4. Proximal gradient method

- motivation
- proximal mapping
- proximal gradient method with fixed step size
- proximal gradient method with line search

Proximal mapping

the **proximal mapping** (or **prox-operator**) of a convex function h is defined as

$$\operatorname{prox}_h(x) = \operatorname*{argmin}_u \left(h(u) + \frac{1}{2}\|u - x\|_2^2 \right)$$

$$prox_{th}(x) = \arg\min_u \{t \cdot h(u) + \frac{1}{2} \| u - x \|_2^2\}$$

$$\boxed{prox_{th}(x) = \arg\min_u \{h(u) + \frac{1}{2t} \| u - x \|_2^2\}}$$

Examples

- $h(x) = 0$: $\operatorname{prox}_h(x) = x$
- $h(x)$ is indicator function of closed convex set C: prox_h is projection on C

$$\operatorname{prox}_h(x) = \operatorname*{argmin}_{u \in C} \|u - x\|_2^2 = P_C(x)$$

- $h(x) = \|x\|_1$: prox_h is the "soft-threshold" (shrinkage) operation

$$\operatorname{prox}_h(x)_i = \begin{cases} x_i - 1 & x_i \geq 1 \\ 0 & |x_i| \leq 1 \\ x_i + 1 & x_i \leq -1 \end{cases}$$

Proximal gradient method

unconstrained optimization with objective split in two components

$$\text{minimize} \quad f(x) = g(x) + h(x)$$

- g convex, differentiable, $\mathbf{dom}\, g = \mathbf{R}^n$
- h convex with inexpensive prox-operator

Proximal gradient algorithm

$$x_{k+1} = \mathrm{prox}_{t_k h}(x_k - t_k \nabla g(x_k))$$

- $t_k > 0$ is step size, constant or determined by line search
- can start at infeasible x_0 (however $x_k \in \mathbf{dom}\, f = \mathbf{dom}\, h$ for $k \geq 1$)

Interpretation

$$x^+ = \text{prox}_{th}(x - t\nabla g(x))$$

from definition of proximal mapping:

$$\begin{aligned} x^+ &= \underset{u}{\text{argmin}} \left(h(u) + \frac{1}{2t}\|u - x + t\nabla g(x)\|_2^2 \right) \\ &= \underset{u}{\text{argmin}} \left(h(u) + g(x) + \nabla g(x)^T(u-x) + \frac{1}{2t}\|u-x\|_2^2 \right) \end{aligned}$$

x^+ minimizes $h(u)$ plus a simple quadratic local model of $g(u)$ around x

$$\begin{aligned} g(u) &= g(x) + \nabla g(x)^T(u-x) + \frac{1}{2}(u-x)^T\nabla^2 g(x)(u-x) + O(\|u-x\|^2) \\ &= g(x) + \nabla g(x)^T(u-x) + \frac{1}{2}(u-x)^T\nabla^2 g(\xi)(u-x) \\ &\le g(x) + \nabla g(x)^T(u-x) + \frac{1}{2}\lambda_m\|u-x\|^2 \end{aligned}$$

Examples

$$\text{minimize} \quad g(x) + h(x)$$

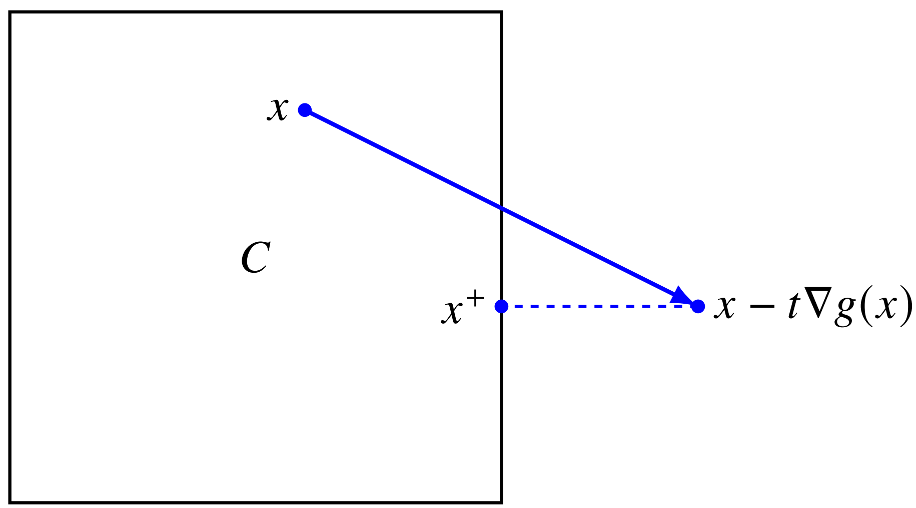
Gradient method: special case with $h(x) = 0$

