Lecture 6: Smooth unconstrained minimization

- terminology
- general descent method
- gradient & steepest descent methods
- Newton's method
- quasi-Newton methods
- self-concordance & Newton's method

Terminology

unconstrained minimization problem

$$\text{minimize} \ f(x)$$

$f : \mathbf{R}^n \to \mathbf{R}$, convex, differentiable
(hence $\mathbf{dom}\, f$ is open . . .)

minimizing sequence: $x^{(k)}$, $k \to \infty$

$$f(x^{(k)}) \to f^\star$$

optimality condition

$$\nabla f(x^\star) = 0$$

set of nonlinear equations, usually no analytical solution; more generally, if $\nabla^2 f(x) \succeq mI$, then

$$f(x) - f^\star \leq \frac{1}{2m} \|\nabla f(x)\|^2$$

. . . yields stopping criterion (if you know m)

Examples

unconstrained quadratic minimization

$$\text{minimize } x^T P x + 2q^T x + r$$

$(P = P^T \succeq 0)$

unconstrained geometric programming

$$\text{minimize } \log \sum_{i=1}^{m} e^{a_i^T x + b_i}$$

analytic center of linear inequalities

$$\text{minimize } -\sum_i \log(b_i - a_i^T x)$$

$(\mathbf{dom}\, f = \{x | a_i^T x < b_i,\ i = 1, \ldots, m\})$

Descent method

given starting point $x \in \mathbf{dom}\, f$
repeat
1. *Compute a search direction* v
2. *Line search.* Choose step size $t > 0$
3. *Update.* $x := x + tv$

until stopping criterion is satisfied

Descent method: $f(x^{(k+1)}) < f(x^{(k)})$

Since f convex, v must be a **descent direction:** $\nabla f(x^{(k)})^T v^{(k)} < 0$

examples

- $v^{(k)} = -\nabla f(x^{(k)})$
- $v^{(k)} = -H^{(k)} \nabla f(x^{(k)})$, $H^{(k)} = H^{(k)T} \succ 0$
- $v^{(k)} = -\nabla^2 f(x^{(k)})^{-1} \nabla f(x^{(k)})$

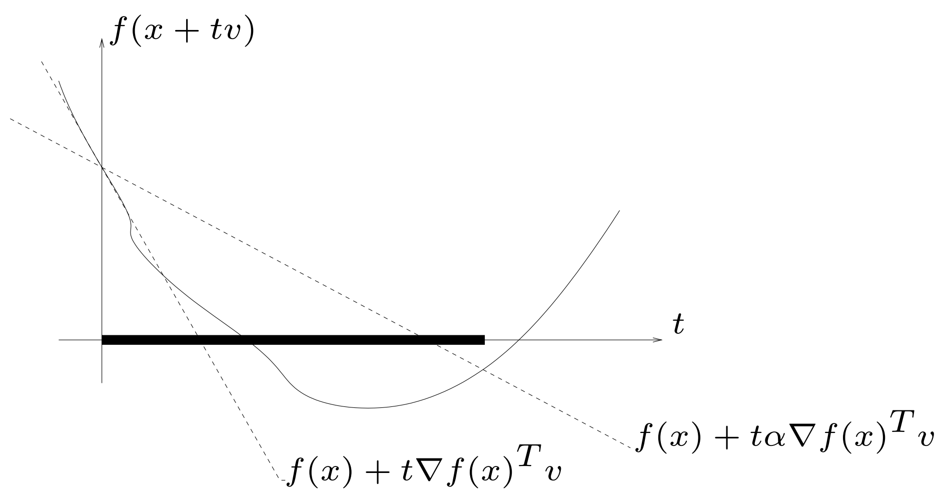
Line search types

two simple & effective line search types:

exact line search: $t = \mathrm{argmin}_{s>0} f(x + sv)$

backtracking line search $(0 < \beta < 1,\ 0 < \alpha < 0.5)$

- starting with $t = 1$, $t := \beta t$
- until $f(x + tv) \le f(x) + t\alpha\nabla f(x)^T v$



