Aspects of Convex, Nonconvex, and Geometric Optimization
(Lecture 2)

Suvrit Sra

Massachusetts Institute of Technology

Hausdorff Institute for Mathematics (HIM)
Trimester: Mathematics of Signal Processing
January 2016

Outline

- Convex analysis, optimality
- First-order methods
- Proximal methods, operator splitting
- Stochastic optimization, incremental methods
- Nonconvex models, algorithms
- Geometric optimization

Challenge - volume of convex sets

Recall polar of convex set C defined as

$$C^\circ := \{y \mid \forall x \in C, \langle x,\, y\rangle \leq 1\}\,.$$

("Dual" set; e.g., polar of r-sphere is $1/r$-sphere)

Challenge - volume of convex sets

Recall polar of convex set C defined as

$$C^\circ := \{y \mid \forall x \in C, \langle x, y\rangle \leq 1\}.$$

("Dual" set; e.g., polar of r-sphere is $1/r$-sphere)

Mahler Volume. Let C be a symmetric convex body centered at 0. Let $V(C) := \int_{x \in C} dx$ be its volume; its Mahler volume is

$$M(C) := V(C)V(C^\circ).$$

Challenge 1. Upper bound on M achieved by Euclidean ball.

Conjecture. $M(C) \geq \frac{4^n}{n!}$ for n-dimensional sets.

Open since 1939! (known as Mahler's conjecture)

[T. Tao, Blog entry]

Recap
gradient descent

Gradient-descent

Assumption: Lipschitz continuous gradient; denoted $f \in C_L^1$

$$\|\nabla f(x) - \nabla f(y)\|_2 \leq L\|x - y\|_2$$

- ♣ Gradient vectors of closeby points are close to each other
- ♣ Objective function has "bounded curvature"
- ♣ Speed at which gradient varies is bounded

Lemma (Descent). Let $f \in C_L^1$. Then,

$$f(y) \leq f(x) + \langle \nabla f(x), y - x \rangle + \frac{L}{2}\|y - x\|_2^2$$

Theorem Let $f \in C_L^1$ and $\{x^k\}$ be sequence generated as above, with $\alpha_k = 1/L$. Then, $f(x^{k+1}) - f(x^*) = O(1/k)$.

Descent lemma – corollaries

Cor. 1 If $f \in C_L^1$, and $0 < \alpha_k < 2/L$, then $f(x^{k+1}) < f(x^k)$

$$
\begin{aligned}
f(x^{k+1}) &\leq f(x^k) + \langle \nabla f(x^k), x^{k+1} - x^k \rangle + \tfrac{L}{2}\|x^{k+1} - x^k\|_2 \\
&= f(x^k) - \alpha_k \|\nabla f(x^k)\|_2^2 + \tfrac{\alpha_k^2 L}{2}\|\nabla f(x^k)\|_2^2 \\
&= f(x^k) - \alpha_k (1 - \tfrac{\alpha_k}{2} L)\|\nabla f(x^k)\|_2^2
\end{aligned}
$$

Thus, if $\alpha_k < 2/L$ we have descent. Minimize over α_k to get best bound: this yields $\alpha_k = 1/L$

Descent lemma – corollaries

Cor. 1 If $f \in C^1_L$, and $0 < \alpha_k < 2/L$, then $f(x^{k+1}) < f(x^k)$

$$
\begin{aligned}
f(x^{k+1}) &\leq f(x^k) + \langle \nabla f(x^k), x^{k+1} - x^k \rangle + \tfrac{L}{2}\|x^{k+1} - x^k\|_2 \\
&= f(x^k) - \alpha_k \|\nabla f(x^k)\|_2^2 + \tfrac{\alpha_k^2 L}{2}\|\nabla f(x^k)\|_2^2 \\
&= f(x^k) - \alpha_k(1 - \tfrac{\alpha_k}{2}L)\|\nabla f(x^k)\|_2^2
\end{aligned}
$$

Thus, if $\alpha_k < 2/L$ we have descent. Minimize over α_k to get best bound: this yields $\alpha_k = 1/L$

Cor. 2 If $f \in C^1_L$, then

$$\langle f(x) - f(y), x - y \rangle \geq \tfrac{1}{L}\|\nabla f(x) - \nabla f(y)\|^2$$

Linear convergence

Assumption: **Strong convexity**; denote $f \in S^1_{L,\mu}$

$$f(x) \geq f(y) + \langle \nabla f(y), x - y \rangle + \frac{\mu}{2}\|x - y\|_2^2$$

- Setting $\alpha_k = 2/(\mu + L)$ yields linear rate ($\mu > 0$)

Strongly convex – linear rate

Theorem. If $f \in S^1_{L,\mu}$, $0 < \alpha < 2/(L+\mu)$, then the gradient method generates a sequence $\{x^k\}$ that satisfies

$$\|x^k - x^*\|_2^2 \leq \left(1 - \frac{2\alpha\mu L}{\mu + L}\right)^k \|x^0 - x^*\|_2.$$

Moreover, if $\alpha = 2/(L+\mu)$ then

$$f(x^k) - f^* \leq \frac{L}{2}\left(\frac{\kappa - 1}{\kappa + 1}\right)^{2k} \|x^0 - x^*\|_2^2,$$

where $\kappa = L/\mu$ is the **condition number**.

(Proof: see slides of Lecture 1)

Nonsmooth problems

Subgradient method

$$\min f(x)$$

$$x^{k+1} = x^k - \alpha_k g^k$$

where $g^k \in \partial f(x^k)$ is **any** subgradient

Subgradient method

$$\min f(x)$$

$$x^{k+1} = x^k - \alpha_k g^k$$

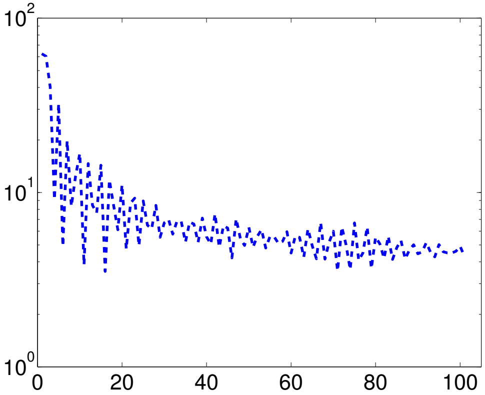
where $g^k \in \partial f(x^k)$ is **any** subgradient

- Method generates sequence $\{x^k\}_{k \geq 0}$
- Does this sequence converge to an optimal solution x^*?
- If yes, then how fast?
- Typically easier to bound $f(x^k) - f(x^*)$

Example

$$\min \quad \tfrac{1}{2}\|Ax - b\|_2^2 + \lambda\|x\|_1$$

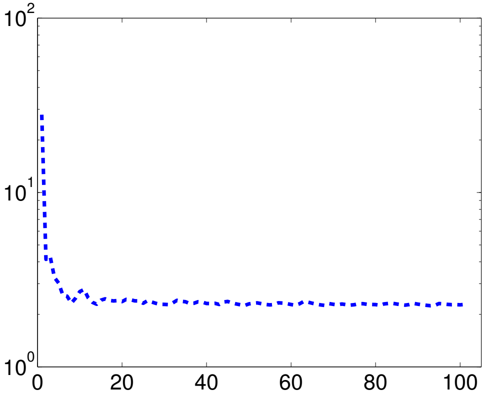
$$x^{k+1} = x^k - \alpha_k(A^T(Ax^k - b) + \lambda\,\mathrm{sgn}(x^k))$$



Example – different impl.

$$\min \quad \tfrac{1}{2}\|Ax-b\|_2^2 + \lambda\|x\|_1$$

$$x^{k+1} = x^k - \alpha_k(A^T(Ax^k - b) + \lambda\,\mathrm{sgn}(x^k))$$



Subgradient method – stepsizes

- **Constant** Set $\alpha_k = \alpha > 0$, for $k \geq 0$
- **Scaled constant** $\alpha_k = \alpha/\|g^k\|_2 \quad (\|x^{k+1} - x^k\|_2 = \alpha)$

Subgradient method – stepsizes

- **Constant** Set $\alpha_k = \alpha > 0$, for $k \geq 0$
- **Scaled constant** $\alpha_k = \alpha/\|g^k\|_2 \quad (\|x^{k+1} - x^k\|_2 = \alpha)$
- **Square summable but not summable**

$$\sum\nolimits_k \alpha_k^2 < \infty, \qquad \sum\nolimits_k \alpha_k = \infty$$

- **Diminishing scalar**

$$\lim_k \alpha_k = 0, \qquad \sum\nolimits_k \alpha_k = \infty$$

- **Adaptive stepsizes** (not covered)

Not a descent method!
Work with best f^k so far: $f^k_{\min} := \min_{0 \leq i \leq k} f^i$

Convergence rates

- Let $\phi(x) = |x|$ for $x \in \mathbb{R}$

Convergence rates

- Let $\phi(x) = |x|$ for $x \in \mathbb{R}$
- Subgradient method $x^{k+1} = x^k - \alpha_k g^k$, where $g^k \in \partial|x^k|$.

Convergence rates

- Let $\phi(x) = |x|$ for $x \in \mathbb{R}$
- Subgradient method $x^{k+1} = x^k - \alpha_k g^k$, where $g^k \in \partial|x^k|$.
- If $x^0 = 1$ and $\alpha_k = \frac{1}{\sqrt{k+1}} + \frac{1}{\sqrt{k+2}}$ (optimal step), then $|x^k| = \frac{1}{\sqrt{k+1}}$

Convergence rates

- Let $\phi(x) = |x|$ for $x \in \mathbb{R}$
- Subgradient method $x^{k+1} = x^k - \alpha_k g^k$, where $g^k \in \partial|x^k|$.
- If $x^0 = 1$ and $\alpha_k = \frac{1}{\sqrt{k+1}} + \frac{1}{\sqrt{k+2}}$ (optimal step), then $|x^k| = \frac{1}{\sqrt{k+1}}$
- Thus, $O(\frac{1}{\epsilon^2})$ iterations are needed to obtain ϵ-accuracy.
- This behavior typical for the subgradient method which exhibits $O(1/\sqrt{k})$ convergence in general

Convergence analysis

Assumptions

- Min is attained: $f^* := \inf_x f(x) > -\infty$, with $f(x^*) = f^*$

Convergence analysis

Assumptions

- Min is attained: $f^* := \inf_x f(x) > -\infty$, with $f(x^*) = f^*$
- Bounded subgradients: $\|g\|_2 \leq G$ for all $g \in \partial f$
- Bounded domain: $\|x^0 - x^*\|_2 \leq R$

Convergence results for: $f^k_{\min} := \min_{0 \leq i \leq k} f^i$

Subgradient method – convergence

Lyapunov function: Distance to x^*, not function values

$$\|x^{k+1} - x^*\|_2^2 = \|x^k - \alpha_k g^k - x^*\|_2^2$$

Subgradient method – convergence

Lyapunov function: **Distance to x^***, not function values

$$\begin{aligned} \|x^{k+1} - x^*\|_2^2 &= \|x^k - \alpha_k g^k - x^*\|_2^2 \\ &= \|x^k - x^*\|_2^2 + \alpha_k^2 \|g^k\|_2^2 - 2\langle \alpha_k g^k, x^k - x^* \rangle \end{aligned}$$