$$\boxed{\min g(x) \;\; s.t.\; h(x) = \delta_C(x).}$$

$$x^+ = x - t\nabla g(x)$$

Gradient projection method: special case with $h(x) = \delta_C(x)$ (indicator of C)

$$x^+ = P_C\left(x - t\nabla g(x)\right)$$



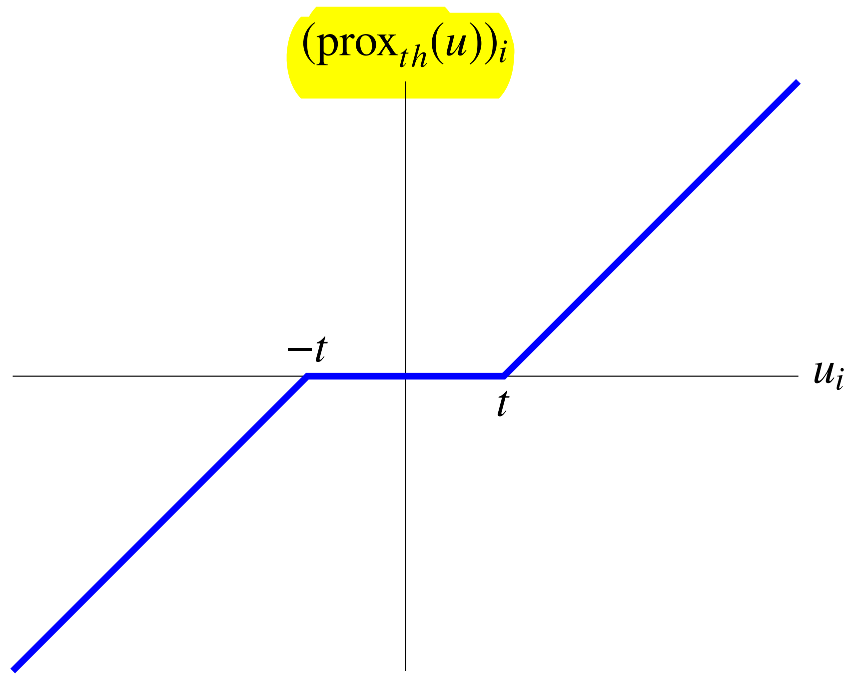
Examples

Soft-thresholding: special case with $h(x) = \|x\|_1$ $\longrightarrow$ $\boxed{\min\ g(x) + \|x\|_1^1}$

$$x^+ = \mathrm{prox}_{th}\left(x - t\nabla g(x)\right)$$

where

$$(\mathrm{prox}_{th}(u))_i = \begin{cases} u_i - t & u_i \geq t \\ 0 & -t \leq u_i \leq t \\ u_i + t & u_i \leq -t \end{cases}$$



Outline

- motivation
- **proximal mapping**
- proximal gradient method with fixed step size
- proximal gradient method with line search

Proximal mapping

if h is convex and closed (has a closed epigraph), then

$$\text{prox}_h(x) = \underset{u}{\text{argmin}} \left(h(u) + \frac{1}{2}\|u - x\|_2^2 \right)$$

exists and is unique for all x

- will be studied in more detail in one of the next lectures
- from optimality conditions of minimization in the definition:

凸优化的极值点的一阶充要条件是导数为零

$$\begin{aligned} u = \text{prox}_h(x) \quad &\Longleftrightarrow \quad x - u \in \partial h(u) \\ &\Longleftrightarrow \quad h(z) \geq h(u) + (x-u)^T(z-u) \quad \text{for all } z \end{aligned}$$

