Engineering, The Hebrew University of Jerusalem

"Optimization for Machine Learning", NIPS Workshop, 2012
Regularized Loss Minimization

\[
\min_w P(w) := \left[ \frac{1}{n} \sum_{i=1}^{n} \phi_i(w^\top x_i) + \frac{\lambda}{2} \|w\|^2 \right].
\]
Regularized Loss Minimization

\[
\min_w P(w) := \left[ \frac{1}{n} \sum_{i=1}^{n} \phi_i(w^\top x_i) + \frac{\lambda}{2} \|w\|^2 \right].
\]

Examples:

<table>
<thead>
<tr>
<th></th>
<th>(\phi_i(z))</th>
<th>Lipschitz</th>
<th>smooth</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVM</td>
<td>(\max{0, 1 - y_i z})</td>
<td>✓</td>
<td>✗</td>
</tr>
<tr>
<td>Logistic regression</td>
<td>(\log(1 + \exp(-y_i z)))</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Abs-loss regression</td>
<td>(</td>
<td>z - y_i</td>
<td>)</td>
</tr>
<tr>
<td>Square-loss regression</td>
<td>((z - y_i)^2)</td>
<td>✗</td>
<td>✓</td>
</tr>
</tbody>
</table>
Dual Coordinate Ascent (DCA)

Primal problem:

$$\min_w P(w) := \left[ \frac{1}{n} \sum_{i=1}^{n} \phi_i(w^\top x_i) + \frac{\lambda}{2} \|w\|^2 \right]$$

Dual problem:

$$\max_{\alpha \in \mathbb{R}^n} D(\alpha) := \left[ \frac{1}{n} \sum_{i=1}^{n} -\phi^*_i(-\alpha_i) - \frac{\lambda}{2} \left\| \frac{1}{\lambda n} \sum_{i=1}^{n} \alpha_i x_i \right\|^2 \right]$$

- **DCA**: At each iteration, optimize $D(\alpha)$ w.r.t. a single coordinate, while the rest of the coordinates are kept in tact.
- **Stochastic Dual Coordinate Ascent (SDCA)**: Choose the updated coordinate uniformly at random.
SDCA vs. SGD — update rule

Stochastic Gradient Descent (SGD) update rule:

\[ w^{(t+1)} = (1 - \frac{1}{t}) w^{(t)} - \frac{\phi'_i(w^{(t)} \top x_i)}{\lambda t} x_i \]

SDCA update rule:

1. \( \alpha^{(t+1)} = \arg\max_{\Delta \in \mathbb{R}} D(\alpha^{(t)} + \Delta e_i) \)
2. \( w^{(t+1)} = w^{(t)} + \frac{\Delta}{\lambda n} x_i \)

- Rather similar update rules.
- SDCA has several advantages:
  - Stopping criterion
  - No need to tune learning rate
SVM with the hinge loss: \( \phi_i(w) = \max\{0, 1 - y_i w^\top x_i\} \)

SGD update rule:

\[
w^{(t+1)} = (1 - \frac{1}{t}) w^{(t)} - \frac{1[y_i x_i^\top w^{(t)} < 1]}{\lambda t} x_i
\]

SDCA update rule:

1. \( \Delta = y_i \max \left(0, \min \left(1, \frac{1 - y_i x_i^\top w^{(t-1)}}{\|x_i\|^2 / (\lambda n)} + y_i \alpha_i^{(t-1)}\right)\right) - \alpha_i^{(t-1)} \)

   1. \( \alpha^{(t+1)} = \alpha^{(t)} + \Delta e_i \)

   2. \( w^{(t+1)} = w^{(t)} + \frac{\Delta}{\lambda n} x_i \)
SDCA vs. SGD — experimental observations

- On CCAT dataset, $\lambda = 10^{-6}$, smoothed loss

![Graph showing comparison between SDCA, SDCA-Perm, and SGD](image-url)
On CCAT dataset, $\lambda = 10^{-5}$, hinge-loss
How many iterations are required to guarantee $P(w^{(t)}) \leq P(w^*) + \epsilon$?

- For SGD: $\tilde{O}\left(\frac{1}{\lambda \epsilon}\right)$
- For SDCA:
  - Hsieh et al. (ICML 2008), following Luo and Tseng (1992): $O\left(\frac{1}{\nu} \log(1/\epsilon)\right)$, but, $\nu$ can be arbitrarily small
  - S and Tewari (2009), Nesterov (2010):
    - $O(n/\epsilon)$ for general $n$-dimensional coordinate ascent
    - Can apply it to the dual problem
    - Resulting rate is slower than SGD
    - And, the analysis does not hold for logistic regression (it requires smooth dual)
  - Analysis is for dual sub-optimality
Dual vs. Primal sub-optimality

- Take data which is linearly separable using a vector \( w_0 \)
- Set \( \lambda = \frac{2\epsilon}{\|w_0\|^2} \) and use the hinge-loss
- \( P(w^*) \leq P(w_0) = \epsilon \)
- \( D(0) = 0 \Rightarrow D(\alpha^*) - D(0) = P(w^*) - D(0) \leq \epsilon \)
- But, \( w(0) = 0 \) so \( P(w(0)) - P(w^*) = 1 - P(w^*) \geq 1 - \epsilon \)
- **Conclusion**: In the “interesting” regime, \( \epsilon \)-sub-optimality on the dual can be meaningless w.r.t. the primal!
Our results