Subgradient method – convergence

Lyapunov function: Distance to x^*, not function values

$$
\begin{aligned}
\|x^{k+1} - x^*\|_2^2 &= \|x^k - \alpha_k g^k - x^*\|_2^2 \\
&= \|x^k - x^*\|_2^2 + \alpha_k^2 \|g^k\|_2^2 - 2\langle \alpha_k g^k, x^k - x^* \rangle \\
&\leq \|x^k - x^*\|_2^2 + \alpha_k^2 \|g^k\|_2^2 - 2\alpha_k (f(x^k) - f^*),
\end{aligned}
$$

since $f^* = f(x^*) \geq f(x^k) + \langle g^k, x^* - x^k \rangle$

Subgradient method – convergence

Lyapunov function: Distance to x^*, not function values

$$\begin{aligned}
\|x^{k+1} - x^*\|_2^2 &= \|x^k - \alpha_k g^k - x^*\|_2^2 \\
&= \|x^k - x^*\|_2^2 + \alpha_k^2 \|g^k\|_2^2 - 2\langle \alpha_k g^k, x^k - x^* \rangle \\
&\leq \|x^k - x^*\|_2^2 + \alpha_k^2 \|g^k\|_2^2 - 2\alpha_k (f(x^k) - f^*),
\end{aligned}$$

since $f^* = f(x^*) \geq f(x^k) + \langle g^k, x^* - x^k \rangle$

Apply same argument to $\|x^k - x^*\|_2^2$ recursively

Subgradient method – convergence

Lyapunov function: **Distance to x^***, not function values

$$\begin{aligned}
\|x^{k+1} - x^*\|_2^2 &= \|x^k - \alpha_k g^k - x^*\|_2^2 \\
&= \|x^k - x^*\|_2^2 + \alpha_k^2 \|g^k\|_2^2 - 2\langle \alpha_k g^k, x^k - x^* \rangle \\
&\leq \|x^k - x^*\|_2^2 + \alpha_k^2 \|g^k\|_2^2 - 2\alpha_k (f(x^k) - f^*),
\end{aligned}$$

since $f^* = f(x^*) \geq f(x^k) + \langle g^k, x^* - x^k \rangle$

Apply same argument to $\|x^k - x^*\|_2^2$ recursively

$$\|x^{k+1} - x^*\|_2^2 \leq \|x^0 - x^*\|_2^2 + \sum\nolimits_{t=1}^{k} \alpha_t^2 \|g^k\|_2^2 - 2 \sum\nolimits_{t=1}^{k} \alpha_t (f^t - f^*).$$

Subgradient method – convergence

Lyapunov function: **Distance to x^***, not function values

$$\begin{aligned} \|x^{k+1} - x^*\|_2^2 &= \|x^k - \alpha_k g^k - x^*\|_2^2 \\ &= \|x^k - x^*\|_2^2 + \alpha_k^2 \|g^k\|_2^2 - 2\langle \alpha_k g^k, x^k - x^* \rangle \\ &\leq \|x^k - x^*\|_2^2 + \alpha_k^2 \|g^k\|_2^2 - 2\alpha_k (f(x^k) - f^*), \end{aligned}$$

since $f^* = f(x^*) \geq f(x^k) + \langle g^k, x^* - x^k \rangle$

Apply same argument to $\|x^k - x^*\|_2^2$ recursively

$$\|x^{k+1} - x^*\|_2^2 \leq \|x^0 - x^*\|_2^2 + \sum\nolimits_{t=1}^{k} \alpha_t^2 \|g^k\|_2^2 - 2\sum\nolimits_{t=1}^{k} \alpha_t (f^t - f^*).$$

Now use our convenient assumptions!

Subgradient method – convergence

$$\|x^{k+1} - x^*\|_2^2 \leq R^2 + G^2 \sum_{t=1}^{k} \alpha_t^2 - 2 \sum_{t=1}^{k} \alpha_t (f^t - f^*).$$

- To get a bound on the last term, simply notice (for $t \leq k$)

$$f^t \geq f^t_{\min} \geq f^k_{\min} \qquad \text{since} \quad f^t_{\min} := \min_{0 \leq i \leq t} f(x^i)$$

Subgradient method – convergence

$$\|x^{k+1} - x^*\|_2^2 \leq R^2 + G^2 \sum_{t=1}^{k} \alpha_t^2 - 2\sum_{t=1}^{k} \alpha_t (f^t - f^*).$$

- To get a bound on the last term, simply notice (for $t \leq k$)

$$f^t \geq f^t_{\min} \geq f^k_{\min} \qquad \text{since} \quad f^t_{\min} := \min_{0 \leq i \leq t} f(x^i)$$

- Plugging this in yields the bound

$$2\sum_{t=1}^{k} \alpha_t (f^t - f^*) \geq 2(f^k_{\min} - f^*) \sum_{t=1}^{k} \alpha_t.$$

Subgradient method – convergence

$$\|x^{k+1} - x^*\|_2^2 \le R^2 + G^2 \sum_{t=1}^k \alpha_t^2 - 2\sum_{t=1}^k \alpha_t(f^t - f^*).$$

- To get a bound on the last term, simply notice (for $t \le k$)

$$f^t \ge f^t_{\min} \ge f^k_{\min} \qquad \text{since} \quad f^t_{\min} := \min_{0 \le i \le t} f(x^i)$$

- Plugging this in yields the bound

$$2\sum_{t=1}^k \alpha_t(f^t - f^*) \ge 2(f^k_{\min} - f^*)\sum_{t=1}^k \alpha_t.$$

- So that we finally have

$$0 \le \|x^{k+1} - x^*\|_2 \le R^2 + G^2 \sum_{t=1}^k \alpha_t^2 - 2(f^k_{\min} - f^*)\sum_{t=1}^k \alpha_t$$

Subgradient method – convergence

$$\|x^{k+1} - x^*\|_2^2 \leq R^2 + G^2 \sum_{t=1}^{k} \alpha_t^2 - 2 \sum_{t=1}^{k} \alpha_t (f^t - f^*).$$

- To get a bound on the last term, simply notice (for $t \leq k$)

$$f^t \geq f_{\min}^t \geq f_{\min}^k \qquad \text{since} \quad f_{\min}^t := \min_{0 \leq i \leq t} f(x^i)$$

- Plugging this in yields the bound

$$2 \sum_{t=1}^{k} \alpha_t (f^t - f^*) \geq 2(f_{\min}^k - f^*) \sum_{t=1}^{k} \alpha_t.$$

- So that we finally have

$$0 \leq \|x^{k+1} - x^*\|_2 \leq R^2 + G^2 \sum_{t=1}^{k} \alpha_t^2 - 2(f_{\min}^k - f^*) \sum_{t=1}^{k} \alpha_t$$

$$\boxed{f_{\min}^k - f^* \leq \frac{R^2 + G^2 \sum_{t=1}^{k} \alpha_t^2}{2 \sum_{t=1}^{k} \alpha_t}}$$

Subgradient method – convergence

- Suppose we want $f_{\text{min}}^k - f^* \leq \varepsilon$, how big should k be?

Subgradient method – convergence

- Suppose we want $f_{\min}^k - f^* \leq \varepsilon$, how big should k be?
- Optimize the bound for α_t: want

$$f_{\min}^k - f^* \leq \qquad\qquad\qquad \varepsilon$$

Subgradient method – convergence

- Suppose we want $f_{\min}^k - f^* \le \varepsilon$, how big should k be?
- Optimize the bound for α_t: want

$$f_{\min}^k - f^* \le \frac{R^2 + G^2 \sum_{t=1}^k \alpha_t^2}{2 \sum_{t=1}^k \alpha_t} \quad \le \quad \varepsilon$$

Subgradient method – convergence

- Suppose we want $f_{\min}^k - f^* \leq \varepsilon$, how big should k be?
- Optimize the bound for α_t: want

$$f_{\min}^k - f^* \leq \frac{R^2 + G^2 \sum_{t=1}^k \alpha_t^2}{2\sum_{t=1}^k \alpha_t} \leq \varepsilon$$

- For fixed k: best possible stepsize is constant α

$$\frac{R^2 + G^2 k\alpha^2}{2k\alpha} \leq \epsilon \quad \Rightarrow \quad \alpha = \frac{R}{G\sqrt{k}}$$

- Then, after k steps $f_{\min}^k - f^* \leq RG/\sqrt{k}$.
- For accuracy ϵ, we need at least $(RG/\epsilon)^2 = O(1/\epsilon^2)$ steps
- (quite slow)

Subgradient method – Exercise 1

Support vector machines

- Let $\mathcal{D} := \{(x_i, y_i) \mid x_i \in \mathbb{R}^n, y_i \in \{\pm 1\}\}$
- We wish to find $w \in \mathbb{R}^n$ and $b \in \mathbb{R}$ such that

$$\min_{w,b} \quad \tfrac{1}{2}\|w\|_2^2 + C \sum\nolimits_{i=1}^{m} \max[0, 1 - y_i(w^T x_i + b)]$$

- Derive and implement a subgradient method
- Plot evolution of objective function
- Experiment with different values of $C > 0$
- Plot and keep track of $f_{\min}^k := \min_{0 \le t \le k} f(x^t)$

Exercise 2 – Geometric median

Geometric median / Fermat-Weber

- Let $A \in \mathbb{R}^{m \times n}$ be a matrix
- Let $f(x) = \sum_i \|x - a_i\|_p$
- Implement different subgradient methods to minimize f
- Compare against CVX (interior point)

Exercise 3 – Polyak's stepsize

- Assume f^* is known (or can be estimated). Then use

$$\alpha_t = \frac{f^t - f^*}{\|g^t\|_2^2}$$

Exercise 3 – Polyak's stepsize

- Assume f^* is known (or can be estimated). Then use

$$\alpha_t = \frac{f^t - f^*}{\|g^t\|_2^2}$$

- Motivation: recall bound and minimize RHS:

$$\|x^{t+1} - x^*\|^2 \le \|x^t - x^*\|^2 - 2\alpha_t(f^t - f^*) + \alpha_t^2\|g^t\|^2$$

- Let's plug in α_t:

$$\|x^{t+1} - x^*\|^2 \le \|x^t - x^*\|^2 - \frac{(f^t - f^*)^2}{\|g_t\|^2}$$

Exercise 3 – Polyak's stepsize

$$\|x^{t+1} - x^*\|^2 \leq \|x^t - x^*\|^2 - \frac{(f^t - f^*)^2}{\|g_t\|^2}$$

Exercise 3 – Polyak's stepsize

$$\|x^{t+1} - x^*\|^2 \leq \|x^t - x^*\|^2 - \frac{(f^t - f^*)^2}{\|g_t\|^2}$$

- **Observation 1** $\|x^t - x^*\|$ decreases
- Recursion:

$$\sum_{t=1}^{k} \frac{(f^t - f^*)^2}{\|g_t\|^2} \quad \leq \quad \|x^1 - x^*\|^2 \quad \leq \quad R^2$$

- Now use $\|g^t\| \leq G$

$$\sum_{t=1}^{k} (f^t - f^*)^2 \leq R^2 G^2$$

- **Observation 2** $f^t \to f^*$
- for accuracy ε, need $k = (RG/\varepsilon)^2$

Nonsmooth complexity

Theorem Let $\mathcal{B} = \{x \mid \|x - x^0\|_2 \le D\}$. Assume, $x^* \in \mathcal{B}$. There exists a convex function f in $C_L^0(\mathcal{B})$ (with $L > 0$), such that for $0 \le k \le n-1$, the lower-bound

$$f(x^k) - f(x^*) \ge \frac{LD}{2(1+\sqrt{k+1})},$$

holds for **any algorithm** that generates x^k by combining the previous iterates and subgradients.