凸函数一阶等价条件定义：函数曲线在直线上方

Projection on closed convex set

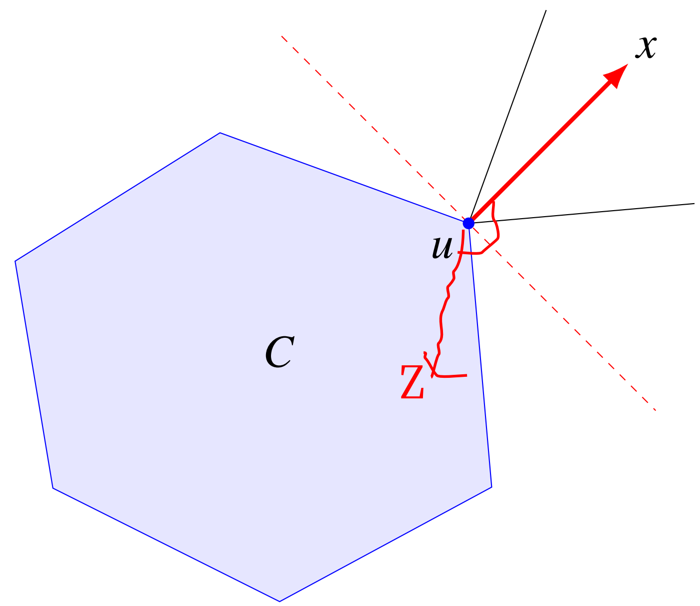
proximal mapping of indicator function δ_C is Euclidean projection on C

$$\mathrm{prox}_{\delta_C}(x) = \underset{u \in C}{\mathrm{argmin}}\, \|u - x\|_2^2 = P_C(x)$$

$$u = P_C(x)$$

$$\Updownarrow$$

$$(x - u)^T(z - u) \le 0 \quad \forall z \in C$$



we will see that proximal mappings have many properties of projections

Firm nonexpansiveness

近端映射是牢固非膨胀的

proximal mappings are **firmly nonexpansive** (co-coercive with constant 1):

$$\boxed{\|prox_h(x) - prox_h(y)\| \cdot \| x - y \|}$$

$$\geqslant (\mathrm{prox}_h(x) - \mathrm{prox}_h(y))^T (x - y) \geq \|\mathrm{prox}_h(x) - \mathrm{prox}_h(y)\|_2^2$$

- follows from page 4.7: if $u = \mathrm{prox}_h(x)$, $v = \mathrm{prox}_h(y)$, then

$$x - u \in \partial h(u), \qquad y - v \in \partial h(v)$$

combining this with monotonicity of subdifferential (page 2.9) gives

次微分单调增的

$$(x - u - y + v)^T (u - v) \geq 0$$

- a weaker property is **nonexpansiveness** (Lipschitz continuity with constant 1):

$$\|\mathrm{prox}_h(x) - \mathrm{prox}_h(y)\|_2 \leq \|x - y\|_2$$

follows from firm nonexpansiveness and Cauchy–Schwarz inequality

Outline

- motivation
- proximal mapping
- **proximal gradient method with fixed step size**
- proximal gradient method with line search

Assumptions

$$\text{minimize} \quad f(x) = g(x) + h(x)$$

- h is closed and convex (so that prox_{th} is well defined)
- g is differentiable with $\mathrm{dom}\, g = \mathbf{R}^n$, and L-smooth for the Euclidean norm, *i.e.*,

$$\boxed{g(x) \le g(x_0) + \nabla g(x_0)^T (x - x_0) + \frac{L}{2} \| x - x_0 \|_2^2}$$

$$\frac{L}{2} x^T x - g(x) \quad \text{is convex}$$

- there exists a constant $m \ge 0$ such that

$$g(x) - \frac{m}{2} x^T x \quad \text{is convex}$$

$$\boxed{g(x) \ge g(x_0) + \nabla g(x_0)^T (x - x_0) + \frac{m}{2} \| x - x_0 \|_2^2}$$

when $m > 0$ this is m-strong convexity for the Euclidean norm

- the optimal value $f^\star$ is finite and attained at $x^\star$ (not necessarily unique)

Implications of assumptions on g

Lower bound

- convexity of the the function $g(x) - (m/2)x^T x$ implies (page 1.19):

$$g(y) \geq g(x) + \nabla g(x)^T (y - x) + \frac{m}{2}\|y - x\|_2^2 \qquad \text{for all } x, y \tag{1}$$

- if $m = 0$, this means g is convex; if $m > 0$, strongly convex (lecture 1)

Upper bound

- convexity of the function $(L/2)x^T x - g(x)$ implies (page 1.12):

$$g(y) \leq g(x) + \nabla g(x)^T (y - x) + \frac{L}{2}\|y - x\|_2^2 \qquad \text{for all } x, y \tag{2}$$