Gradient method

given starting point $x \in \mathbf{dom}\, f$

repeat

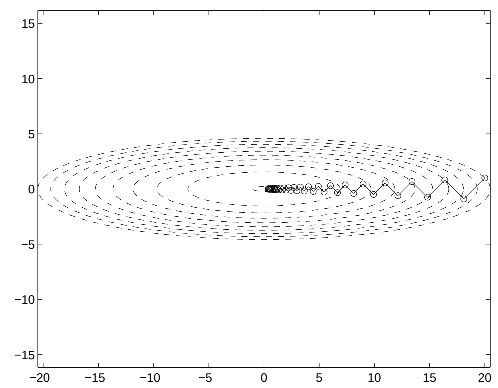
1. *Compute search direction* $v = -\nabla f(x)$
2. *Line search.* Choose step size t
3. *Update.* $x := x + tv$

until stopping criterion is satisfied

- converges with exact or bactracking line search
- can be very slow
- rarely used in practice

Example

$$\text{minimize } \frac{1}{2}(x_1^2 + M x_2^2)$$



where $M > 0$; optimal point is $x^\star = 0$

- use exact line search
- start at $x^{(0)} = (M, 1)$ (to simplify formulas)

iterates are then

$$x^{(k)} = \left(M \left(\frac{M-1}{M+1} \right)^k, \ \left(-\frac{M-1}{M+1} \right)^k \right)$$

convergence is

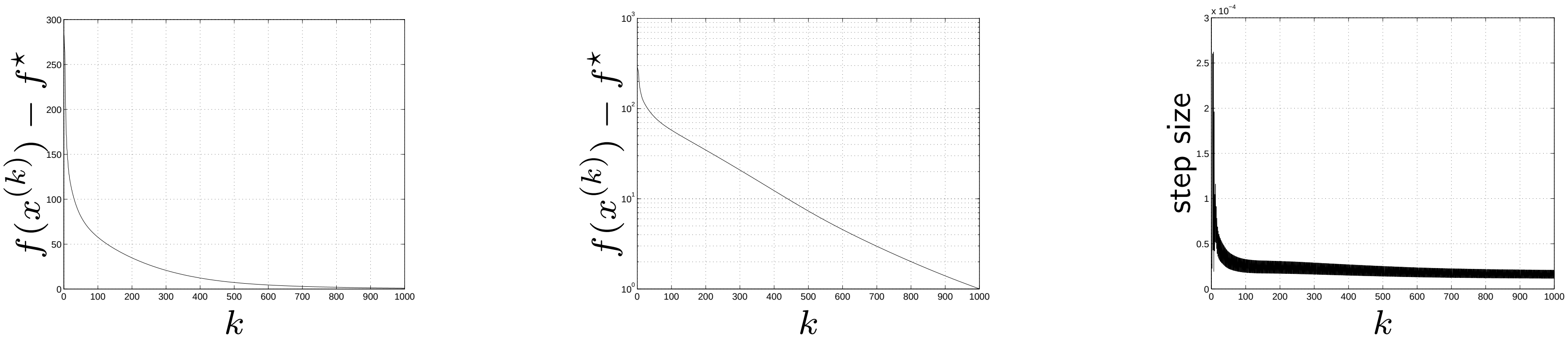
- fast if M close to 1
- slow, zig-zagging if $M \gg 1$ or $M \ll 1$

Numerical example: gradient method

$$\text{minimize } c^T x - \sum_{i=1}^{m} \log(a_i^T x + b_i)$$

$m = 100$, $n = 50$

gradient method with exact line search



slow convergence; zig-zagging

Steepest descent direction

first-order approximation of f at x:

$$f(x+z) \approx f(x) + \nabla f(x)^T z$$

$\nabla f(x)^T z$ gives approximate decrease in f for (small) step z

steepest descent direction for general norm $\|\cdot\|$:

$$v_{\mathrm{sd}} = \mathrm{argmin}\{\nabla f(x)^T v \mid \|v\| = 1\}$$

gives greatest (approximate) decrease in f, per length of step (measured by $\|\cdot\|$)