(See [Nemirovski-Yudin 1983, Nesterov 2003])

Nonsmooth complexity

Theorem Let $\mathcal{B} = \{x \mid \|x - x^0\|_2 \leq D\}$. Assume, $x^* \in \mathcal{B}$. There exists a convex function f in $C_L^0(\mathcal{B})$ (with $L > 0$), such that for $0 \leq k \leq n-1$, the lower-bound

$$f(x^k) - f(x^*) \geq \frac{LD}{2(1+\sqrt{k+1})},$$

holds for **any algorithm** that generates x^k by combining the previous iterates and subgradients.

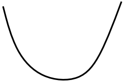
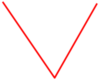
(See [Nemirovski-Yudin 1983, Nesterov 2003])

Can we do better?

Composite objectives

Frequently nonsmooth problems take the form

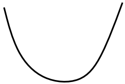
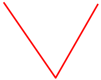
$$\text{minimize } f(x) := \ell(x) + r(x)$$

$\ell \in$  $+ \; r \in$ 

Composite objectives

Frequently nonsmooth problems take the form

$$\text{minimize } f(x) := \ell(x) + r(x)$$

$\ell \in$  $+ \; r \in$ 

Example: $\ell(x) = \frac{1}{2}\|Ax - b\|^2$ and $r(x) = \lambda\|x\|_1$

Lasso, L1-LS, compressed sensing

Example: $\ell(x)$: Logistic loss, and $r(x) = \lambda\|x\|_1$

L1-Logistic regression, sparse LR

Composite objective minimization

minimize $f(x) := \ell(x) + r(x)$

subgradient: $x^{k+1} = x^k - \alpha^k g^k$, $g^k \in \partial f(x^k)$

subgradient: converges slowly at rate $O(1/\sqrt{k})$

but: f is *smooth* plus *nonsmooth*

we should **exploit:** smoothness of ℓ for better method!

Proximal Gradient Method

$$\min \quad f(x) \quad x \in \mathcal{X}$$

Projected gradient

$$x \leftarrow P_{\mathcal{X}}(x - \alpha \nabla f(x))$$

Proximal Gradient Method

$$\min \quad f(x) \quad x \in \mathcal{X}$$

Projected gradient

$$x \leftarrow P_{\mathcal{X}}(x - \alpha \nabla f(x))$$

$$\min \quad f(x) + h(x)$$

Proximal gradient

$$x \leftarrow \mathrm{prox}_{\alpha h}(x - \alpha \nabla f(x))$$

$\mathrm{prox}_{\alpha h}$ denotes **Euclidean** proximity operator for h

Proximity operator

Projection

$$P_{\mathcal{X}}(y) := \underset{x \in \mathbb{R}^n}{\operatorname{argmin}} \quad \tfrac{1}{2}\|x - y\|_2^2 + \mathbb{1}_{\mathcal{X}}(x)$$

Proximity operator

Projection

$$P_{\mathcal{X}}(y) := \underset{x\in\mathbb{R}^n}{\operatorname{argmin}} \quad \tfrac{1}{2}\|x-y\|_2^2 + \mathbb{1}_{\mathcal{X}}(x)$$

Proximity: Replace $\mathbb{1}_{\mathcal{X}}$ by a closed convex function

$$\operatorname{prox}_r(y) := \underset{x\in\mathbb{R}^n}{\operatorname{argmin}} \quad \tfrac{1}{2}\|x-y\|_2^2 + r(x)$$

Prox operators – Exercise 1

Example: Let $r(x) = \|x\|_1$. Solve $\text{prox}_{\lambda r}(y)$.

$$\min_{x \in \mathbb{R}^n} \quad \tfrac{1}{2}\|x - y\|_2^2 + \lambda\|x\|_1.$$

Hint 1: The above problem decomposes into n independent subproblems of the form

$$\min_{x \in \mathbb{R}} \quad \tfrac{1}{2}(x - y)^2 + \lambda|x|.$$

Hint 2: Consider the two cases: either $x = 0$ or $x \neq 0$

Aka: Soft-thresholding operator

Prox operators – Exercise 2

Moreau Decomposition

- **Aim:** Compute $\mathrm{prox}_r\, y$
- Sometimes it is easier to compute $\mathrm{prox}_{r^*}\, y$

$$r^*(u) := \sup_x u^T x - r(x)$$

- Moreau decomposition: $y = \mathrm{prox}_r\, y + \mathrm{prox}_{r^*}\, y$

(Hint: Consider $\min \frac{1}{2}\|x - y\|_2^2 + r(x)$; introduce $z = x$; duality)

Prox operators – Challenge

Inf-norm prox: Develop an $O(n)$ algorithm to solve

$$\min \quad \tfrac{1}{2}\|x - y\|^2 + \lambda\|x\|_\infty$$

L1-TV: Develop an $O(n)$ algorithm to solve

$$\min \quad \tfrac{1}{2}\|x - y\|^2 + \lambda \sum_{i=1}^{n-1} |x_i - x_{i+1}|$$

Prox operators – Explore

- Let $(\mathcal{X}, d)$ be a reasonable metric space.
- Study the generalized prox operator

$$\min_{x \in \mathcal{X}} \tfrac{1}{2} d^2(x, y) + \lambda r(x).$$

(Example: consider vector spaces, manifolds, etc.)

Where does it come from?

$$\min f(x) + h(x)$$

Lemma $x^* = \operatorname{prox}_{\alpha h}(x^* - \alpha \nabla f(x^*)), \forall \alpha > 0$

Where does it come from?

$$\min f(x) + h(x)$$

Lemma $x^* = \text{prox}_{\alpha h}(x^* - \alpha \nabla f(x^*)), \forall \alpha > 0$

$$0 \in \nabla f(x^*) + \partial h(x^*)$$

Where does it come from?

$$\min f(x) + h(x)$$

Lemma $x^* = \text{prox}_{\alpha h}(x^* - \alpha \nabla f(x^*)), \forall \alpha > 0$

$$\begin{aligned} 0 &\in \nabla f(x^*) + \partial h(x^*) \\ 0 &\in \alpha \nabla f(x^*) + \alpha \partial h(x^*) \end{aligned}$$

Where does it come from?

$$\min f(x) + h(x)$$

Lemma $x^* = \text{prox}_{\alpha h}(x^* - \alpha \nabla f(x^*)), \forall \alpha > 0$

$$\begin{aligned} 0 &\in \nabla f(x^*) + \partial h(x^*) \\ 0 &\in \alpha \nabla f(x^*) + \alpha \partial h(x^*) \\ x^* &\in \alpha \nabla f(x^*) + (I + \alpha \partial h)(x^*) \end{aligned}$$

Where does it come from?

$$\min f(x) + h(x)$$

Lemma $x^* = \text{prox}_{\alpha h}(x^* - \alpha \nabla f(x^*)), \forall \alpha > 0$

$$\begin{aligned}
0 &\in \nabla f(x^*) + \partial h(x^*) \\
0 &\in \alpha \nabla f(x^*) + \alpha \partial h(x^*) \\
x^* &\in \alpha \nabla f(x^*) + (I + \alpha \partial h)(x^*) \\
x^* - \alpha \nabla f(x^*) &\in (I + \alpha \partial h)(x^*)
\end{aligned}$$

Where does it come from?

$$\min f(x) + h(x)$$

Lemma $x^* = \text{prox}_{\alpha h}(x^* - \alpha \nabla f(x^*)), \forall \alpha > 0$

$$\begin{aligned}
0 &\in \nabla f(x^*) + \partial h(x^*) \\
0 &\in \alpha \nabla f(x^*) + \alpha \partial h(x^*) \\
x^* &\in \alpha \nabla f(x^*) + (I + \alpha \partial h)(x^*) \\
x^* - \alpha \nabla f(x^*) &\in (I + \alpha \partial h)(x^*) \\
x^* &= (I + \alpha \partial h)^{-1}(x^* - \alpha \nabla f(x^*))
\end{aligned}$$

Where does it come from?

$$\min f(x) + h(x)$$

Lemma $x^* = \text{prox}_{\alpha h}(x^* - \alpha \nabla f(x^*)), \forall \alpha > 0$

$$\begin{aligned}
0 &\in \nabla f(x^*) + \partial h(x^*) \\
0 &\in \alpha \nabla f(x^*) + \alpha \partial h(x^*) \\
x^* &\in \alpha \nabla f(x^*) + (I + \alpha \partial h)(x^*) \\
x^* - \alpha \nabla f(x^*) &\in (I + \alpha \partial h)(x^*) \\
x^* &= (I + \alpha \partial h)^{-1}(x^* - \alpha \nabla f(x^*)) \\
x^* &= \text{prox}_{\alpha h}(x^* - \alpha \nabla f(x^*))
\end{aligned}$$

Where does it come from?

$$\min f(x) + h(x)$$

Lemma $x^* = \text{prox}_{\alpha h}(x^* - \alpha \nabla f(x^*)), \forall \alpha > 0$

$$\begin{aligned}
0 &\in \nabla f(x^*) + \partial h(x^*) \\
0 &\in \alpha \nabla f(x^*) + \alpha \partial h(x^*) \\
x^* &\in \alpha \nabla f(x^*) + (I + \alpha \partial h)(x^*) \\
x^* - \alpha \nabla f(x^*) &\in (I + \alpha \partial h)(x^*) \\
x^* &= (I + \alpha \partial h)^{-1}(x^* - \alpha \nabla f(x^*)) \\
x^* &= \text{prox}_{\alpha h}(x^* - \alpha \nabla f(x^*))
\end{aligned}$$

Above fixed-point eqn suggests iteration

$$x_{k+1} = \text{prox}_{\alpha_k h}(x_k - \alpha_k \nabla f(x_k))$$

Why does it work?

$$\begin{aligned} x_{k+1} &= \mathrm{prox}_{\alpha_k h}(x_k - \alpha_k \nabla f(x_k)) \\ x_{k+1} &= x_k - \alpha_k G_{\alpha_k}(x_k). \end{aligned}$$

Why does it work?

$$x_{k+1} = \operatorname{prox}_{\alpha_k h}(x_k - \alpha_k \nabla f(x_k))$$
$$x_{k+1} = x_k - \alpha_k G_{\alpha_k}(x_k).$$