- this is equivalent to Lipschitz continuity and co-coercivity of gradient (lecture 1)

Gradient map

$$G_t(x) = \frac{1}{t}\left(x - \mathrm{prox}_{th}(x - t\nabla g(x))\right)$$

$G_t(x)$ is the negative "step" in the proximal gradient update

迭代公式：
$$\begin{aligned} \boxed{x^+ \quad = \quad \mathrm{prox}_{th}\left(x - t\nabla g(x)\right)} \\ = \quad x - tG_t(x) \end{aligned}$$

- $G_t(x)$ is not a gradient or subgradient of $f = g + h$
- from subgradient definition of prox-operator (page 4.7),

$$G_t(x) \in \nabla g(x) + \partial h\left(x - tG_t(x)\right)$$

- $G_t(x) = 0$ if and only if x minimizes $f(x) = g(x) + h(x)$

Consequences of quadratic bounds on g

substitute $y = x - tG_t(x)$ in the bounds (1) and (2): for all t,

$$\frac{mt^2}{2}\|G_t(x)\|_2^2 \leq g\left(x - tG_t(x)\right) - g(x) + t\nabla g(x)^T G_t(x) \leq \frac{Lt^2}{2}\|G_t(x)\|_2^2$$

- if $0 < t \leq 1/L$, then the upper bound implies

$$g(x - tG_t(x)) \leq g(x) - t\nabla g(x)^T G_t(x) + \frac{t}{2}\|G_t(x)\|_2^2 \qquad (3)$$

- if the inequality (3) is satisfied and $tG_t(x) \neq 0$, then $mt \leq 1$
- if the inequality (3) is satisfied, then for all z,

$$f(x - tG_t(x)) \leq f(z) + G_t(x)^T(x - z) - \frac{t}{2}\|G_t(x)\|_2^2 - \frac{m}{2}\|x - z\|_2^2 \qquad (4)$$

(proof on next page)

Proof of (4):

$$
\begin{aligned}
& f(x - tG_t(x)) \\
\leq \quad & g(x) - t\nabla g(x)^T G_t(x) + \frac{t}{2}\|G_t(x)\|_2^2 + h(x - tG_t(x)) \\
\leq \quad & g(z) - \nabla g(x)^T (z - x) - \frac{m}{2}\|z - x\|_2^2 - t\nabla g(x)^T G_t(x) + \frac{t}{2}\|G_t(x)\|_2^2 \\
& + h(x - tG_t(x)) \\
\leq \quad & g(z) - \nabla g(x)^T (z - x) - \frac{m}{2}\|z - x\|_2^2 - t\nabla g(x)^T G_t(x) + \frac{t}{2}\|G_t(x)\|_2^2 \\
& + h(z) - (G_t(x) - \nabla g(x))^T (z - x + tG_t(x)) \\
= \quad & g(z) + h(z) + G_t(x)^T (x - z) - \frac{t}{2}\|G_t(x)\|_2^2 - \frac{m}{2}\|x - z\|_2^2
\end{aligned}
$$

- in the first step we add $h(x - tG_t(x))$ to both sides of the inequality (3)
- in the next step we use the lower bound on $g(z)$ from (1)
- in step 3, we use $G_t(x) - \nabla g(x) \in \partial h(x - tG_t(x))$ (see page 4.12)

Progress in one iteration

for a step size t that satisfies the inequality (3), define

$$x^+ = x - tG_t(x)$$

- inequality (4) with $z = x$ shows that the algorithm is a descent method:

$$f(x^+) \le f(x) - \frac{t}{2}\|G_t(x)\|_2^2$$

- inequality (4) with $z = x^\star$ shows that

$$\begin{aligned}
f(x^+) - f^\star &\le G_t(x)^T(x - x^\star) - \frac{t}{2}\|G_t(x)\|_2^2 - \frac{m}{2}\|x - x^\star\|_2^2 \\
&= \frac{1}{2t}\left(\left\|x - x^\star\right\|_2^2 - \left\|x - x^\star - tG_t(x)\right\|_2^2\right) - \frac{m}{2}\|x - x^\star\|_2^2 \\
&= \frac{1}{2t}\left((1 - mt)\|x - x^\star\|_2^2 - \|x^+ - x^\star\|_2^2\right) \qquad (5) \\
&\le \frac{1}{2t}\left(\|x - x^\star\|_2^2 - \|x^+ - x^\star\|_2^2\right) \qquad (6)
\end{aligned}$$