Euclidean norm: $v_{\mathrm{sd}} = -\nabla f(x)/\|\nabla f(x)\|$

quadratic norm: $\|z\|_P = \left(z^T P z\right)^{1/2}$, $P = P^T \succ 0$

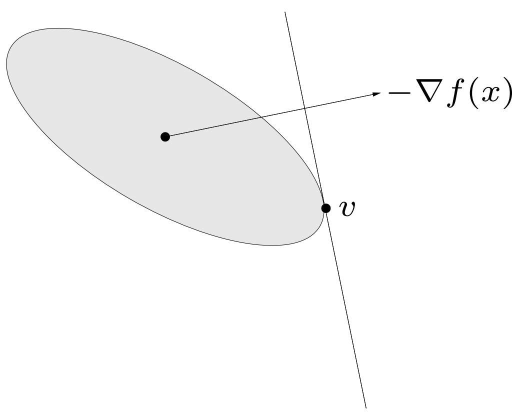
$$v_{\mathrm{sd}} = -\left(\nabla f(x)^T P^{-1} \nabla f(x)\right)^{-1/2} P^{-1} \nabla f(x)$$

can express $v_{\rm sd}$ as

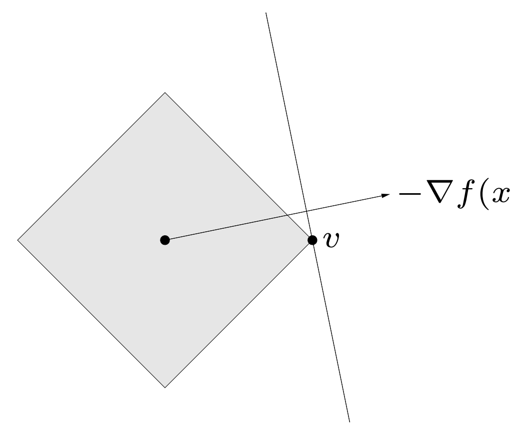
$$v_{\rm sd} = \mathop{\rm argmin}\{\nabla f(x)^T v \mid \|v\| \leq 1\}$$

geometric interpretation: go as far as possible in direction $-\nabla f(x)$, while staying in unit ball

quadratic norm:



ℓ_1-norm:



Steepest descent method

given starting point $x \in \mathbf{dom}\, f$

repeat

1. *Compute steepest descent direction*
   $v_{\rm sd} = \mathrm{argmin}\{\nabla f(x)^T v \mid \|v\| = 1\}$
2. *Line search.* Choose a step size t
3. *Update.* $x := x + tv$

until stopping criterion is satisfied

- converges with exact or backtracking line search
- sometimes $v_{\rm sd}$ is scaled between 1 and 2
- can be very slow
- used in special cases where v is cheap to compute

