today: finish prox example
  * catalyst ⇒ accelerate
  * non-convex opt.
  * submodular opt.

finish prox example

  * to be useful, need $\text{prox}_{\frac{2}{\lambda}}$ to be efficiently computable

  $$\text{prox}_{\frac{2}{\lambda}}(z) = \arg \min_w \frac{1}{2} \|w - z\|_2^2$$

  

  $\text{"soft-thresholding" (component-wise)}:$

  $$\text{sign}(z) \left[ |z| - \lambda \right]^+ \text{ if } |z| > \lambda$$

  $$0 \text{ otherwise.}$$

  used e.g. for loss: $l_2$-reg, least-square

  FISTA ⇒ accelerated prox. gradient method

  2nd state-of-the-art for batch lasso

  * scikit-learn ⇒ use SAGA for lasso \[\text{next class}\]

  could accelerate using "catalyst"

  Catalyst algorithm \[\text{Lin, Ma, and Tanaka} \text{ NIPS 2015}\]
"meta-algorithm": outer loop which uses a slowly convergent alg. in the inner loop to get overall acceleration (?)

main idea: use the accelerated proximal point algorithm with approximation inner loop of prox operator

proximal pt. alg.: is proximal gradient with $\beta = 0$

$$w_{t+1} = \text{prox}_{\frac{\beta}{2}}(w_t)$$

(to solve $\min_w \mathcal{L}(w)$)

Catalyst alg.: (for $\mu$-strongly convex $F(w)$)

let $\gamma \leftarrow \frac{\mu + \beta}{\mu - \beta}$ ($\gamma$ is an algorithmic parameter)

$$A \leftarrow G_t(w)$$

repeat:

$$w_{t+1} \leftarrow \text{argmin}_w \left( F(w) + \frac{1}{2\lambda} \|w - z_t\|^2 \right)$$

st. $G_t(w_{t+1}) - \min_w G_t(w) \leq \varepsilon_t$

using inner loop optimization alg. [eg. SAGA or AFW]

$$Z_{t+1} = w_{t+1} + \beta_{t+1}(w_{t+1} - w_t)$$

like a "momentum"

"extrapolation"
\( \beta_{t+1} \) is found using fancy equations so that everything works

- solve for \( \alpha_{t+1} \) in eq. \( \alpha_{t+1}^2 = (1-\alpha_{t+1})\alpha_t^2 + q\alpha_{t+1} \)
  
  (pick \( \alpha_{t+1} \in [0,1] \))

\[ \beta_{t+1} = \frac{\alpha_{t+1} (1-\alpha_{t+1})}{\alpha_t^2 + \alpha_{t+1}} \]

catalyst trick: use \( \tilde{x} \leq \epsilon \)

\( \tilde{t} \): overall # of inner loop calls

S.t. overall acceleration give an overall acceleration

with clever analysis of warm starting.

acceleration results: \( \tilde{t} \) is inner loop kth. Has convergence \( \exp(-\gamma \tilde{t}) \)

\( \tilde{\mu} > \mu + \frac{1}{t} \)

then with correct constants:

(Strong case) linear rate: \( P = \frac{1}{k} \) becomes \( \approx \frac{1}{\sqrt{k}} \) for catalyst

(weaker case) \( \frac{1}{t} \) on f becomes \( \frac{1}{2t} \)
result is can get accelerated $S_{\text{Ada}}$
\begin{align*}
A &= W \\
&\text{etc...}
\end{align*}

15h22

**Non-convex optimization**

recall: FW with line search on $f$, non-convex

\[ g(w_t) \leq O\left(\frac{1}{\sqrt{t}}\right) \]

FW-gap

\[ F(w_t) - F^* \leq \epsilon_t \]

\[ \|\nabla F(w_t)\|^2 \leq 3\epsilon_t \]

\[\text{Gradient method:}\]

\[ f(w) \leq f(w_t) + \nabla f(w_t)^T (w - w_t) + \frac{1}{2\lambda} \|w - w_t\|^2 \]

\[ w_{t+1} = w_t - \lambda \nabla f(w_t) \]

\[ \rightarrow f(w_{t+1}) \leq f(w_t) - \frac{1}{2\lambda} \|\nabla F(w_t)\|^2 \]
Faster nonconvex optimization via VR

(Reddi, Hefny, Sra, Poczos, Smola, 2016; Reddi et al., 2016)

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Nonconvex (Lipschitz smooth)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SGD</td>
<td>$O\left(\frac{1}{\epsilon^2}\right)$</td>
</tr>
<tr>
<td>GD</td>
<td>$O\left(\frac{n}{\epsilon}\right)$</td>
</tr>
<tr>
<td>SVRG</td>
<td>$O\left(n + \frac{n^{2/3}}{\epsilon}\right)$</td>
</tr>
<tr>
<td>SAGA</td>
<td>$O\left(n + \frac{n^{2/3}}{\epsilon}\right)$</td>
</tr>
<tr>
<td>MSVRG</td>
<td>$O\left(\min\left(\frac{1}{\epsilon^2}, \frac{n^{2/3}}{\epsilon}\right)\right)$</td>
</tr>
</tbody>
</table>