Analysis for fixed step size

add inequalities (6) with $x = x_i$, $x^+ = x_{i+1}$, $t = t_i = 1/L$ from $i = 0$ to $i = k - 1$

$$
\begin{aligned}
\sum_{i=1}^{k} (f(x_i) - f^\star) &\leq \frac{1}{2t} \sum_{i=0}^{k-1} \left(\|x_i - x^\star\|_2^2 - \|x_{i+1} - x^\star\|_2^2 \right) \\
&= \frac{1}{2t} \left(\|x_0 - x^\star\|_2^2 - \|x_k - x^\star\|_2^2 \right) \\
&\leq \frac{1}{2t} \|x_0 - x^\star\|_2^2
\end{aligned}
$$

since $f(x_i)$ is nonincreasing,

$$
f(x_k) - f^\star \leq \frac{1}{k} \sum_{i=1}^{k} (f(x_i) - f^\star) \leq \frac{1}{2kt} \|x_0 - x^\star\|_2^2
$$

Distance to optimal set

- from (5) and $f(x^+) \geq f^\star$, the distance to the optimal set does not increase:

$$
\begin{aligned}
\|x^+ - x^\star\|_2^2 &\leq (1 - mt)\|x - x^\star\|_2^2 \\
&\leq \|x - x^\star\|_2^2
\end{aligned}
$$

- for fixed step size $t_k = 1/L$

$$
\|x_k - x^\star\|_2^2 \leq c^k \|x_0 - x^\star\|_2^2, \qquad c = 1 - \frac{m}{L}
$$

i.e., linear convergence if g is strongly convex $(m > 0)$

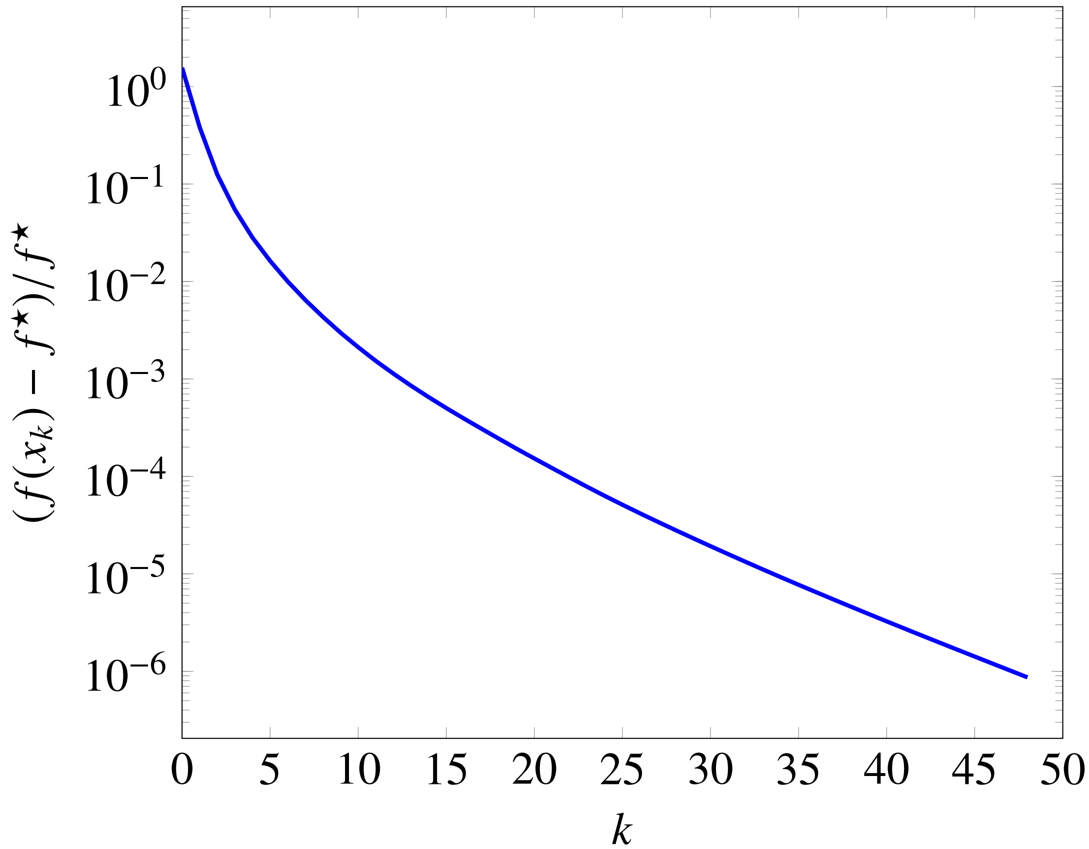
思考：有哪些评判收敛速度的方式？ https://zhuanlan.zhihu.com/p/27644403

