Convex Optimization

Prof. Nati Srebro

Lecture 5:

Conjugate Gradient Descent

Quasi-Newton (BFGS)

Reading: Nocedal and Wright 5.1-5.2,6.1,7.2

Accelerated Gradient Descent

Reading: Bubeck 3.7

Newton vs Gradient Descent

Gradient Descent

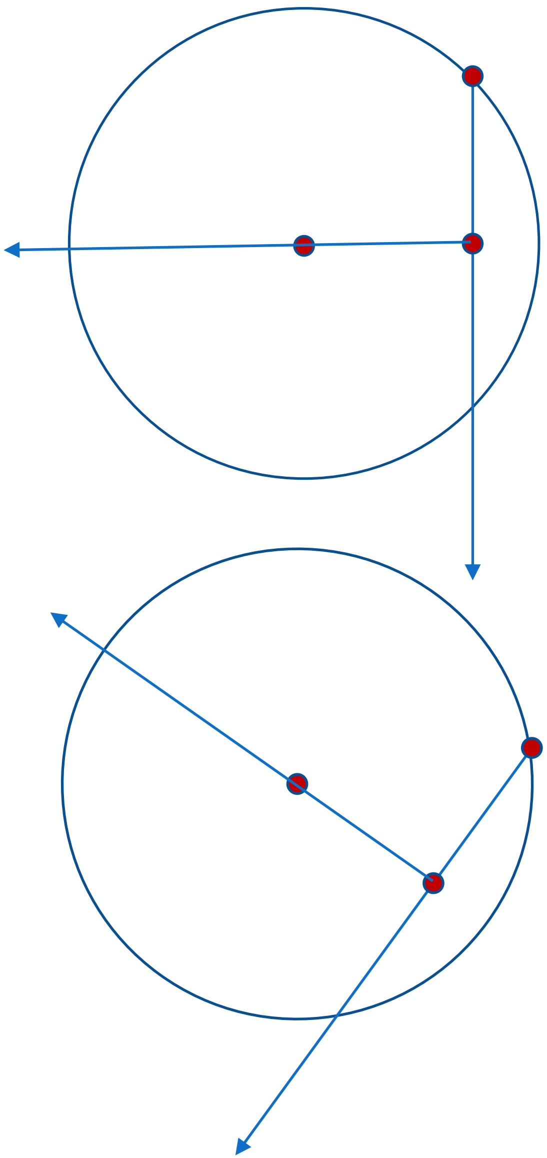
- Depends on choice of basis
- #iter $\propto \kappa$
- Linear convergence,
$$\propto \log \frac{1}{\epsilon}$$
- 1^{st} order oracle
- Just vector ops—
$O(n)$ time per iter

Newton

- Affine invariant
- Bad conditioning OK
- Quadratic convergence,
$$\propto \log \log \frac{1}{\epsilon}$$
- 2^{nd} order oracle
- Matrix inversion—
$O(n^3)$ per iter

Conjugate Gradient Descent

Optimizing a Spherical Quadratic



$$x^{(k+1)} = \arg\min_t f\left(x^{(k)} + te_k\right)$$

$$\Downarrow$$

$$x^{(k+1)} = \arg \min_{x \in x^{(0)} + span(e_0, \ldots, e_k)} f(x)$$

$x = x^{(0)} + \sum_{i=0}^{k} c_i e_i$

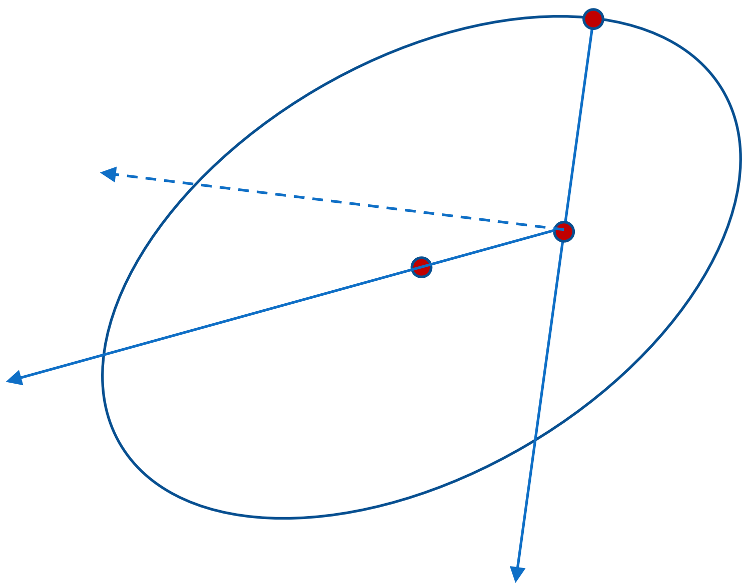
For **orthogonal** $\Delta x^{(0)}, \Delta x^{(1)}, \ldots$