Gradient mapping: the "gradient-like object"

$$G_\alpha(x) = \frac{1}{\alpha}(x - P_{\alpha h}(x - \alpha \nabla f(x)))$$

Mimic proof of $x \leftarrow x - \alpha \nabla f(x)$

- Our lemma shows: $G_\alpha(x) = 0$ if and only if x is optimal
- So G_α analogous to ∇f
- If x locally optimal, then $G_\alpha(x) = 0$ (nonconvex f)
- Analysis yields $O(1/k)$ convergence

Convergence analysis: descent

$$f(y) \le f(x) + \langle \nabla f(x), y - x \rangle + \tfrac{L}{2}\|y - x\|_2^2$$

Let $y = x - \alpha G_\alpha(x)$, then

$$f(y) \le f(x) - \alpha\langle \nabla f(x), G_\alpha(x)\rangle + \frac{\alpha^2 L}{2}\|G_\alpha(x)\|_2^2.$$

Corollary. So if $0 \le \alpha \le 1/L$, we have

$$f(y) \le f(x) - \alpha\langle \nabla f(x), G_\alpha(x)\rangle + \frac{\alpha}{2}\|G_\alpha(x)\|_2^2.$$

Lemma Let $y = x - \alpha G_\alpha(x)$. Then, for any z we have

$$f(y) + h(y) \le f(z) + h(z) + \langle G_\alpha(x), x - z\rangle - \tfrac{\alpha}{2}\|G_\alpha(x)\|_2^2.$$

Exercise: Prove! (hint: f, h are convex, $G_\alpha(x) - \nabla f(x) \in \partial h(y)$)

Convergence analysis

We've actually shown $x' = x - \alpha G_\alpha(x)$ is a descent method.
Write $\phi = f + h$; plug in $z = x$ to obtain

$$\phi(x') \leq \phi(x) - \tfrac{\alpha}{2}\|G_\alpha(x)\|_2^2.$$

Exercise: Why this inequality suffices to show convergence.
Use $z = x^*$ in corollary to obtain progress in terms of iterates:

$$\begin{aligned}
\phi(x') - \phi^* &\leq \langle G_\alpha(x), x - x^*\rangle - \tfrac{\alpha}{2}\|G_\alpha(x)\|_2^2 \\
&= \frac{1}{2\alpha}\left[2\langle \alpha G_\alpha(x), x - x^*\rangle - \|\alpha G_\alpha(x)\|_2^2\right] \\
&= \frac{1}{2\alpha}\left[\|x - x^*\|_2^2 - \|x - x^* - \alpha G_\alpha(x)\|_2^2\right] \\
&= \frac{1}{2\alpha}\left[\|x - x^*\|_2^2 - \|x' - x^*\|_2^2\right].
\end{aligned}$$

Convergence rate

Set $x \leftarrow x_k$, $x' \leftarrow x_{k+1}$, and $\alpha = 1/L$. Then add

$$\begin{aligned}
\sum_{i=1}^{k+1}(\phi(x_i) - \phi^*) &\le \tfrac{L}{2}\sum_{i=1}^{k+1}\left[\|x_i - x^*\|_2^2 - \|x_{i+1} - x^*\|_2^2\right] \\
&= \tfrac{L}{2}\left[\|x_1 - x^*\|_2^2 - \|x_{k+1} - x^*\|_2^2\right] \\
&\le \tfrac{L}{2}\|x_1 - x^*\|_2^2.
\end{aligned}$$

Since $\phi(x_k)$ is a decreasing sequence, it follows that

$$\phi(x_{k+1}) - \phi^* \le \frac{1}{k+1}\sum_{i=1}^{k+1}(\phi(x_i) - \phi^*) \le \frac{L}{2(k+1)}\|x_1 - x^*\|_2^2.$$

This is the well-known $O(1/k)$ rate.
But for C_L^1 convex functions, optimal rate is $O(1/k^2)$

Faster methods*

Optimal gradient methods

♠ Efficiency estimates for the gradient method:

$$f \in C_L^1: \qquad f(x^k) - f^* \leq \frac{2L\|x^0 - x^*\|_2^2}{k+4}$$

$$f \in S_{L,\mu}^1: \qquad f(x^k) - f^* \leq \tfrac{L}{2}\left(\frac{L-\mu}{L+\mu}\right)^{2k}\|x^0 - x^*\|_2^2.$$

Optimal gradient methods

♠ Efficiency estimates for the gradient method:

$$f \in C_L^1 : \qquad f(x^k) - f^* \leq \frac{2L\|x^0 - x^*\|_2^2}{k+4}$$

$$f \in S_{L,\mu}^1 : \qquad f(x^k) - f^* \leq \tfrac{L}{2}\left(\frac{L-\mu}{L+\mu}\right)^{2k} \|x^0 - x^*\|_2^2.$$

♠ **Lower complexity bounds**:

$$f \in C_L^1 : \qquad f(x^k) - f(x^*) \geq \frac{3L\|x^0 - x^*\|_2^2}{32(k+1)^2}$$

$$fS_{L,\mu}^\infty : \qquad f(x^k) - f(x^*) \geq \frac{\mu}{2}\left(\frac{\sqrt{L}-\sqrt{\mu}}{\sqrt{L}+\sqrt{\mu}}\right)^{2k} \|x^0 - x^*\|_2^2.$$

Optimal gradient methods

♠ Subgradient method upper and lower bounds

$$f(x^k) - f(x^*) \le O(1/\sqrt{k})$$
$$f(x^k) - f(x^*) \ge \frac{LD}{2(1+\sqrt{k+1})}.$$

♠ Composite objective problems: proximal gradient gives same bounds as gradient methods.

Optimal Proximal Gradient

$$\min \phi(x) = f(x) + h(x)$$

Let $x^0 = y^0 \in \operatorname{dom} h$. For $k \geq 1$:

$$x^k = \operatorname{prox}_{\alpha_k h}(y^{k-1} - \alpha_k \nabla f(y^{k-1}))$$
$$y^k = x_k + \frac{k-1}{k+2}(x^k - x^{k-1}).$$

Framework due to: Nesterov (1983, 2004); also Beck, Teboulle (2009).

Simplified analysis: Tseng (2008).

Optimal Proximal Gradient

$$\min \phi(x) = f(x) + h(x)$$

Let $x^0 = y^0 \in \operatorname{dom} h$. For $k \geq 1$:

$$x^k = \operatorname{prox}_{\alpha_k h}(y^{k-1} - \alpha_k \nabla f(y^{k-1}))$$

$$y^k = x_k + \frac{k-1}{k+2}(x^k - x^{k-1}).$$

Framework due to: Nesterov (1983, 2004); also Beck, Teboulle (2009). Simplified analysis: Tseng (2008).

- Uses extra “memory” for interpolation
- Same computational cost as ordinary prox-grad
- Convergence rate theoretically optimal

$$\phi(x^k) - \phi^* \leq \frac{2L}{(k+1)^2} \|x^0 - x^*\|_2^2.$$

The operator view

Set-valued mappings

Think of ∂f as a **set-valued map**

$$\partial f = x \rightrightarrows \partial f(x).$$

Set-valued mappings

Think of ∂f as a **set-valued map**

$$\partial f = x \rightrightarrows \partial f(x).$$

Relation R is a subset of $\mathbb{R}^n \times \mathbb{R}^n$

Set-valued mappings

Think of ∂f as a **set-valued map**

$$\partial f = x \rightrightarrows \partial f(x).$$

Relation R is a subset of $\mathbb{R}^n \times \mathbb{R}^n$

- **Empty relation**: $\emptyset$
- **Identity**: $I := \{(x, x) \mid x \in \mathbb{R}^n\}$
- **Zero**: $0 := \{(x, 0) \mid x \in \mathbb{R}^n\}$
- **Subdifferential**: $\partial f := \{(x, g) \mid x \in \mathbb{R}^n, g \in \partial f(x)\}$
- We will write $R(x)$ to mean $\{y \mid (x, y) \in R\}$.
- Example: $\partial f(x) = \{g \mid (x, g) \in \partial f\}$

Why this notation?

- **Goal:** solve *generalized equation* $0 \in R(x)$
- That is, find $x \in \mathbb{R}^n$ such that $(x, 0) \in R$
- **Example:** Say $R \equiv \partial f$, then goal

$$0 \in R(x) \Leftrightarrow 0 \in \partial f(x),$$

  means we want to find an x that minimizes f.
- Helps succinctly write / analyze problems and algorithms

Which operators are “easier”?

Def. The set valued operator $R \subset \mathbb{R}^n \times \mathbb{R}^n$ is called **monotone** if

$$\langle R(x) - R(y), x - y \rangle \geq 0, \quad x, y \in \mathbb{R}^n.$$

Examples:

- Any positive semidefinite matrix $\langle Ax - Ay, x - y \rangle \geq 0$

Which operators are "easier"?

Def. The set valued operator $R \subset \mathbb{R}^n \times \mathbb{R}^n$ is called **monotone** if

$$\langle R(x) - R(y), x - y \rangle \geq 0, \quad x, y \in \mathbb{R}^n.$$

Examples:

- Any positive semidefinite matrix $\langle Ax - Ay, x - y \rangle \geq 0$
- The subdifferential ∂f of a convex function (verify!)

Which operators are "easier"?

Def. The set valued operator $R \subset \mathbb{R}^n \times \mathbb{R}^n$ is called **monotone** if

$$\langle R(x) - R(y), x - y \rangle \geq 0, \quad x, y \in \mathbb{R}^n.$$

Examples:

- Any positive semidefinite matrix $\langle Ax - Ay, x - y \rangle \geq 0$
- The subdifferential ∂f of a convex function (verify!)
- Any monotonically nondecreasing function $T : \mathbb{R} \to \mathbb{R}$

Which operators are "easier"?

Def. The set valued operator $R \subset \mathbb{R}^n \times \mathbb{R}^n$ is called **monotone** if

$$\langle R(x) - R(y), x - y \rangle \geq 0, \quad x, y \in \mathbb{R}^n.$$

Examples:

- Any positive semidefinite matrix $\langle Ax - Ay, x - y \rangle \geq 0$
- The subdifferential ∂f of a convex function (verify!)
- Any monotonically nondecreasing function $T : \mathbb{R} \to \mathbb{R}$
- Projection and proximity operators

Which operators are "easier"?