	Gradient Descent	Convergence	Insure
Lipschitz continuous gradient (non-convex)	$O(\frac{1}{\epsilon^2})$	Sublinear	$\min_t \|f'(x_t)\| \leqslant \epsilon$
Lipschitz continuous gradient + Convex	$O(\frac{1}{\epsilon})$	Sublinear	$\lvert f(x_T) - f^\star \rvert \leqslant \epsilon$
Lipschitz continuous gradient + Strongly Convex	$O(\log\frac{1}{\epsilon})$	Linear	$\lvert f(x_T) - f^\star \rvert \leqslant \epsilon$ $\|x_T - x^\star\| \leqslant \epsilon$

	Sublinear	Linear	Quadratic
Order	$O(1/\epsilon^k)$	$O(\log(1/\epsilon))$	$O(\log\log(1/\epsilon))$
Iteration Function	$\|x_T - x^\star\| \leqslant \frac{1}{T^{\frac{1}{k}}}\|x_0 - x^\star\|$	$\|x_{t+1} - x^\star\| \leqslant q\|x_t - x^\star\|$	$\|x_{t+1} - x^\star\| \leqslant q\|x_t - x^\star\|^2$

Example: quadratic program with box constraints

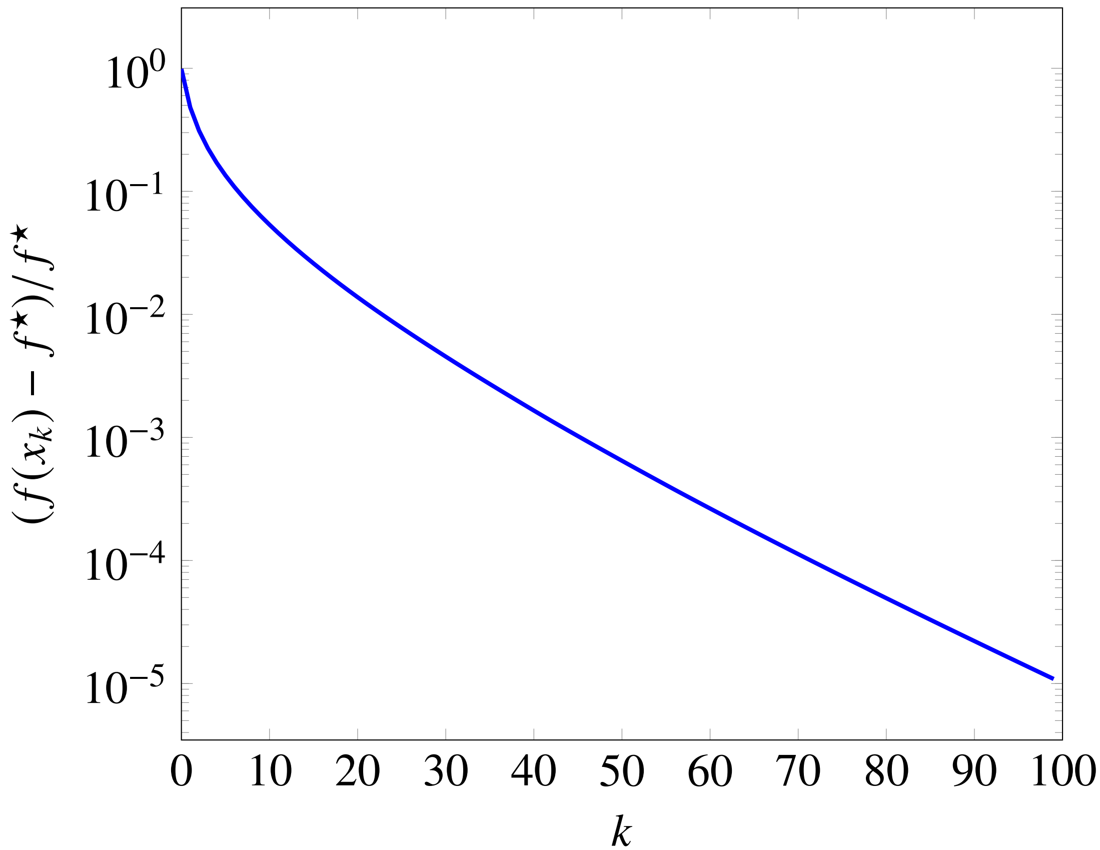
$$\begin{array}{ll} \text{minimize} & (1/2)x^T Ax + b^T x \\ \text{subject to} & \mathbf{0} \preceq x \preceq \mathbf{1} \end{array}$$