$$x^{(k+1)} = \arg\min_t f\left(x^{(k)} + t\Delta x^{(k)}\right)$$

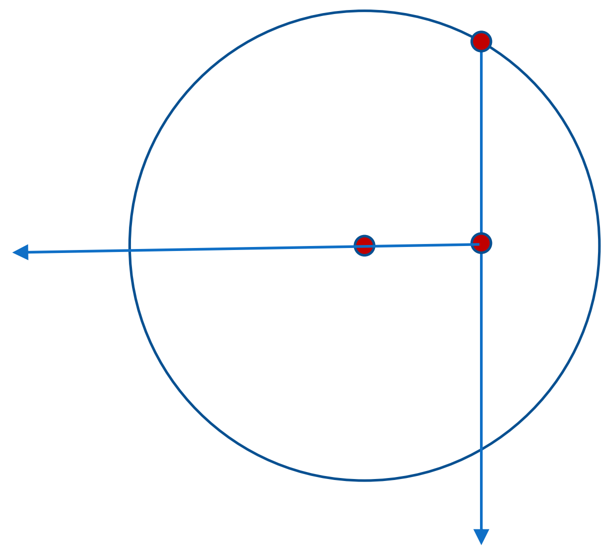
$$\Downarrow$$

$$x^{(k+1)} = \arg \min_{x \in x^{(0)} + span\left(\Delta x^{(0)}, \ldots, \Delta x^{(k)}\right)} f(x)$$

Optimizing a Non-Spherical Quadratic



$f(x) = x^\top H x$



$\tilde{f}(\tilde{x}) = \tilde{x}^\top \tilde{x} = \left(H^{1/2}x\right)^\top (H^{1/2}x)$

$\tilde{x} = H^{1/2}x$

<u>Def</u>: For $H \succ 0$, $\{v_i\}$ are ***H*-conjugate** iff $\forall_{i \neq j}\ v_i^\top H v_j = 0$

- I-conjugate $\Leftrightarrow$ orthogonal
- $\{v_i\}$ are H-conjugate iff $\widetilde{v_i} = H^{1/2} v_i$ are orthogonal

Conjugate Directions

- <u>Def</u>: For $H \succcurlyeq 0$, $\{v_i\}$ are ***H*-conjugate** iff $\forall_{ij}\ v_i^\top H v_j = 0$

- <u>Claim</u>: for $f(x) = x^\top H x + b^\top x$, if $\left\{\Delta x^{(k)}\right\}$ are H-conjugate and

$$x^{(k+1)} = \arg\min_t f\left(x^{(k)} + t\Delta x^{(k)}\right)$$

then $\quad x^{(k+1)} = \displaystyle\arg\min_{x \in x^{(0)} + span\left(\Delta x^{(0)},\ldots,\Delta x^{(k)}\right)} f(x)$

Obtaining Conjugate Directions

- Suppose $\{\Delta x^{(0)}, \dots, \Delta x^{(k-1)}\}$ are H-conjugate
- New "good" direction $d = -\nabla f\left(x^{(k)}\right)$
- How do we find H-conj $\Delta x^{(k)}$ that spans same space, i.e. s.t. $span\left(\Delta x^{(0)}, \dots, \Delta x^{(k-1)}, d\right) = span\left(\Delta x^{(0)}, \dots, \Delta x^{(k-1)}, \Delta x^{(k)}\right)$?
- In transformed space ($\Delta\tilde{x}^{(i)} = H^{1/2}\Delta x^{(i)}$, $\tilde{d} = H^{1/2}d$): find orthogonal basis spanning same space