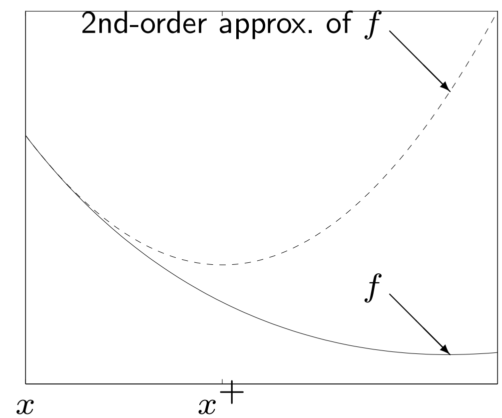
The Newton step

the *Newton step* (at x) is

$$v = -\nabla^2 f(x)^{-1} \nabla f(x)$$

the *Newton iteration* (at x) is

$$x^+ = x + v = x - \nabla^2 f(x)^{-1} \nabla f(x)$$



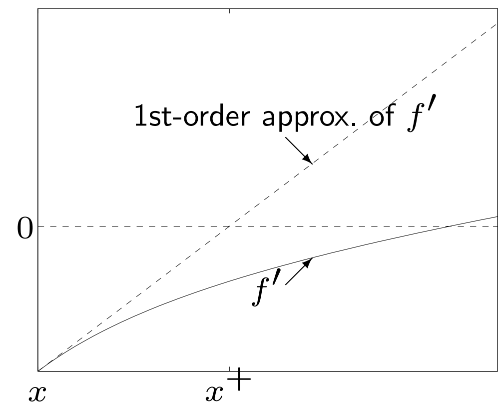
interpretations: $y = x^+$

- minimizes 2nd order expansion of f (at x),

$$f(x)+\nabla f(x)^T(y-x)+\frac{1}{2}(y-x)^T\nabla^2 f(x)(y-x)$$

- solves linearized optimality condition:

$$0 = \nabla f(x) + \nabla^2 f(x)(y - x)$$



Local convergence of Newton iteration

assumptions: $\nabla^2 f(x) \succeq mI$ and Hessian satisfies Lipschitz condition:

$$\left\| \nabla^2 f(x) - \nabla^2 f(y) \right\| \leq L\|x - y\|$$

(L small means f nearly quadratic)

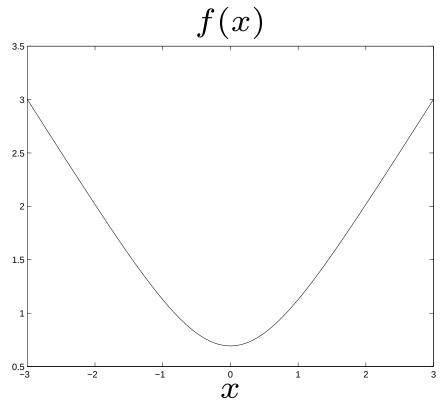
result

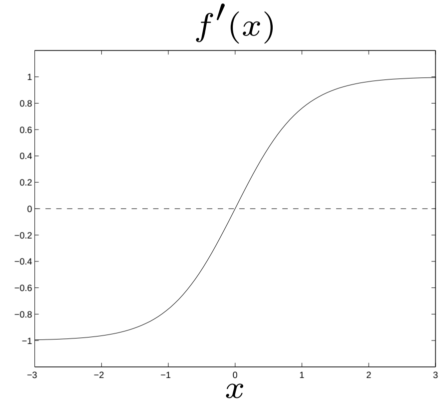
$$\frac{L}{2m^2} \left\| \nabla f(x^+) \right\| \leq \left(\frac{L}{2m^2} \|\nabla f(x)\| \right)^2$$

- region of **quadratic convergence**: $\|\nabla f(x)\|$ (hence, $f(x) - f^\star$) decreases very rapidly if
$$\left\| \nabla f(x^{(0)}) \right\| < m^2/L$$
- bound on #iterations for accuracy $f(x) - f^\star \leq \epsilon$: $\log_2 \log_2(\epsilon_0/\epsilon), \quad \epsilon_0 = m^3/L^2$
- practical rule of thumb: 5–6 iterations

Global behavior of Newton iteration

Newton iteration can diverge **example:** $f(x) = \log(e^x + e^{-x})$, start at $x^{(0)} = 1.1$





k	$x^{(k)}$	$f(x^{(k)}) - f^\star$
1	$-1.129 \cdot 10^0$	$5.120 \cdot 10^{-1}$
2	$1.234 \cdot 10^0$	$5.349 \cdot 10^{-1}$
3	$-1.695 \cdot 10^0$	$6.223 \cdot 10^{-1}$
4	$5.715 \cdot 10^0$	$1.035 \cdot 10^0$
5	$-2.302 \cdot 10^4$	$2.302 \cdot 10^4$