$n = 3000$; fixed step size $t = 1/\lambda_{\max}(A)$

Example: 1-norm regularized least-squares

$$\text{minimize} \quad \frac{1}{2}\|Ax - b\|_2^2 + \|x\|_1$$



randomly generated $A \in \mathbf{R}^{2000 \times 1000}$; step $t_k = 1/L$ with $L = \lambda_{\max}(A^T A)$

Outline

- introduction
- proximal mapping
- proximal gradient method with fixed step size
- **proximal gradient method with line search**

Line search

- the analysis for fixed step size (page 4.13) starts with the inequality

$$g(x - tG_t(x)) \leq g(x) - t\nabla g(x)^T G_t(x) + \frac{t}{2}\|G_t(x)\|_2^2 \tag{3}$$

  this inequality is known to hold for $0 < t \leq 1/L$

- if L is not known, we can satisfy (3) by a backtracking line search:

  start at some $t := \hat{t} > 0$ and backtrack ($t := \beta t$) until (3) holds

- step size t selected by the line search satisfies $t \geq t_{\min} = \min\{\hat{t}, \beta/L\}$

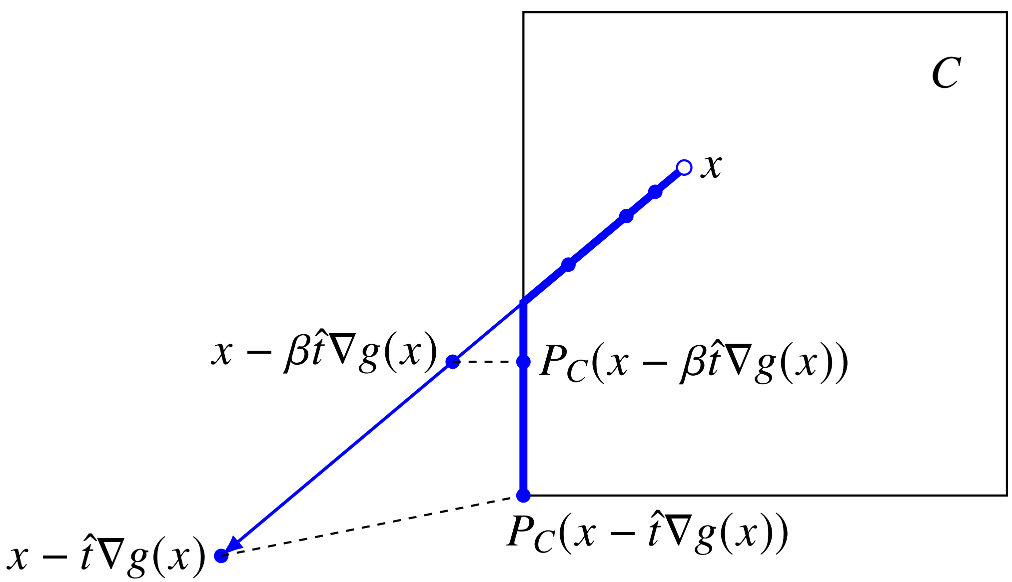
- requires one evaluation of g and prox_{th} per line search iteration

several other types of line search work

Example

line search for gradient projection method

$$x^+ = P_C\left(x - t\nabla g(x)\right) = x - tG_t(x)$$



backtrack until $P_C(x - t\nabla g(x))$ satisfies the "sufficient decrease" inequality (3)

Analysis with line search

from page 4.15, if (3) holds in iteration i, then $f(x_{i+1}) < f(x_i)$ and

$$t_i(f(x_{i+1}) - f^\star) \leq \frac{1}{2}\left(\|x_i - x^\star\|_2^2 - \|x_{i+1} - x^\star\|_2^2\right)$$