➔Project $\tilde{d}$ onto subspace orthogonal to $\Delta\tilde{x}^{(0)} \dots \Delta\tilde{x}^{(k-1)}$ (Gram Schmidt)

$$\Delta\tilde{x}^{(k)} = \tilde{d} - \sum_{i=0}^{k-1} \frac{\langle \tilde{d}, \Delta\tilde{x}^{(i)} \rangle}{\langle \Delta\tilde{x}^{(i)}, \Delta\tilde{x}^{(i)} \rangle} \Delta\tilde{x}^{(i)}$$

$$\Delta x^{(k)} = H^{-\frac{1}{2}}\Delta\tilde{x}^{(k)} = d - \sum_{i=0}^{k-1} \frac{d^\top H \Delta x^{(i)}}{\Delta x^{(i)\top} H \Delta x^{(i)}} \Delta x^{(i)}$$

Linear Conjugate Gradient Descent (minimizing a quadratic objective)

$$\min_x f(x) = \frac{1}{2}x^\top H x + b^\top x \quad \equiv \quad \text{solve } Hx + b = 0$$

Init $\quad x^{(0)} \in dom(f)$

Iterate $\quad d^{(k)} = -\nabla f\left(x^{(k)}\right)$

$$\Delta x^{(k)} = d^{(k)} - \sum_{i=0}^{k-1} \frac{{d^{(k)}}^\top H \Delta x^{(i)}}{{\Delta x^{(i)}}^\top H \Delta x^{(i)}} \Delta x^{(i)}$$

$$t^{(k)} = \arg\min_t f\left(x^{(k)} + t\Delta x^{(k)}\right)$$

$$x^{(k+1)} \leftarrow x^{(k)} + t^{(k)} \Delta x^{(k)}$$

Claim: $\Delta x^{(k)} = d^{(k)} + \beta^{(k)} \Delta x^{(k-1)}$

$$\beta^{(k)} = \frac{\langle d^{(k)}, d^{(k)} \rangle}{\langle d^{(k-1)}, d^{(k-1)} \rangle}$$

Linear Conjugate Gradient Descent – Efficient Implementation

$$\min_x f(x) = \frac{1}{2} x^\top H x + b^\top x \quad \equiv \quad \text{solve } Hx + b = 0$$

Init $\quad x^{(0)} \in dom(f)$

Iterate $\quad d^{(k)} = -\nabla f\left(x^{(k)}\right)$

$$\beta^{(k)} = \frac{\langle d^{(k)}, d^{(k)} \rangle}{\langle d^{(k-1)}, d^{(k-1)} \rangle}$$

$$\Delta x^{(k)} = d^{(k)} + \beta^{(k)} \Delta x^{(k-1)}$$

$$t^{(k)} = \arg\min_t f\left(x^{(k)} + t\Delta x^{(k)}\right) = -\frac{\langle \Delta x^{(k)}, \nabla f\left(x^{(k)}\right) \rangle}{\langle \Delta x^{(k)}, H\Delta x^{(k)} \rangle}$$

$$x^{(k+1)} \leftarrow x^{(k)} + t^{(k)} \Delta x^{(k)}$$

Memory: $O(n)$

Time per iteration: $O(n^2)$ in order to calculate $\nabla f(x) = Hx + b$

Number of required iterations: n, and we have $x^{(n)} = x^*$

Total time until $x^{(n)} = x^*$: $O(n^3)$

Conjugate Gradient Descent – analysis for Quadratic Objectives

- $x^{(n)} = x^*$

- Claim: $\left(f\left(x^{(k+1)}\right) - p^*\right) \leq 2\left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)\left(f\left(x^{(k)}\right) - p^*\right)$

$$\approx 2\left(1 - \frac{2}{\sqrt{\kappa}}\right)\left(f\left(x^{(k)}\right) - p^*\right)$$

- Conclusion: #iter to reach ϵ-suboptimality:

$$k = O\left(\sqrt{\kappa}\log\frac{1}{\epsilon}\right)$$

$$\kappa = \frac{\lambda_{max}(H)}{\lambda_{min}(H)}$$

Conjugate Gradient Descent for a Non-Quadratic

- Pretend its quadratic…
- Directions are no longer guaranteed to be conjugate
  - But: if locally quadratic, last few directions are approximately $\nabla^2 f(x)$-conjugate
- Non-exact line-search: formula for $\beta^{(k)}$ doesn't yield projection even if quadratic
  - For quadratic with exact line-search: several exact formulas (all equivalent)
  - For non-quadratic or non-exact-search: it matters which is used
- Warning: $\Delta x^{(k)}$ might not be descent direction!