Def. The set valued operator $R \subset \mathbb{R}^n \times \mathbb{R}^n$ is called **monotone** if

$$\langle R(x) - R(y), x - y \rangle \geq 0, \quad x, y \in \mathbb{R}^n.$$

Examples:

- Any positive semidefinite matrix $\langle Ax - Ay, x - y \rangle \geq 0$
- The subdifferential ∂f of a convex function (verify!)
- Any monotonically nondecreasing function $T : \mathbb{R} \to \mathbb{R}$
- Projection and proximity operators

Generalize notion of monotonicity to vectors

♠ Abstraction takes linear-algebra intuition to optimization

Importance of resolvent operators

Aim: solve generalized equation

$$0 \in R(x)$$

Importance of resolvent operators

Aim: solve generalized equation

$$0 \in R(x)$$

Theorem The solutions to the generalized equation coincide with points that satisfy the **resolvent equation** $x = (I + \alpha R)^{-1}(x)$

Importance of resolvent operators

Aim: solve generalized equation

$$0 \in R(x)$$

Theorem The solutions to the generalized equation coincide with points that satisfy the **resolvent equation** $x = (I + \alpha R)^{-1}(x)$

Proof:

$0 \in R(x)$

Importance of resolvent operators

Aim: solve generalized equation

$$0 \in R(x)$$

Theorem The solutions to the generalized equation coincide with points that satisfy the **resolvent equation** $x = (I + \alpha R)^{-1}(x)$

Proof:

$0 \in R(x) \leftrightarrow 0 \in \alpha R(x)$

Importance of resolvent operators

Aim: solve generalized equation

$$0 \in R(x)$$

Theorem The solutions to the generalized equation coincide with points that satisfy the **resolvent equation** $x = (I + \alpha R)^{-1}(x)$

Proof:

$$0 \in R(x) \leftrightarrow 0 \in \alpha R(x) \leftrightarrow x \in (I + \alpha R)(x)$$

Importance of resolvent operators

Aim: solve generalized equation

$$0 \in R(x)$$

Theorem The solutions to the generalized equation coincide with points that satisfy the **resolvent equation** $x = (I + \alpha R)^{-1}(x)$

Proof:

$$0 \in R(x) \leftrightarrow 0 \in \alpha R(x) \leftrightarrow x \in (I + \alpha R)(x) \leftrightarrow x = (I + \alpha R)^{-1}(x)$$

Rederiving proximal-gradient

Theorem Let h be a closed convex function, and $\lambda > 0$, then

$$(I + \lambda \partial h)^{-1}(y) = \mathrm{prox}_{\lambda h}(y).$$

Rederiving proximal-gradient

Theorem Let h be a closed convex function, and $\lambda > 0$, then

$$(I + \lambda\partial h)^{-1}(y) = \mathrm{prox}_{\lambda h}(y).$$

- Suppose $(I + \lambda\partial h)^{-1}$ is single valued
- Then, $x = (I + \lambda\partial h)^{-1}(y) \implies y \in (I + \lambda\partial h)(x)$
- That is, $y \in x + \lambda\partial h(x)$
- Equivalently, $x - y + \lambda\partial h(x) \ni 0$

Rederiving proximal-gradient

Theorem Let h be a closed convex function, and $\lambda > 0$, then

$$(I + \lambda\partial h)^{-1}(y) = \text{prox}_{\lambda h}(y).$$

- Suppose $(I + \lambda\partial h)^{-1}$ is single valued
- Then, $x = (I + \lambda\partial h)^{-1}(y) \implies y \in (I + \lambda\partial h)(x)$
- That is, $y \in x + \lambda\partial h(x)$
- Equivalently, $x - y + \lambda\partial h(x) \ni 0$
- Nothing other than optimality condition for prox-operator

$$\text{prox}_{\lambda h}(y) \equiv y \mapsto \operatorname*{argmin}_x \ \tfrac{1}{2}\|x - y\|_2^2 + \lambda h(x)$$

More proximal splitting

$$\ell(x) + f(x) + h(x)$$

- Direct use of prox-grad not easy
- Requires computation of: $\text{prox}_{\lambda(f+h)}$ (i.e., $(I + \lambda(\partial f + \partial h))^{-1}$)

More proximal splitting

$$\ell(x) + f(x) + h(x)$$

- Direct use of prox-grad not easy
- Requires computation of: $\text{prox}_{\lambda(f+h)}$ (i.e., $(I + \lambda(\partial f + \partial h))^{-1}$)

Example:

$$\min \quad \tfrac{1}{2}\|x - y\|_2^2 + \underbrace{\lambda\|x\|_2}_{f(x)} + \underbrace{\mu \sum\nolimits_{i=1}^{n-1} |x_{i+1} - x_i|}_{h(x)}.$$

More proximal splitting

$$\ell(x) + f(x) + h(x)$$

- Direct use of prox-grad not easy
- Requires computation of: $\text{prox}_{\lambda(f+h)}$ (i.e., $(I + \lambda(\partial f + \partial h))^{-1}$)

Example:

$$\min \quad \tfrac{1}{2}\|x - y\|_2^2 + \underbrace{\lambda\|x\|_2}_{f(x)} + \underbrace{\mu \sum\nolimits_{i=1}^{n-1} |x_{i+1} - x_i|}_{h(x)}.$$

- But good feature: prox_f and prox_h separately easier
- Can we exploit that?

Proximal splitting – operator notation

- If $(I + \partial f + \partial h)^{-1}$ hard, but $(I + \partial f)^{-1}$ and $(I + \partial h)^{-1}$ "easy"

Proximal splitting – operator notation

- If $(I + \partial f + \partial h)^{-1}$ hard, but $(I + \partial f)^{-1}$ and $(I + \partial h)^{-1}$ "easy"
- Let us derive a fixed-point equation that "splits" the operators

Proximal splitting – operator notation

- If $(I + \partial f + \partial h)^{-1}$ hard, but $(I + \partial f)^{-1}$ and $(I + \partial h)^{-1}$ "easy"
- Let us derive a fixed-point equation that "splits" the operators

Assume we are solving

$$\min \quad f(x) + h(x),$$

where both f and h are convex but potentially nondifferentiable.

Notice: We implicitly assumed: $\partial(f + h) = \partial f + \partial h$.

Proximal splitting

$$0 \in \partial f(x) + \partial h(x)$$

Proximal splitting

$$\begin{aligned} 0 &\in \partial f(x) + \partial h(x) \\ 2x &\in (I + \partial f)(x) + (I + \partial h)(x) \end{aligned}$$

Proximal splitting

$$
\begin{aligned}
0 &\in \partial f(x) + \partial h(x) \\
2x &\in (I + \partial f)(x) + (I + \partial h)(x)
\end{aligned}
$$

Key idea of splitting: new variable!

$$z \in (I + \partial h)(x) \implies x = \text{prox}_h(z)$$

Proximal splitting

$$\begin{aligned} 0 &\in \partial f(x) + \partial h(x) \\ 2x &\in (I + \partial f)(x) + (I + \partial h)(x) \end{aligned}$$

Key idea of splitting: new variable!

$$z \in (I + \partial h)(x) \implies x = \text{prox}_h(z)$$

$$2x - z \in (I + \partial f)(x)$$

Proximal splitting

$$\begin{aligned} 0 &\in \partial f(x) + \partial h(x) \\ 2x &\in (I + \partial f)(x) + (I + \partial h)(x) \end{aligned}$$

Key idea of splitting: new variable!

$$z \in (I + \partial h)(x) \implies x = \text{prox}_h(z)$$

$$2x - z \in (I + \partial f)(x) \implies x \in (I + \partial f)^{-1}(2x - z)$$

Proximal splitting

$$\begin{aligned} 0 &\in \partial f(x) + \partial h(x) \\ 2x &\in (I + \partial f)(x) + (I + \partial h)(x) \end{aligned}$$

Key idea of splitting: new variable!

$$z \in (I + \partial h)(x) \implies x = \mathrm{prox}_h(z)$$

$$2x - z \in (I + \partial f)(x) \implies x \in (I + \partial f)^{-1}(2x - z)$$

- Not a fixed-point equation yet

Proximal splitting

$$\begin{aligned} 0 &\in \partial f(x) + \partial h(x) \\ 2x &\in (I + \partial f)(x) + (I + \partial h)(x) \end{aligned}$$

Key idea of splitting: new variable!

$$z \in (I + \partial h)(x) \implies x = \mathrm{prox}_h(z)$$

$$2x - z \in (I + \partial f)(x) \implies x \in (I + \partial f)^{-1}(2x - z)$$

- Not a fixed-point equation yet
- We need one more idea

Douglas-Rachford splitting

Reflection operator

$$R_h(z) := 2\,\text{prox}_h(z) - z$$

Douglas-Rachford splitting

Reflection operator

$$R_h(z) := 2\,\mathrm{prox}_h(z) - z$$

Douglas-Rachford method

$$z \in (I + \partial h)(x), \quad x = \mathrm{prox}_h(z)$$

Douglas-Rachford splitting

Reflection operator

$$R_h(z) := 2\,\mathrm{prox}_h(z) - z$$

Douglas-Rachford method

$$z \in (I + \partial h)(x), \quad x = \mathrm{prox}_h(z) \implies R_h(z) = 2x - z$$

Douglas-Rachford splitting

Reflection operator

$$R_h(z) := 2\,\mathrm{prox}_h(z) - z$$

Douglas-Rachford method

$$z \in (I + \partial h)(x), \quad x = \mathrm{prox}_h(z) \implies R_h(z) = 2x - z$$

$$\begin{aligned} 0 &\in \partial f(x) + \partial g(x) \\ 2x &\in (I + \partial f)(x) + (I + \partial g)(x) \\ 2x - z &\in (I + \partial f)(x) \end{aligned}$$

Douglas-Rachford splitting

Reflection operator

$$R_h(z) := 2\,\mathrm{prox}_h(z) - z$$

Douglas-Rachford method

$$z \in (I + \partial h)(x), \quad x = \mathrm{prox}_h(z) \implies R_h(z) = 2x - z$$

$$\begin{aligned} 0 &\in \partial f(x) + \partial g(x) \\ 2x &\in (I + \partial f)(x) + (I + \partial g)(x) \\ 2x - z &\in (I + \partial f)(x) \\ x &= \mathrm{prox}_f(R_h(z)) \end{aligned}$$

Douglas-Rachford splitting

Reflection operator

$$R_h(z) := 2\,\mathrm{prox}_h(z) - z$$

Douglas-Rachford method

$$z \in (I + \partial h)(x), \quad x = \mathrm{prox}_h(z) \implies R_h(z) = 2x - z$$

$$\begin{aligned}
0 &\in \partial f(x) + \partial g(x) \\
2x &\in (I + \partial f)(x) + (I + \partial g)(x) \\
2x - z &\in (I + \partial f)(x) \\
x &= \mathrm{prox}_f(R_h(z)) \\
\text{but } R_h(z) &= 2x - z \implies \\
z &= 2x - R_h(z)
\end{aligned}$$

Douglas-Rachford splitting

Reflection operator

$$R_h(z) := 2\,\mathrm{prox}_h(z) - z$$

Douglas-Rachford method

$$z \in (I + \partial h)(x), \quad x = \mathrm{prox}_h(z) \implies R_h(z) = 2x - z$$

$$\begin{aligned}
0 &\in \partial f(x) + \partial g(x) \\
2x &\in (I + \partial f)(x) + (I + \partial g)(x) \\
2x - z &\in (I + \partial f)(x) \\
x &= \mathrm{prox}_f(R_h(z)) \\
\text{but } R_h(z) &= 2x - z \implies \\
z &= 2x - R_h(z) \\
z &= 2\,\mathrm{prox}_f(R_h(z)) - R_h(z) =
\end{aligned}$$