- adding inequalities for $i = 0$ to $i = k - 1$ gives

$$(\sum_{i=0}^{k-1} t_i)(f(x_k) - f^\star) \leq \sum_{i=0}^{k-1} t_i(f(x_{i+1}) - f^\star) \leq \frac{1}{2}\|x_0 - x^\star\|_2^2$$

  first inequality holds because $f(x_i)$ is nonincreasing

- since $t_i \geq t_{\min}$, we obtain a similar $1/k$ bound as for fixed step size

$$f(x_k) - f^\star \leq \frac{1}{2\sum_{i=0}^{k-1} t_i}\|x_0 - x^\star\|_2^2 \leq \frac{1}{2kt_{\min}}\|x_0 - x^\star\|_2^2$$

Distance to optimal set

from page 4.15, if (3) holds in iteration i, then

$$\begin{aligned}
\|x_{i+1} - x^\star\|_2^2 &\leq (1 - mt_i)\|x_i - x^\star\|_2^2 \\
&\leq (1 - mt_{\min})\,\|x_i - x^\star\|_2^2 \\
&= c\,\|x_i - x^\star\|_2^2 \\
\|x_k - x^\star\|_2^2 &\leq c^k\,\|x_0 - x^\star\|_2^2
\end{aligned}$$

with

$$c = 1 - mt_{\min} = \max\{1 - \frac{\beta m}{L}, 1 - m\hat{t}\}$$

hence linear convergence if $m > 0$

Summary: proximal gradient method

- minimizes sums of differentiable and non-differentiable convex functions

$$f(x) = g(x) + h(x)$$

- useful when nondifferentiable term h is simple (has inexpensive prox-operator)
- convergence properties are similar to standard gradient method ($h(x) = 0$)

虽然不可微，但是收敛速度类似梯度下降法

- less general but faster than subgradient method

1. 没有次梯度方法普适，2. 但是近端梯度法比次梯度法快速收敛。

References

- A. Beck, *First-Order Methods in Optimization* (2017), §10.4 and §10.6.
- A. Beck and M. Teboulle, *A fast iterative shrinkage-thresholding algorithm for linear inverse problems*, SIAM Journal on Imaging Sciences (2009).
- A. Beck and M. Teboulle, *Gradient-based algorithms with applications to signal recovery*, in: Y. Eldar and D. Palomar (Eds.), *Convex Optimization in Signal Processing and Communications* (2009).
- Yu. Nesterov, *Lectures on Convex Optimization* (2018), §2.2.3–2.2.4.
- B. T. Polyak, *Introduction to Optimization* (1987), §7.2.1.

附 录：

$$\min_{\mathbf{x}\in\mathbb{R}^n} F(\mathbf{x}) = f(\mathbf{x}) + g(\mathbf{x}),$$

where f is differentiable (it can be nonconvex) and g can be both nonconvex and nonsmooth.

Table 1: Comparisons of GD (General Descent Method), iPiano, GIST, GDPA, IR, IFB, APG, UAG and our method for problem (1). The measurements include the assumption, whether the methods accelerate for convex programs (CP) and converge for nonconvex programs (NCP).

Method name	Assumption	Accelerate (CP)	converge (NCP)
GD [16, 17]	$f+g$: KL	No	Yes
iPiano [18]	nonconvex f, convex g	No	Yes
GIST [19]	nonconvex f, $g=g_1-g_2$, g_1, g_2 convex	No	Yes
GDPA [20]	nonconvex f, $g=g_1-g_2$, g_1, g_2 convex	No	Yes
IR [8, 21]	special f and g	No	Yes
IFB [22]	nonconvex f, nonconvex g	No	Yes
APG [12, 13]	convex f, convex g	Yes	Unclear
UAG [15]	nonconvex f, convex g	Yes	Yes
Ours	nonconvex f, nonconvex g	Yes	Yes

常见梯度算法总结：https://blog.csdn.net/heyongluoyao8/article/details/52478715

1. Huan Li, Zhouchen Lin: Accelerated Proximal Gradient Methods for Nonconvex Programming. NIPS 2015: 379-387.
2. Sebastian Ruder: An overview of gradient descent optimization algorithms. CoRR abs/1609.04747 (2016).