- Solution: might need to "restart" to $\Delta x^{(k)} = d^{(k)}$ occasionally
  - Every set number of iterations
  - If $\langle \nabla f\left(x^{(k)}\right), \Delta x^{(k)} \rangle \geq 0$, or not negative enough
  - If $\left\|\Delta x^{(k)}\right\|$ small
  - Happens automatically in some formulas for $\beta^{(k)}$

Polak-Ribière Non-Linear Conjugate Gradient Descent

Init	$x^{(0)} \in dom(f)$
Iterate	$d^{(k)} = -\nabla f\left(x^{(k)}\right)$
	$\beta^{(k)} = \frac{\langle d^{(k)}, d^{(k)} - d^{(k-1)} \rangle}{\langle d^{(k-1)}, d^{(k-1)} \rangle}$
	If "reset" (or $k = 0$): $\beta^{(k)} \leftarrow 0$
	$\Delta x^{(k)} = d^{(k)} + \beta^{(k)} \Delta x^{(k-1)}$
	Set $t^{(k)}$ by backtracking (or other) linesearch
	$x^{(k+1)} \leftarrow x^{(k)} + t^{(k)} \Delta x^{(k)}$

Memory: $O(n)$

Time per iteration: calculate $\nabla f(x)$ + $O(n)$

Quasi Newton

Newton's Method

Init	$x^{(0)} \in dom(f)$
Iterate	$\Delta x^{(k)} = -\nabla^2 f\left(x^{(k)}\right)^{-1} \nabla f\left(x^{(k)}\right)$
	Set $t^{(k)}$ by backtracking linesearch with params α, β
	$x^{(k+1)} \leftarrow x^{(k)} + t^{(k)} \Delta x^{(k)}$

- Need 2nd order oracle (access to Hessian)
- Need to invert Hessian at each iteration

How to Approximate $\nabla^2 f(x)^{-1}$

- In one dimension:

$$f''\left(x^{(k)}\right) \approx \frac{f'\left(x^{(k)}\right) - f'\left(x^{(k-1)}\right)}{x^{(k)} - x^{(k-1)}}$$

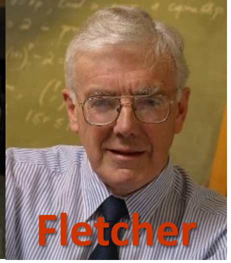
$$\Rightarrow \frac{1}{f''\left(x^{(k)}\right)}\left(f'\left(x^{(k)}\right) - f'\left(x^{(k-1)}\right)\right) \approx \left(x^{(k)} - x^{(k-1)}\right)$$

- In $\mathbb{R}^n$:

$$\nabla^2 f\left(x^{(k)}\right)^{-1} \underbrace{\left(\nabla f\left(x^{(k)}\right) - \nabla f\left(x^{(k-1)}\right)\right)}_{y^{(k)}} \approx \underbrace{\left(x^{(k)} - x^{(k-1)}\right)}_{s^{(k)}}$$

- Solve: $D^{(k)} y^{(k)} = s^{(k)}$
- Use: $\Delta x^{(k)} = -D^{(k)} \nabla f\left(x^{(k)}\right)$

- Problem: underconstrained (n equations, n^2 variables)



BFGS

$$s^{(k)} = x^{(k)} - x^{(k-1)} \qquad y^{(k)} = \nabla f\left(x^{(k)}\right) - \nabla f\left(x^{(k-1)}\right)$$

$$D^{(k)} = \arg\min_D \left\| D - D^{(k-1)} \right\|_{W^{(k)}} \ s.t. \quad D y^{(k)} = s^{(k)}$$

$$= \left(I - \frac{s^{(k)} {y^{(k)}}^\top}{\langle y^{(k)}, s^{(k)} \rangle} \right) D^{(k-1)} \left(I - \frac{y^{(k)} {s^{(k)}}^\top}{\langle y^{(k)}, s^{(k)} \rangle} \right) + \frac{s^{(k)} {s^{(k)}}^\top}{\langle y^{(k)}, s^{(k)} \rangle}$$

$$x^{(k+1)} = x^{(k)} - t^{(k)} D^{(k)} \nabla f\left(x^{(k)}\right)$$

Claim: for a quadratic $f(x) = \frac{1}{2} x^\top H x + b^\top x$, with exact linesearch,