- For \((1/\gamma)\)-smooth loss:
  \[
  \tilde{O} \left( \left( n + \frac{1}{\lambda} \right) \log \frac{1}{\epsilon} \right)
  \]

- For \(L\)-Lipschitz loss:
  \[
  \tilde{O} \left( n + \frac{1}{\lambda \epsilon} \right)
  \]

- For “almost smooth” loss functions (e.g. the hinge-loss):
  \[
  \tilde{O} \left( n + \frac{1}{\lambda \epsilon^{1/(1+\nu)}} \right)
  \]

where \(\nu > 0\) is a data dependent quantity.
On CCAT dataset, $\lambda = 10^{-4}$, smoothed hinge-loss

In particular, the bound of Luo and Tseng holds for cyclic order, hence must be inferior to our bound.
Smoothing the hinge-loss

\[ \phi(x) = \begin{cases} 
0 & x > 1 \\
1 - x - \frac{\gamma}{2} & x < 1 - \gamma \\
\frac{1}{2\gamma}(1 - x)^2 & \text{o.w.}
\end{cases} \]
Smoothing the hinge-loss

- Mild effect on 0-1 error

![Graphs showing the effect of γ on 0-1 error for astro-ph, CCAT, and cov1.](image-url)
Smoothing the hinge-loss

- Improves training time

- Duality gap as a function of runtime for different smoothing parameters
SDCA vs. DCA — Randomization is crucial

- On CCAT dataset, $\lambda = 10^{-4}$, smoothed hinge-loss

- In particular, the bound of Luo and Tseng holds for cyclic order, hence must be inferior to our bound
Collins et al (2008): For smooth loss, similar bound to ours (for smooth loss) but for a more complicated algorithm (Exponentiated Gradient on dual)

Lacoste-Julien, Jaggi, Schmidt, Pletscher (preprint on Arxiv):
- Study Frank-Wolfe algorithm for the dual of structured prediction problems.
- Boils down to SDCA for the case of binary hinge-loss.
- Same bound as our bound for the Lipschitz case

Le Roux, Schmidt, Bach (NIPS 2012): A variant of SGD for smooth loss and finite sample. Also obtain $\log(1/\epsilon)$. 
Extensions

- Slightly better rates for SDCA with SGD initialization
- “Proximal Stochastic Dual Coordinate Ascent” (in preparation, a preliminary version is on Arxiv)
  - Solve:
    \[
    \min_w P(w) := \left[ \frac{1}{n} \sum_{i=1}^{n} \phi_i(X_i^T w) + \lambda g(w) \right]
    \]
  - For example: \( g(w) = \frac{1}{2} \|w\|_2^2 + \frac{\sigma}{\lambda} \|w\|_1 \) leads to thresholding operator
  - For example: \( \phi_i : \mathbb{R}^k \to \mathbb{R}, \phi_i(v) = \max_{y'} \delta(y_i, y') + v_{y_i} - v_{y'} \) is useful for structured output prediction
  - Approximate dual maximization (can obtain closed form approximate solutions while still maintaining same guarantees on the convergence rate)
Proof Idea

- Main lemma: for any $t$ and $s \in [0, 1]$,

$$\mathbb{E}[D(\alpha^{(t)}) - D(\alpha^{(t-1)})] \geq \frac{s}{n} \mathbb{E}[P(w^{(t-1)}) - D(\alpha^{(t-1)})] - \left(\frac{s}{n}\right)^2 \frac{G^{(t)}}{2\lambda}$$

- $G^{(t)} = O(1)$ for Lipschitz losses

- With appropriate $s$, $G^{(t)} \leq 0$ for smooth losses
Proof Idea

- **Main lemma**: for any $t$ and $s \in [0, 1]$,

\[
\mathbb{E}[D(\alpha^{(t)}) - D(\alpha^{(t-1)})] \geq \frac{s}{n} \mathbb{E}[P(w^{(t-1)}) - D(\alpha^{(t-1)})] - \left(\frac{s}{n}\right)^2 \frac{G(t)}{2\lambda}
\]

- **Bounding dual sub-optimality**: Since $P(w^{(t-1)}) \geq D(\alpha^*)$, the above lemma yields a convergence rate for the dual sub-optimality.

- **Bounding duality gap**: Summing the inequality for iterations $T_0 + 1, \ldots, T$ and choosing a random $t \in \{T_0 + 1, \ldots, T\}$ yields,

\[
\mathbb{E} \left[ (P(w^{(t-1)}) - D(\alpha^{(t-1)})) \right] \leq \frac{n}{s(T - T_0)} \mathbb{E}[D(\alpha^{(T)}) - D(\alpha^{(T_0)})] + \frac{s G}{2\lambda n}
\]
Summary

- SDCA works very well in practice
- So far, theoretical guarantees were unsatisfactory
- Our analysis shows that SDCA is an excellent choice in many scenarios