Douglas-Rachford splitting

Reflection operator

$$R_h(z) := 2\,\mathrm{prox}_h(z) - z$$

Douglas-Rachford method

$$z \in (I + \partial h)(x), \quad x = \mathrm{prox}_h(z) \implies R_h(z) = 2x - z$$

$$\begin{aligned}
0 &\in \partial f(x) + \partial g(x) \\
2x &\in (I + \partial f)(x) + (I + \partial g)(x) \\
2x - z &\in (I + \partial f)(x) \\
x &= \mathrm{prox}_f(R_h(z)) \\
\text{but } R_h(z) &= 2x - z \implies \\
z &= 2x - R_h(z) \\
z &= 2\,\mathrm{prox}_f(R_h(z)) - R_h(z) = R_f(R_h(z))
\end{aligned}$$

Finally, z is on both sides of the eqn

Douglas-Rachford method

$$0 \in \partial f(x) + \partial h(x) \Leftrightarrow \begin{cases} x = \text{prox}_h(z) \\ z = R_f(R_h(z)) \end{cases}$$

DR method: given z_0, iterate for $k \geq 0$

$$\begin{aligned} x_k &= \text{prox}_h(z_k) \\ v_k &= \text{prox}_f(2x_k - z_k) \\ z_{k+1} &= z_k + \gamma_k(v_k - x_k) \end{aligned}$$

Douglas-Rachford method

$$0 \in \partial f(x) + \partial h(x) \Leftrightarrow \begin{cases} x = \text{prox}_h(z) \\ z = R_f(R_h(z)) \end{cases}$$

DR method: given z_0, iterate for $k \geq 0$

$$\begin{aligned} x_k &= \text{prox}_h(z_k) \\ v_k &= \text{prox}_f(2x_k - z_k) \\ z_{k+1} &= z_k + \gamma_k(v_k - x_k) \end{aligned}$$

Theorem If $f + h$ admits minimizers, and (γ_k) satisfy

$$\gamma_k \in [0, 2], \qquad \sum_k \gamma_k(2 - \gamma_k) = \infty,$$

then the DR-iterates v_k and x_k converge to a minimizer.

Douglas-Rachford method

For $\gamma_k = 1$, we have

$$z_{k+1} = z_k + v_k - x_k$$
$$z_{k+1} = z_k + \text{prox}_f(2\,\text{prox}_h(z_k) - z_k) - \text{prox}_h(z_k)$$

Douglas-Rachford method

For $\gamma_k = 1$, we have

$$z_{k+1} = z_k + v_k - x_k$$
$$z_{k+1} = z_k + \mathrm{prox}_f(2\,\mathrm{prox}_h(z_k) - z_k) - \mathrm{prox}_h(z_k)$$

Dropping superscripts, writing $P \equiv \mathrm{prox}$, we have

$$z \leftarrow Tz$$
$$T = I + P_f(2P_h - I) - P_h$$

Douglas-Rachford method

For $\gamma_k = 1$, we have

$$z_{k+1} = z_k + v_k - x_k$$
$$z_{k+1} = z_k + \mathrm{prox}_f(2\,\mathrm{prox}_h(z_k) - z_k) - \mathrm{prox}_h(z_k)$$

Dropping superscripts, writing $P \equiv \mathrm{prox}$, we have

$$z \leftarrow Tz$$
$$T = I + P_f(2P_h - I) - P_h$$

Lemma DR can be written as: $z \leftarrow \frac{1}{2}(R_f R_h + I)z$, where R_f denotes the *reflection operator* $2P_f - I$ (similarly R_h).

Challenge

Develop generalization of DR to 3 functions.

Partial solutions: Borwein 2013; see this webpage!

Other methods

- ADMM (DR on dual: **nontrivial theorem**)
- Proximal-Dykstra
- Proximal methods for $f_1 + f_2 + \cdots + f_n$
- Peaceman-Rachford
- Proximal quasi-Newton, Newton
- Nonconvex proximal methods
- ...

Large-scale problems

(Bonus material)

Large-scale ML

Regularized Empirical Risk Minimization

$$\min_{w} \quad \frac{1}{n}\sum_{i=1}^{n} \ell(y_i, w^T x_i) + \lambda r(w).$$

This is the $f(w) + r(w)$ "composite objective" form we saw.

(e.g., regression, logistic regression, lasso, CRFs, etc.)

Large-scale ML

Regularized Empirical Risk Minimization

$$\min_w \quad \frac{1}{n}\sum_{i=1}^{n} \ell(y_i, w^T x_i) + \lambda r(w).$$

This is the $f(w) + r(w)$ "composite objective" form we saw.
(e.g., regression, logistic regression, lasso, CRFs, etc.)

- training data: $(x_i, y_i) \in \mathbb{R}^d \times \mathcal{Y}$ (i.i.d.)
- large-scale ML: Both d and n are large:
  - d: dimension of each input sample
  - n: number of training data points / samples
- Assume training data "sparse"; so total datasize $\ll dn$.
- Running time $O(\#\text{nnz})$

Regularized Risk Minimization

Empirical: $\widehat{F}(w) = \frac{1}{n}\sum_{i=1}^{n} \ell(y_i, w^T x_i) + \lambda r(w)$

Generalization: $F(w) = \mathbb{E}_{(x,y)}[\ell(y, w^T x)] + \lambda r(w)$

Regularized Risk Minimization

Empirical: $\widehat{F}(w) = \frac{1}{n}\sum_{i=1}^{n} \ell(y_i, w^T x_i) + \lambda r(w)$

Generalization: $F(w) = \mathbb{E}_{(x,y)}[\ell(y, w^T x)] + \lambda r(w)$

Single pass through data for $F(w)$ by sampling n iid points

Multiple passes if only minimizing empirical cost $\widehat{F}(w)$

Stochastic optimization

$$\min_{x\in\mathcal{X}} F(x) := \mathbb{E}_\xi[f(x,\xi)]$$
(f: loss; x: parameters; ξ: data samples)

Setup

1. $\mathcal{X} \subset \mathbb{R}^d$ compact convex set

Stochastic optimization

$$\min_{x \in \mathcal{X}} F(x) := \mathbb{E}_\xi[f(x, \xi)]$$
(f: loss; x: parameters; ξ: data samples)

Setup

1. $\mathcal{X} \subset \mathbb{R}^d$ compact convex set
2. ξ r.v. with distribution P on $\Omega \subset \mathbb{R}^d$

Stochastic optimization

$$\min_{x \in \mathcal{X}} F(x) := \mathbb{E}_\xi[f(x, \xi)]$$
(f: loss; x: parameters; ξ: data samples)

Setup

1. $\mathcal{X} \subset \mathbb{R}^d$ compact convex set
2. ξ r.v. with distribution P on $\Omega \subset \mathbb{R}^d$
3. The expectation

$$\mathbb{E}_\xi[f(x, \xi)] = \int_\Omega f(x, \xi) dP(\xi)$$

is well-defined and finite valued for every $x \in \mathcal{X}$.

Stochastic optimization

$$\min_{x \in \mathcal{X}} F(x) := \mathbb{E}_\xi[f(x, \xi)]$$

(f: loss; x: parameters; ξ: data samples)

Setup

1. $\mathcal{X} \subset \mathbb{R}^d$ compact convex set

2. ξ r.v. with distribution P on $\Omega \subset \mathbb{R}^d$

3. The expectation

$$\mathbb{E}_\xi[f(x, \xi)] = \int_\Omega f(x, \xi) dP(\xi)$$

is well-defined and finite valued for every $x \in \mathcal{X}$.

4. For every $\xi \in \Omega$, $f(\cdot, \xi)$ is convex

Stochastic optimization

Assumption 1: Possible to generate iid samples $\xi_1, \xi_2, \ldots$
Assumption 2: Oracle yields stochastic gradient $g(x, \xi)$, i.e.,

$$G(x) := \mathbb{E}[g(x, \xi)] \quad \text{s.t.} \quad G(x) \in \partial F(x).$$

Stochastic optimization

Assumption 1: Possible to generate iid samples $\xi_1, \xi_2, \ldots$
Assumption 2: Oracle yields stochastic gradient $g(x, \xi)$, i.e.,

$$G(x) := \mathbb{E}[g(x, \xi)] \quad \text{s.t.} \quad G(x) \in \partial F(x).$$

Theorem Let $\xi \in \Omega$; If $f(\cdot, \xi)$ is convex, and $F(\cdot)$ is finite valued in a neighborhood of x, then

$$\partial F(x) = \mathbb{E}[\partial_x f(x, \xi)].$$

Stochastic optimization

Assumption 1: Possible to generate iid samples $\xi_1, \xi_2, \ldots$
Assumption 2: Oracle yields stochastic gradient $g(x, \xi)$, i.e.,

$$G(x) := \mathbb{E}[g(x, \xi)] \quad \text{s.t.} \quad G(x) \in \partial F(x).$$

Theorem Let $\xi \in \Omega$; If $f(\cdot, \xi)$ is convex, and $F(\cdot)$ is finite valued in a neighborhood of x, then

$$\partial F(x) = \mathbb{E}[\partial_x f(x, \xi)].$$

- So $g(x, \omega) \in \partial_x f(x, \omega)$ is a stochastic subgradient.

Stochastic optimization methods

♣ Stochastic Approximation (SA) / Stochastic gradient (SGD)

- Sample ξ iid
- Generate stochastic subgradient $g(x, \xi)$
- Use that in a subgradient method

Stochastic optimization methods

- ♣ Stochastic Approximation (SA) / Stochastic gradient (SGD)
  - Sample ξ iid
  - Generate stochastic subgradient $g(x, \xi)$
  - Use that in a subgradient method
- ♣ Sample average approximation (SAA)
  - Generate n iid samples, $\xi_1, \ldots, \xi_n$
  - Consider **empirical objective** $\hat{F}_n := n^{-1} \sum_i f(x, \xi_i)$
  - SAA refers to creation of this **sample average problem**
  - Minimizing $\hat{F}_n$ still needs to be done!

Stochastic gradient

SA or stochastic (sub)-gradient

- Let $x_0 \in \mathcal{X}$
- For $k \geq 0$
  - Sample ξ_k; compute $g(x_k, \xi_k)$ using oracle
  - Update $x_{k+1} = P_{\mathcal{X}}(x_k - \alpha_k g(x_k, \xi_k))$, where $\alpha_k > 0$

We'll simply write

$$x_{k+1} = P_{\mathcal{X}}\big(x_k - \alpha_k g_k\big)$$

Does this work?