- $D^{(k)} y^{(i)} = s^{(i)}$ for $i = 1..k$

  ➔ $D^{(n)} = H^{-1} = \nabla^2 f(x)^{-1}$

- $\Delta x^{(k)} = -D^{(k)} \nabla f\left(x^{(k)}\right)$ are H-conjugate

  ➔ With $D^{(0)} = I$, this is equivalent to conj grad descent

BFGS

Init $\quad x^{(0)} \in dom(f), D^{(0)}$

Iterate $\quad s^{(k)} = x^{(k)} - x^{(k-1)}$

$$y^{(k)} = \nabla f\left(x^{(k)}\right) - \nabla f\left(x^{(k-1)}\right)$$

$$D^{(k)} = \left(I - \frac{s^{(k)} {y^{(k)}}^{\top}}{\langle y^{(k)}, s^{(k)} \rangle}\right) D^{(k-1)} \left(I - \frac{y^{(k)} {s^{(k)}}^{\top}}{\langle y^{(k)}, s^{(k)} \rangle}\right) + \frac{s^{(k)} {s^{(k)}}^{\top}}{\langle y^{(k)}, s^{(k)} \rangle}$$

$$\Delta x^{(k)} = -D^{(k)} \nabla f\left(x^{(k)}\right)$$

Set $t^{(k)}$ by backtracking (or other) linesearch

$$x^{(k+1)} \leftarrow x^{(k)} + t^{(k)} \Delta x^{(k)}$$

- 1st order Oracle
- Memory: $O(n^2)$
- Time per iteration: eval $\nabla f(x)$ + $O(n^2)$

L-BFGS

$$D^{(k)} = \left(I - \frac{s^{(k)} {y^{(k)}}^\top}{\langle y^{(k)}, s^{(k)} \rangle}\right) D^{(k-1)} \left(I - \frac{y^{(k)} {s^{(k)}}^\top}{\langle y^{(k)}, s^{(k)} \rangle}\right) + \frac{s^{(k)} {s^{(k)}}^\top}{\langle y^{(k)}, s^{(k)} \rangle}$$

Full BFGS:

- keep $2k$ vectors $s^{(1)}, y^{(1)}, \ldots, s^{(k)}, y^{(k)}$,
- Implement $D^{(k)} v$ by unrolling the recursion down to $D^{(0)} = D_{init}$

$$D^{(k)} v = u - \frac{s^{(k)} \langle y^{(k)}, u \rangle}{\langle y^{(k)}, s^{(k)} \rangle} + \frac{s^{(k)} \langle s^{(k)}, v \rangle}{\langle y^{(k)}, s^{(k)} \rangle} \quad \text{where} \quad u = D^{(k-1)} \left(v - \frac{y^{(k)} \langle s^{(k)}, v \rangle}{\langle y^{(k)}, s^{(k)} \rangle} \right)$$

- Need $9k$ vector operations

L-BFGS:

- keep only last $2r$ vectors $s^{(k+1-r)}, y^{(k+1-r)}, \ldots, s^{(k)}, y^{(k)}$
- Unroll the recursion down to $D^{(k-r)}$, but replace $D^{(k-r)} = D_{init}$
- Need $9r$ vector operations
- Memory: $O(rn)$
- Runtime per iteration: eval $\nabla f(x)$ + $O(rn)$

	Memory	Per iter	#iter $\mu \preccurlyeq \nabla^2 \preccurlyeq M$	#iter quadratic
GD	$O(n)$	$\nabla f + O(n)$	$\kappa \log 1/\epsilon$	$\kappa \log 1/\epsilon$
A-GD	$O(n)$	$\nabla f + O(n)$	$\sqrt{\kappa} \log 1/\epsilon$	$\sqrt{\kappa} \log 1/\epsilon$
Conj-GD	$O(n)$	$\nabla f + O(n)$		$\sqrt{\kappa} \log 1/\epsilon$
L-BFGS	$O(rn)$	$\nabla f + O(rn)$		
BFGS	$O(n^2)$	$\nabla f + O(n^2)$		$\sqrt{\kappa} \log 1/\epsilon$
Newton	$O(n^2)$	$\nabla^2 f + O(n^3)$	$\frac{f\left(x^{(0)}\right) - p^*}{\gamma} + \log\log 1/\epsilon$	1