Newton's method

given starting point $x \in \mathbf{dom}\, f$

repeat

1. *Compute Newton direction*
   $v = -\nabla^2 f(x)^{-1} \nabla f(x)$
2. *Line search.* Choose a step size t
3. *Update.* $x := x + tv$

until stopping criterion is satisfied

(also called *damped* or *guarded* Newton method)

- global convergence with backtracking or exact line search
- quadratic local convergence
  (hence, stopping criterion not an issue)

Affine invariance of Newton method

- use new coords $x = T\bar{x}$, $\det T \neq 0$
- apply Newton to $g(\bar{x}) = f(T\bar{x})$
- then $x^{(k)} = T\bar{x}^{(k)}$

$e.g.$, Newton method not affected by variable scaling (cf. gradient, steepest descent)

Convergence analysis

assumptions: $mI \preceq \nabla^2 f(x) \preceq MI$ and Lipschitz condition

$$\left\|\nabla^2 f(x) - \nabla^2 f(y)\right\| \leq L\|x - y\|$$

results: two phases

1. damped Newton phase: $\|\nabla f(x)\| \geq \eta_1$: $f(x^+) \leq f(x) - \eta_2$, hence

$$\#\text{iterations} \leq \eta_2^{-1}(f(x^{(0)}) - f^\star)$$

2. quadratically convergent phase: $\|\nabla f(x)\| < \eta_1$

$$\#\text{iterations} \leq \log_2 \log_2(\epsilon_0/\epsilon)$$

total #iterations bounded by

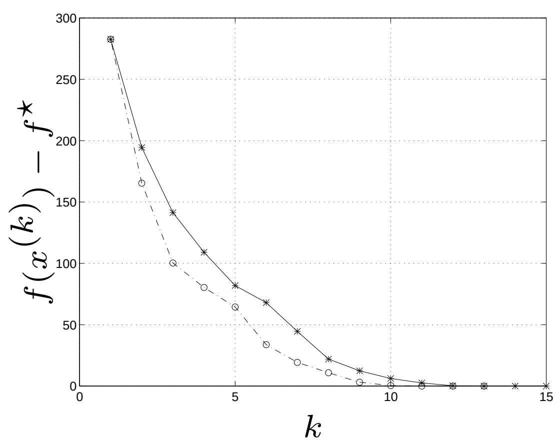
$$\eta_2^{-1}(f(x^{(0)}) - f^\star) + \log_2 \log_2(\epsilon_0/\epsilon)$$

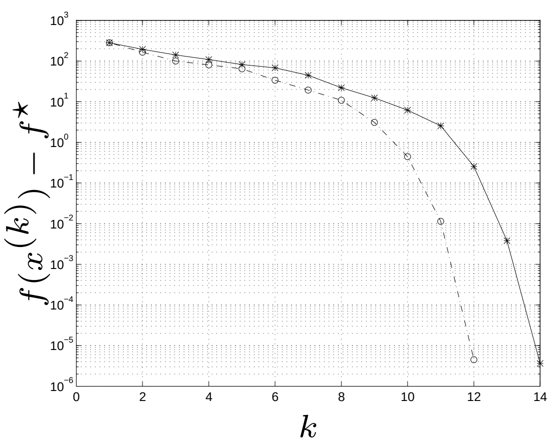
η_1, η_2, ϵ_0 depend on m, M, L (and α, β for backtracking)

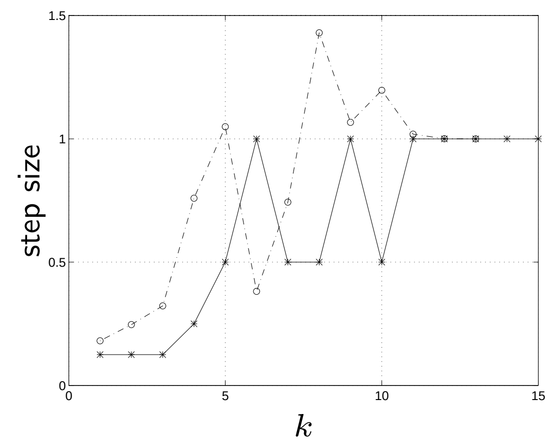
Numerical example: Newton method

$$\text{minimize } c^T x - \sum_{i=1}^{m} \log(a_i^T x + b_i)$$

$m = 100$, $n = 50$







solid line: backtracking ($\beta = 0.5$, $\alpha = 0.2$); dashed line: exact line search; (iters are more expensive than gradient method)