SGD convergence

- x_k depends on rvs $\xi_1, \ldots, \xi_{k-1}$, so itself random
- Of course, x_k does not depend on ξ_k
- Subgradient method analysis hinges upon: $\|x_k - x^*\|^2$
- Stochastic subgradient hinges upon: $\mathbb{E}[\|x_k - x^*\|^2]$

Denote: $R_k := \|x_k - x^*\|^2$ and $r_k := \mathbb{E}[R_k] = \mathbb{E}[\|x_k - x^*\|^2]$

Bounding R_{k+1}

$$
\begin{aligned}
R_{k+1} &= \|x_{k+1} - x^*\|_2^2 = \|P_{\mathcal{X}}(x_k - \alpha_k g_k) - P_{\mathcal{X}}(x^*)\|_2^2 \\
&\leq \|x_k - x^* - \alpha_k g_k\|_2^2 \\
&= R_k + \alpha_k^2 \|g_k\|_2^2 - 2\alpha_k \langle g_k,\, x_k - x^* \rangle.
\end{aligned}
$$

SGD convergence

$$R_{k+1} \leq R_k + \alpha_k^2 \|g_k\|_2^2 - 2\alpha_k \langle g_k,\, x_k - x^* \rangle$$

- **Assume:** $\|g_k\|_2 \leq M$ on $\mathcal{X}$
- Taking expectation:

$$r_{k+1} \leq r_k + \alpha_k^2 M^2 - 2\alpha_k \mathbb{E}[\langle g_k,\, x_k - x^* \rangle].$$

- We need to now get a handle on the last term
- Since x_k is independent of ξ_k, we have

$$\begin{aligned}
\mathbb{E}[\langle x_k - x^*,\, g(x_k, \xi_k) \rangle] &= \mathbb{E}\left\{ \mathbb{E}[\langle x_k - x^*,\, g(x_k, \xi_k) \rangle \mid \xi_{[1..(k-1)]}] \right\} \\
&= \mathbb{E}\left\{ \langle x_k - x^*,\, \mathbb{E}[g(x_k, \xi_k) \mid \xi_{[1..(k-1)]}] \rangle \right\} \\
&= \mathbb{E}[\langle x_k - x^*,\, G_k \rangle], \quad G_k \in \partial F(x_k).
\end{aligned}$$

SGD convergence

It remains to bound: $\mathbb{E}[\langle x_k - x^*, G_k \rangle]$

- Since F is cvx, $F(x) \geq F(x_k) + \langle G_k, x - x_k \rangle$ for any $x \in \mathcal{X}$.
- Thus, in particular

$$2\alpha_k \mathbb{E}[F(x^*) - F(x_k)] \geq 2\alpha_k \mathbb{E}[\langle G_k, x^* - x_k \rangle]$$

Plug this bound back into the r_{k+1} inequality:

$$\begin{aligned} r_{k+1} &\leq r_k + \alpha_k^2 M^2 - 2\alpha_k \mathbb{E}[\langle G_k, x_k - x^* \rangle] \\ 2\alpha_k \mathbb{E}[\langle G_k, x_k - x^* \rangle] &\leq r_k - r_{k+1} + \alpha_k M^2 \\ 2\alpha_k \mathbb{E}[F(x_k) - F(x^*)] &\leq r_k - r_{k+1} + \alpha_k M^2. \end{aligned}$$

We've bounded the expected progress; What now?

SGD convergence

$$2\alpha_k \mathbb{E}[F(x_k) - F(x^*)] \leq r_k - r_{k+1} + \alpha_k M^2.$$

Sum up over $i = 1, \ldots, k$, to obtain

$$\begin{aligned} \sum_{i=1}^{k} (2\alpha_i \mathbb{E}[F(x_i) - f(x^*)]) &\leq r_1 - r_{k+1} + M^2 \sum_i \alpha_i^2 \\ &\leq r_1 + M^2 \sum_i \alpha_i^2. \end{aligned}$$

Divide both sides by $\sum_i \alpha_i$, so

- Set $\gamma_i = \frac{\alpha_i}{\sum_i^k \alpha_i}$.
- Thus, $\gamma_i \geq 0$ and $\sum_i \gamma_i = 1$

$$\mathbb{E}\left[\sum_i \gamma_i (F(x_i) - F(x^*))\right] \leq \frac{r_1 + M^2 \sum_i \alpha_i^2}{2 \sum_i \alpha_i}$$

SGD convergence

- But we wish to say something about x_k
- Since $\gamma_i \geq 0$ and $\sum_i^k \gamma_i = 1$, and we have $\gamma_i F(x_i)$
- Easier to talk about **averaged**

$$\bar{x}_k := \sum_i^k \gamma_i x_i.$$

- $f(\bar{x}_k) \leq \sum_i \gamma_i F(x_i)$ due to convexity
- So we finally obtain the inequality

$$\mathbb{E}\big[F(\bar{x}_k) - F(x^*)\big] \leq \frac{r_1 + M^2 \sum_i \alpha_i^2}{2 \sum_i \alpha_i}.$$

SGD convergence

- Let $D_{\mathcal{X}} := \max_{x \in \mathcal{X}} \|x - x^*\|_2$ (act. only need $\|x_1 - x^*\| \leq D_{\mathcal{X}}$)
- Assume $\alpha_i = \alpha$ is a constant. Observe that

$$\mathbb{E}[F(\bar{x}_k) - F(x^*)] \leq \frac{D_{\mathcal{X}}^2 + M^2 k \alpha^2}{2k\alpha}$$

- Minimize rhs over $\alpha > 0$; thus $\mathbb{E}[F(\bar{x}_k) - F(x^*)] \leq \frac{D_{\mathcal{X}} M}{\sqrt{k}}$
- If k is not fixed in advance, then choose

$$\alpha_i = \frac{\theta D_{\mathcal{X}}}{M\sqrt{i}}, \quad i = 1, 2, \ldots$$

We showed $O(1/\sqrt{k})$ rate

Smooth stochastic optimization

Theorem Let $f(x,\xi)$ be C_L^1 convex. Let $e_k := \nabla F(x_k) - g_k$ satisfy $\mathbb{E}[e_k] = 0$. Let $\|x_i - x^*\| \leq D$. Also, let $\alpha_i = 1/(L+\eta_i)$. Then,

$$\mathbb{E}\Big[\sum\nolimits_{i=1}^{k} F(x_{i+1}) - F(x^*)\Big] \leq \frac{D^2}{2\alpha_k} + \sum\nolimits_{i=1}^{k} \frac{\mathbb{E}[\|e_i\|^2]}{2\eta_i}.$$

Smooth stochastic optimization

Theorem Let $f(x,\xi)$ be C_L^1 convex. Let $e_k := \nabla F(x_k) - g_k$ satisfy $\mathbb{E}[e_k] = 0$. Let $\|x_i - x^*\| \le D$. Also, let $\alpha_i = 1/(L+\eta_i)$. Then,

$$\mathbb{E}\Big[\sum_{i=1}^{k} F(x_{i+1}) - F(x^*)\Big] \le \frac{D^2}{2\alpha_k} + \sum_{i=1}^{k} \frac{\mathbb{E}[\|e_i\|^2]}{2\eta_i}.$$

Plugging in average $\bar{x}_k = \frac{1}{k}\sum_{i=1}^{k} x_{i+1}$ we get

$$\mathbb{E}[F(\bar{x}_k) - F(x^*)] \le \frac{D^2}{2\alpha_k k} + \frac{1}{k}\sum_{i=1}^{k} \frac{\mathbb{E}[\|e_i\|^2]}{2\eta_i}.$$

- Using $\alpha_i = L + \eta_i$ where $\eta_i \propto 1/\sqrt{k}$ we obtain

$$\mathbb{E}[F(\bar{x}_k) - F(x^*)] = O\Big(\frac{LD^2}{k}\Big) + O\Big(\frac{\sigma D}{\sqrt{k}}\Big)$$

where σ bounds the variance $\mathbb{E}[\|e_i\|^2]$

Minimax optimal

Stochastic optimization – strongly convex

Theorem Suppose $f(x,\xi)$ are convex and $F(x)$ is μ-strongly convex. Let $\bar{x}_k := \sum_{i=0}^{k-1} \theta_i x_i$, where $\theta_i = \frac{2(i+1)}{(k+1)(k+2)}$, we obtain

$$\mathbb{E}[F(\bar{x}_k) - F(x^*)] \leq \frac{2M^2}{\mu(k+1)}.$$

(*Lacoste-Julien, Schmidt, Bach (2012)*)

With uniform averaging $\bar{x}_k = \frac{1}{k}\sum_i x_i$, we get $O(\log k/k)$.

SGD convergence summary

Cvx Class	Rate	Iterate	Minimax
C_L^0	$1/\sqrt{k}$	$\bar{x}_k$	Yes
C_L^0	$\log k/\sqrt{k}$	x_k	No
C_L^1	$1/\sqrt{k}$	$\bar{x}_k$	Yes
S_L^0	$(\log k)/k$	$\bar{x}_k$, x_k	No
S_L^1	$1/k$	$\bar{x}_k$, x_k	Yes

Extensions

- Proximal stochastic gradient

$$x_{k+1} = \text{prox}_{\alpha_k h}[x_k - \alpha_k g(x_k, \xi_k)]$$

  (*Xiao 2010; Hu et al. 2009*)

  Accelerated versions also possible
  (*Ghadimi, Lan (2013)*)

- Related methods:
  - Regularized dual averaging (Nesterov, 2009; Xiao 2010)
  - Stochastic mirror-prox (Nemirovski et al. 2009)
- ...

Finite-sum problems

$$\min_{x \in \mathbb{R}^d} \quad f(x) = \frac{1}{n} \sum_{i=1}^{n} f_i(x).$$

Finite-sum problems

$$\min_{x \in \mathbb{R}^d} \quad f(x) = \frac{1}{n} \sum_{i=1}^{n} f_i(x).$$

Gradient / subgradient methods

$$\begin{aligned}
x_{k+1} &= x_k - \alpha_k \nabla f(x_k) \\
x_{k+1} &= x_k - \alpha_k g(x_k), \quad g \in \partial f(x_k) \\
x_{k+1} &= \mathrm{prox}_{\alpha_k r}(x_k - \alpha_k \nabla f(x_k))
\end{aligned}$$

Stochastic gradient

At iteration k, we randomly pick an integer
$i(k) \in \{1, 2, \ldots, m\}$

$$x_{k+1} = x_k - \alpha_k \nabla f_{i(k)}(x_k)$$

- The update requires only gradient for $f_{i(k)}$
- Uses unbiased estimate $\mathbb{E}[\nabla f_{i(k)}] = \nabla f$
- One iteration now n times faster using $\nabla f(x)$
- But how many iterations do we need?

Stochastic gradient

Method	Assumptions	Full	Stochastic
Subgradient	convex	$O(1/\sqrt{k})$	$O(1/\sqrt{k})$
Subgradient	strongly cvx	$O(1/k)$	$O(1/k)$

So using stochastic subgradient, solve n times faster.

Stochastic gradient

Method	Assumptions	Full	Stochastic
Subgradient	convex	$O(1/\sqrt{k})$	$O(1/\sqrt{k})$
Subgradient	strongly cvx	$O(1/k)$	$O(1/k)$

So using stochastic subgradient, solve n times faster.

Method	Assumptions	Full	Stochastic
Gradient	convex	$O(1/k)$	$O(1/\sqrt{k})$
Gradient	strongly cvx	$O((1-\mu/L)^k)$	$O(1/k)$

– For smooth problems, stochastic gradient needs more iterations
– Widely used in ML, rapid initial convergence
– Several speedup techniques studied, but worst case remains same

Hybrid methods

► Hybrid of stochastic gradient with full gradient.