Conjugate Gradient Descent

Init $x^{(0)} \in dom(f)$

Iterate $d^{(k)} = -\nabla f\left(x^{(k)}\right)$

$$\Delta x^{(k)} = d^{(k)} - \sum_{i=0}^{k-1} \frac{{d^{(k)}}^{\mathsf{T}} H \Delta x^{(i)}}{{\Delta x^{(i)}}^{\mathsf{T}} H \Delta x^{(i)}} \Delta x^{(i)}$$

$$= d^{(k)} + \frac{\langle d^{(k)}, d^{(k)} \rangle}{\langle d^{(k-1)}, d^{(k-1)} \rangle} \Delta x^{(k-1)}$$

$$t^{(k)} = \arg\min_t f\left(x^{(k)} + t\Delta x^{(k)}\right)$$

$$x^{(k+1)} \leftarrow x^{(k)} + t^{(k)} \Delta x^{(k)}$$

$$= \arg\min_{x = x^{(0)} + span\left(d^{(0)}, \dots, d^{(k)}\right)} f(x)$$

$$= \arg\min_{x = x^{(k)} + span\left(d^{(k)}, \Delta x^{(k-1)}\right)} f(x)$$

Accelerated Gradient Descent

Nesterov '83

Init	$x^{(0)} \in dom(f), y^{(0)} = x^{(0)}$
Iterate	$x^{(k+1)} = y^{(k)} - \frac{1}{M}\nabla f\left(y^{(k)}\right)$
	$y^{(k+1)} = \left(1 + \frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right) x^{(k+1)} - \frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1} x^{(k)}$

$$x^{(k+1)} = \overbrace{x^{(k)} + \frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\underbrace{\left(x^{(k)} - x^{(k-1)}\right)}_{\textbf{momentum}}}^{y^{(k)}} - \frac{1}{M}\nabla f\left(y^{(k)}\right)$$

- Theorem: $f\left(x^{(k)}\right) \leq p^* + \epsilon$ after at most

$$k \leq \sqrt{\kappa}\log\frac{(\mu+M)\left\|x^{(0)}-x^*\right\|_2^2}{2\epsilon}$$

- No lineasearch—need to use precise step size and "momentum"

Gradient Descent w/ Momentum

$$x^{(k+1)} \leftarrow x^{(k)} + m^{(k)}\left(x^{(k)} - x^{(k-1)}\right) - t^{(k)}\nabla f\left(x^{(k)}\right)$$

or

$$x^{(k+1/2)} = x^{(k)} + m^{(k)}\left(x^{(k)} - x^{(k-1)}\right)$$

	Memory	Per iter	#iter $\mu \preccurlyeq \nabla^2 \preccurlyeq M$	#iter $\nabla^2 \preccurlyeq M$	#iter quadratic
GD	$O(n)$	$\nabla f + O(n)$	$\kappa \log 1/\epsilon$	$\frac{M\|x^*\|^2}{\epsilon}$	$\kappa \log 1/\epsilon$
A-GD	$O(n)$	$\nabla f + O(n)$	$\sqrt{\kappa} \log 1/\epsilon$	$\sqrt{\frac{M\|x^*\|^2}{\epsilon}}$	$\sqrt{\kappa} \log 1/\epsilon$
C-GD	$O(n)$	$\nabla f + O(n)$			$\sqrt{\kappa} \log 1/\epsilon$
L-BFGS	$O(rn)$	$\nabla f + O(rn)$			
BFGS	$O(n^2)$	$\nabla f + O(n^2)$			$\sqrt{\kappa} \log 1/\epsilon$
Newton	$O(n^2)$	$\nabla^2 f + O(n^3)$	$\frac{f(x^{(0)}) - p^*}{\gamma} + \log\log 1/\epsilon$		1

Reduce to strongly convex: optimize $f_\lambda(x) = f(x) + \frac{\lambda}{2}\|x\|^2$ with $\lambda = \frac{\epsilon}{\|x\|^2}$