Remarks

New results for convex case too; additional nonconvex results
For related results, see also (Allen-Zhu, Hazan, 2016)

Linear rates for nonconvex problems

$$\min_{\theta \in \mathbb{R}^d} \ g(\theta) = \frac{1}{n} \sum_{i=1}^{n} f_i(\theta)$$

The Polyak-Łojasiewicz (PL) class of functions

$$g(\theta) - g(\theta^*) \leq \frac{1}{2\mu} \|\nabla g(\theta)\|^2$$

(Polyak, 1963); (Łojasiewicz, 1963)
Linear rates for nonconvex problems

\[ g(\theta) - g(\theta^*) \leq \frac{1}{2\mu} \| \nabla g(\theta) \|^2 \quad \text{and} \quad \mathbb{E}[g(\theta_i) - g^*] \leq \epsilon \]

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Nonconvex</th>
<th>Nonconvex-PL</th>
</tr>
</thead>
<tbody>
<tr>
<td>SGD</td>
<td>( O\left( \frac{1}{\epsilon^2} \right) )</td>
<td>( O\left( \frac{1}{\epsilon^2} \right) )</td>
</tr>
<tr>
<td>GD</td>
<td>( O\left( \frac{n}{\epsilon} \right) )</td>
<td>( O\left( \frac{n}{2\mu \log \frac{1}{\epsilon}} \right) )</td>
</tr>
<tr>
<td>SVRG</td>
<td>( O\left( n + \frac{n^{2/3}}{\epsilon} \right) )</td>
<td>( O\left( (n + \frac{n^{2/3}}{2\mu}) \log \frac{1}{\epsilon} \right) )</td>
</tr>
<tr>
<td>SAGA</td>
<td>( O\left( n + \frac{n^{2/3}}{\epsilon} \right) )</td>
<td>( O\left( (n + \frac{n^{2/3}}{2\mu}) \log \frac{1}{\epsilon} \right) )</td>
</tr>
<tr>
<td>MSVRG</td>
<td>( O\left( \min\left( \frac{1}{\epsilon^2}, \frac{n^{2/3}}{\epsilon} \right) \right) )</td>
<td>—</td>
</tr>
</tbody>
</table>

Variant of nc-SVRG attains this fast convergence!

(Reddi, Hefny, Sra, Poczos, Smola, 2016; Reddi et al., 2016)

---

**Submodular optimization**

Submodularity is the analog of convexity for tractability for set functions (combinatorial opt.)
\[ F : 2^V \to \mathbb{R} \]

\[ V = \{ e_1, \ldots, e_d \} \] is "ground set"

\[ 2^V = \{ \emptyset, V \} \] is set of all subsets of \( V \)

Concrete example:

Using model

\[ E(y) = \sum_i \Theta_i \cdot y_i - \sum_{i < j} G_{ij} \cdot y_i \cdot y_j \]

when \( G_{ij} > 0 \), \( E(y) \) is submodular

"Attractive potential"

MRF here is called "associative Markov network"

\[ F(A_y) \quad \text{where} \quad A_y = \{ i : y_i = 1 \} \]

\[ F \text{ is submodular} \iff F(A) + F(B) \geq F(A \cap B) + F(A \cup B) \quad \forall A, B \]

\[ \iff \text{Function} \ A \to F(A \cup E(i)) - F(A) \quad \text{is non-convex for all} \ i \]

\[ \text{i.e.} \quad F(A \cup \{i\}) - F(A) \leq F(B \cup \{i\}) - F(B) \]
\[ F(A) = g(I_A) \quad \text{if } g \text{ is concave} \]

then \( F \) is submodular

**link with convexity** \( \rightarrow \) Lovasz extension (cts. sol.)

embeds sets as corner of the hypercube \( V(A) = \{1, 0\}^d \)

Lovasz extension \( f \) extends \( F(A) \) from corners to whole hypercube using convex interpolation (piecewise linear function on \( \{0, 1\}^d \))

\( f(w) = F(A) \) when \( w = V(A) \)

\( F \) is submodular \( \iff \) Lovasz extension \( f \) is convex

\( \star \) can write \( f(w) = \max_{S \in B(E)} \) \( 6 \in \mathcal{B}(F) \)

\( \star \) can be computed efficiently using greedy alg.

"Base polytope"
\[
\min_{A \in V} \mathcal{J}(A) = \min_{w \in \text{lin}(A)} \left( \max_{s \in \mathcal{S}(F)} \langle s, w \rangle \right) \quad \text{use projected subgradient method}
\]

\[\frac{\partial f(w)}{\partial w} = \arg \max_{s \in \mathcal{S}(F)} \langle s, w \rangle \]

* with \(l_2\)-regularization, use duality to get a *smooth problem*

\[
\min_{s \in \mathcal{B}(F)} \frac{1}{2} ||s||^2
\]

\[\text{use "min-norm point" alg.}
\]

\[\text{variant FCFW alg.}
\]

\[\text{S.O.T.A. for submodular minimization}
\]