Stochastic Average Gradient (SAG) (Le Roux, Schmidt, Bach 2012)

- store the gradients of ∇f_i for $i = 1, .., n$
- Select uniformly at random $i(k) \in \{1, \ldots, n\}$
- Perform the update

$$x_{k+1} = x_k - \frac{\alpha_k}{n} \sum_{i=1}^{n} y_i^k \quad y_i^k = \begin{cases} \nabla f_i(x_k) & \text{if } i = i(k) \\ y_i^{k-1} & \text{otherwise.} \end{cases}$$

Hybrid methods

- Hybrid of stochastic gradient with full gradient.

Stochastic Average Gradient (SAG) (Le Roux, Schmidt, Bach 2012)

- store the gradients of ∇f_i for $i = 1, .., n$
- Select uniformly at random $i(k) \in \{1, \ldots, n\}$
- Perform the update

$$x_{k+1} = x_k - \frac{\alpha_k}{n} \sum_{i=1}^{n} y_i^k \quad y_i^k = \begin{cases} \nabla f_i(x_k) & \text{if } i = i(k) \\ y_i^{k-1} & \text{otherwise.} \end{cases}$$

- Randomized / stochastic version of incremental gradient method of Blatt et al (2008)
- Storage overhead; acceptable in some ML settings:
  - $f_i(x) = \ell(l_i, x^T\Phi(a_i))$, $\nabla f_i(x) = \nabla\ell(l_i, x^T\Phi(a_i))\Phi(a_i)$
  - Store only n scalars (since depends only on $x^T a_i$)

Convergence rates

Method	Assumptions	Rate
Gradient	convex	$O(1/k)$
Gradient	strongly cvx	$O((1-\mu/L)^k)$
Stochastic	strongly cvx	$O(1/k)$
SAG	strongly convex	$O((1-\min\{\frac{\mu}{n}, \frac{1}{8n}\})^k)$

This speedup also observed in practice

Complicated convergence analysis

Similar rates for many other methods

- stochastic dual coordinate (SDCA); [Shalev-Shwartz, Zhang, 2013]
- stochastic variance reduced gradient (SVRG); [Johnson, Zhang, 2013]
- proximal SVRG [Xiao, Zhang, 2014]
- hybrid of SAG and SVRG, SAGA (also proximal); [Defazio et al, 2014]
- accelerated versions [Lin, Mairal, Harchoui; 2015]
- asynchronous hybrid SVRG [Reddi et al. 2015]
- incremental Newton method, S2SGD and MS2GD, ...

Incremental Gradient Methods

$$\min F(x) = \frac{1}{n}\sum_{i=1}^{n} f_i(x)$$

The incremental gradient method (IGM)

- Let $x_0 \in \mathbb{R}^n$
- For $k \geq 0$

Incremental Gradient Methods

$$\min F(x) = \frac{1}{n} \sum_{i=1}^{n} f_i(x)$$

The incremental gradient method (IGM)

- Let $x_0 \in \mathbb{R}^n$
- For $k \geq 0$
  1. Pick $i(k) \in \{1, 2, \ldots, n\}$ uniformly at random
  2. $x_{k+1} = x_k - \eta_k \nabla f_{i(k)}(x_k)$

Incremental Gradient Methods

$$\min F(x) = \frac{1}{n}\sum_{i=1}^{n} f_i(x)$$

The incremental gradient method (IGM)

- Let $x_0 \in \mathbb{R}^n$
- For $k \geq 0$
  1. Pick $i(k) \in \{1, 2, \ldots, n\}$ uniformly at random
  2. $x_{k+1} = x_k - \eta_k \nabla f_{i(k)}(x_k)$

$g \equiv \nabla f_{i(k)}$ may be viewed as a **stochastic gradient**

Incremental Gradient Methods

$$\min F(x) = \frac{1}{n}\sum_{i=1}^{n} f_i(x)$$

The incremental gradient method (IGM)

- Let $x_0 \in \mathbb{R}^n$
- For $k \geq 0$
  1. Pick $i(k) \in \{1, 2, \ldots, n\}$ uniformly at random
  2. $x_{k+1} = x_k - \eta_k \nabla f_{i(k)}(x_k)$

$g \equiv \nabla f_{i(k)}$ may be viewed as a **stochastic gradient**

$g := g^{\text{true}} + e$, where e is mean-zero noise: $\mathbb{E}[e] = 0$

Incremental Gradient Methods

- Index $i(k)$ chosen uniformly from $\{1, \ldots, n\}$
- Thus, **in expectation:**

$$\mathbb{E}[g] =$$

Incremental Gradient Methods

- Index $i(k)$ chosen uniformly from $\{1, \ldots, n\}$
- Thus, **in expectation:**

$$\mathbb{E}[g] = \mathbb{E}_i[\nabla f_i(x)]$$

Incremental Gradient Methods

- Index $i(k)$ chosen uniformly from $\{1, \ldots, n\}$
- Thus, **in expectation:**

$$\mathbb{E}[g] = \mathbb{E}_i[\nabla f_i(x)] = \sum_i \frac{1}{n} \nabla f_i(x) =$$

Incremental Gradient Methods

- Index $i(k)$ chosen uniformly from $\{1, \ldots, n\}$
- Thus, **in expectation:**

$$\mathbb{E}[g] = \mathbb{E}_i[\nabla f_i(x)] = \sum_i \tfrac{1}{n} \nabla f_i(x) = \nabla F(x)$$

- Alternatively, $\mathbb{E}[g - g^{\text{true}}] = \mathbb{E}[e] = 0$.
- We call g an **unbiased estimate** of the gradient
- Here, we **obtained** g in a two step process:
  - **Sample:** pick an index $i(k)$ unif. at random
  - **Oracle:** Compute a random gradient based on $i(k)$

Incremental Gradient Methods

- Index $i(k)$ chosen uniformly from $\{1, \ldots, n\}$
- Thus, **in expectation:**

$$\mathbb{E}[g] = \mathbb{E}_i[\nabla f_i(x)] = \sum_i \tfrac{1}{n} \nabla f_i(x) = \nabla F(x)$$

- Alternatively, $\mathbb{E}[g - g^{\text{true}}] = \mathbb{E}[e] = 0.$
- We call g an **unbiased estimate** of the gradient
- Here, we **obtained** g in a two step process:
  - **Sample:** pick an index $i(k)$ unif. at random
  - **Oracle:** Compute a random gradient based on $i(k)$
- Individual g_k values can vary a lot
- Variance ($\mathbb{E}[\|g - g^{\text{true}}\|^2]$) influences convergence rate

Controlling variance

- Instead of using $g_k = \nabla f_{i(k)}(x_k)$, **correct** it by using **true gradient** every $m \geq n$ steps (recall: $F = \frac{1}{n}\sum_{i=1}^{n} f_i(x)$)

Controlling variance

- Instead of using $g_k = \nabla f_{i(k)}(x_k)$, **correct** it by using **true gradient** every $m \geq n$ steps (recall: $F = \frac{1}{n}\sum_{i=1}^{n} f_i(x)$)
- Reduces variance of $g_k(x_k, \xi_k)$; speeds up convergence

Controlling variance

- Instead of using $g_k = \nabla f_{i(k)}(x_k)$, **correct** it by using **true gradient** every $m \geq n$ steps (recall: $F = \frac{1}{n}\sum_{i=1}^{n} f_i(x)$)
- Reduces variance of $g_k(x_k, \xi_k)$; speeds up convergence

$$\begin{aligned} \nabla F(\bar{x}) &= \frac{1}{m}\sum_i f_i(\bar{x}) \\ x_{k+1} &= x_k - \eta_k \underbrace{[\nabla f_{i(k)}(x_k) - \nabla f_{i(k)}(\bar{x}) + \nabla F(\bar{x})]}_{g_k(x_k,\xi_k)} \end{aligned}$$

Controlling variance

- Instead of using $g_k = \nabla f_{i(k)}(x_k)$, **correct** it by using **true gradient** every $m \geq n$ steps (recall: $F = \frac{1}{n}\sum_{i=1}^{n} f_i(x)$)
- Reduces variance of $g_k(x_k, \xi_k)$; speeds up convergence

$$\begin{aligned} \nabla F(\bar{x}) &= \frac{1}{m}\sum_i f_i(\bar{x}) \\ x_{k+1} &= x_k - \eta_k \underbrace{[\nabla f_{i(k)}(x_k) - \nabla f_{i(k)}(\bar{x}) + \nabla F(\bar{x})]}_{g_k(x_k, \xi_k)} \end{aligned}$$

- Thus, with $\xi_k = i(k)$, $\mathbb{E}_\xi[g_k | x_k] = \nabla F(x_k)$
  But with lower variance!

Controlling variance

- Instead of using $g_k = \nabla f_{i(k)}(x_k)$, **correct** it by using **true gradient** every $m \geq n$ steps (recall: $F = \frac{1}{n}\sum_{i=1}^{n} f_i(x)$)
- Reduces variance of $g_k(x_k, \xi_k)$; speeds up convergence

$$\begin{aligned}\nabla F(\bar{x}) &= \frac{1}{m}\sum_i f_i(\bar{x}) \\ x_{k+1} &= x_k - \eta_k \underbrace{[\nabla f_{i(k)}(x_k) - \nabla f_{i(k)}(\bar{x}) + \nabla F(\bar{x})]}_{g_k(x_k,\xi_k)}\end{aligned}$$

- Thus, with $\xi_k = i(k)$, $\mathbb{E}_\xi[g_k|x_k] = \nabla F(x_k)$
  But with lower variance!

SVRG

- For $s \geq 1$:
  1. $\bar{x} \leftarrow \bar{x}_{s-1}$
  2. $\bar{g} \leftarrow \nabla F(\bar{x})$ (full gradient computation)
  3. $x_0 = \bar{x}; \quad t \leftarrow \text{RAND}(1, m)$ (randomized stopping)
  4. For $k = 0, 1, \ldots, t-1$
     - Randomly pick $i(k) \in [1..m]$
     - $x_{k+1} = x_k - \eta_k(\nabla f_{i(k)}(x_k) - \nabla f_{i(k)}(\bar{x}) + \bar{g})$
  5. $\bar{x}_s \leftarrow x_t$

SVGR

- For $s \geq 1$:
  1. $\bar{x} \leftarrow \bar{x}_{s-1}$
  2. $\bar{g} \leftarrow \nabla F(\bar{x})$ (full gradient computation)
  3. $x_0 = \bar{x}; \quad t \leftarrow \text{RAND}(1, m)$ (randomized stopping)
  4. For $k = 0, 1, \ldots, t-1$
     - Randomly pick $i(k) \in [1..m]$
     - $x_{k+1} = x_k - \eta_k(\nabla f_{i(k)}(x_k) - \nabla f_{i(k)}(\bar{x}) + \bar{g})$
  5. $\bar{x}_s \leftarrow x_t$

Theorem Assume each $f_i(x)$ is smooth, and $F(x)$ strongly-convex. Then, for sufficiently large n, there is $\alpha < 1$ s.t.

$$\mathbb{E}[F(\bar{x}_s) - F(x^*)] \leq \alpha^s [F(\bar{x}_0) - F(x^*)]$$