Quasi-Newton methods

idea: replace $\nabla^2 f(x)$ by approximation H

> **given** starting point $x \in \mathbf{dom}\, f$, $H \succ 0$
> **repeat**
> 1. *Compute quasi-Newton direction.* $v = -H^{-1}\nabla f(x)$
> 2. *Line search.* Choose a step size t
> 3. *Update* H.
> 4. *Update* x. $x := x + tv$
>
> **until** stopping criterion is satisfied

many update rules $H \rightarrow H^+$, which all satisfy:

- $H = H^T \succ 0$
- secant condition: $\nabla f(x^+) - \nabla f(x) = H^+(x^+ - x)$
- $H^{-1}\nabla f(x)$ more easily computed (than $\nabla^2 f(x)^{-1}\nabla f(x)$)

advantages (compared to Newton method)

- don't need to evaluate $\nabla^2 f(x)$
- v can be computed in $O(n^2)$ operations

disadvantage (compared to Newton method)

- local convergence fast, but not quadratic

quasi-Newton methods

- converge in n steps for (cvx) quadratic f on $\mathbf{R}^n$
- widely used in practice

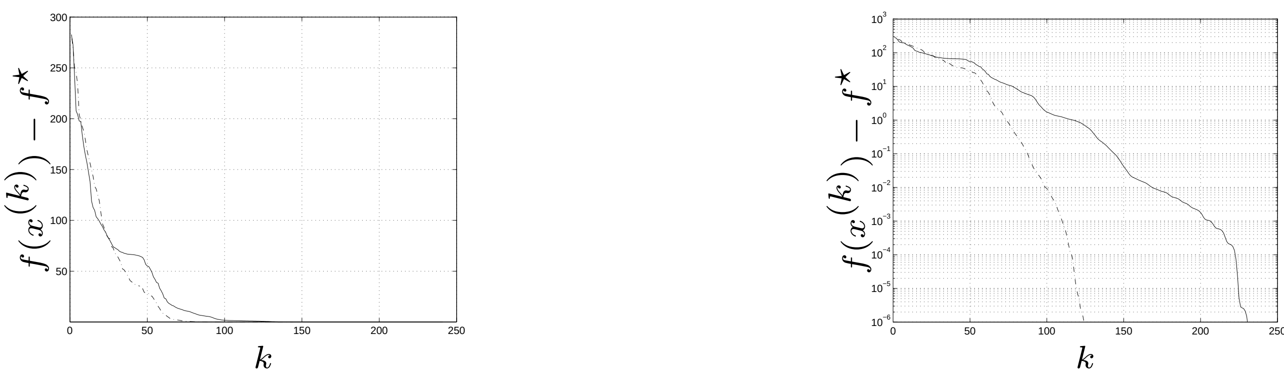
most common is Broyden-Fletcher-Goldfarb-Shanno (BFGS):

$$y = \nabla f(x^+) - \nabla f(x), \quad s = x^+ - x, \quad H^+ = H + \frac{yy^T}{y^T s} - \frac{Hss^T H}{s^T H s}$$

Numerical example: BFGS method

$$\text{minimize } c^T x - \sum_{i=1}^{m} \log(a_i^T x + b_i)$$

$m = 100$, $n = 50$



solid line: backtracking ($\beta = 0.5$, $\alpha = 0.2$); dashed line: exact line search

when comparing with Newton method, remember:

- BFGS is $O(n^2)$ per iter (plus finding ∇f)
- Newton is $O(n^3)$ per iter (plus finding ∇f, $\nabla^2 f$)

Self-concordance: motivation

drawbacks of classical analysis of Newton's method:

- Newton's method is affinely invariant, but convergence analysis is not
- never know m, M, L in practice
- m, M, L can depend on starting point

Nesterov & Nemirovsky's analysis of Newton's method

- is affinely invariant
- involves no unknown constants
- is valid for many (but not all) functions f

Self-concordance: definition

(Nesterov & Nemirovsky)

- function $f : \mathbf{R} \to \mathbf{R}$ is self-concordant if

$$\left| f'''(t) \right| \leq 2 f''(t)^{3/2}$$

- $f : \mathbf{R}^n \to \mathbf{R}$ is self-concordant if restriction to arbitrary line is self-concordant

SC condition

- limits third derivative in terms of second
- is affinely invariant: f is SC $\Longleftrightarrow$ $f(Tx)$ is SC

examples of self-concordant fcts

- linear, (convex) quadratic functions
- $-\log x$ on $\{x|x > 0\}$
- $-\log\det X$ on $\{X|X = X^T \succ 0\}$
- $-\log(t^2 - x^T x)$ on $\{\ (x,t)\ |\ \|x\| < t\ \}$

simple properties:

- affine transformation of domain: f SC $\Longrightarrow g(z) = f(Az + b)$ SC
- sums: f, $\tilde{f}$ SC $\Longrightarrow f + \tilde{f}$ SC
- scaling: f SC, $\alpha \geq 1 \Longrightarrow \alpha f$ SC

hence, *e.g.*,

- $-\sum_i \log(b_i - a_i^T x)$ is SC
- $-\log\det\left(F_0 + x_1 F_1 + \cdots + x_n F_n\right)$ is SC

Convergence analysis via SC

Newton method, with backtracking or exact line search:

$$\#\text{iterations} \leq \frac{f(x^{(0)}) - f^\star}{\eta_2} + \log_2 \log_2(2/\epsilon)$$

where η_2 depends only on backtracking parameters:

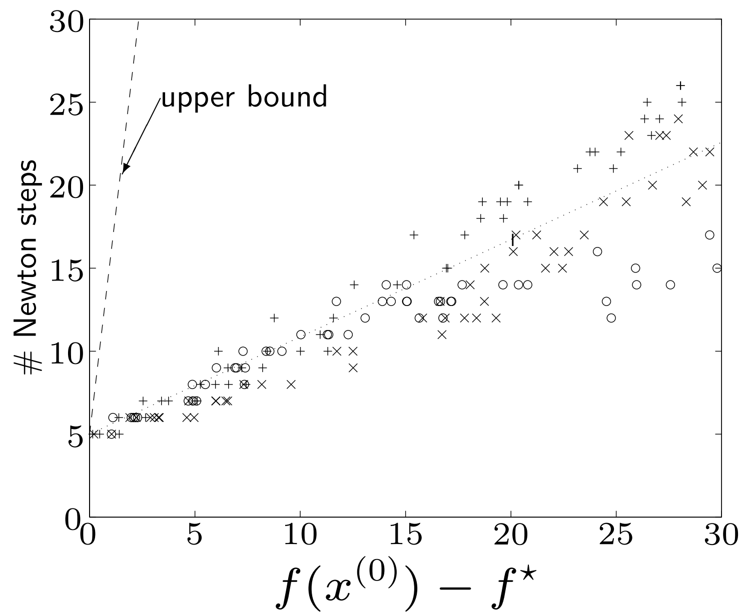
$$\eta_2 = \beta \frac{\alpha(1/2 - \alpha)^2}{5 - 2\alpha}$$

$e.g.$, for $\alpha = 0.2$, $\beta = 0.7$, we have $1/\eta_2 \approx 365$

(a more refined analysis yields smaller bound)

example:

$$f(x) = \log\det\left(F_0 + x_1 F_1 + \cdots + x_n F_n\right)^{-1}$$



conclusion:

- $f(x^{(0)}) - f^\star$ gives upper bound on #iterations
- $f(x^{(0)}) - f^\star$ is also good measure in practice