Nonlinear Optimization

Part III
Numerical algorithms

Instituto Superior Técnico – Carnegie Mellon University

jxavier@isr.ist.utl.pt

Unconstrained minimization

Optimization problem:

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

Assumptions:

- $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ is convex
- $\text{dom}\, f = \{x : f(x) < +\infty\}$ is open
- f is closed
- f is twice continuously differentiable
- problem is solvable: $\exists\, x^\star : f(x^\star) = p^\star := \inf\{f(x) : x\}$

Example: quadratic ($A \succeq 0$)

$$\text{minimize} \quad x^\top A x + b^\top x + c$$

- dom $f = \mathbb{R}^n$
- closed
- bounded below iff $b \in \text{span}(A)$

Example: log-sum-exp

$$\text{minimize} \quad \log\left(\sum_{i=1}^{m} e^{a_i^\top x + b_i}\right)$$

- dom $f = \mathbb{R}^n$
- closed
- bounded below iff $0 \in \text{co}\,\{a_i\}$

Example: log barrier for polyhedron

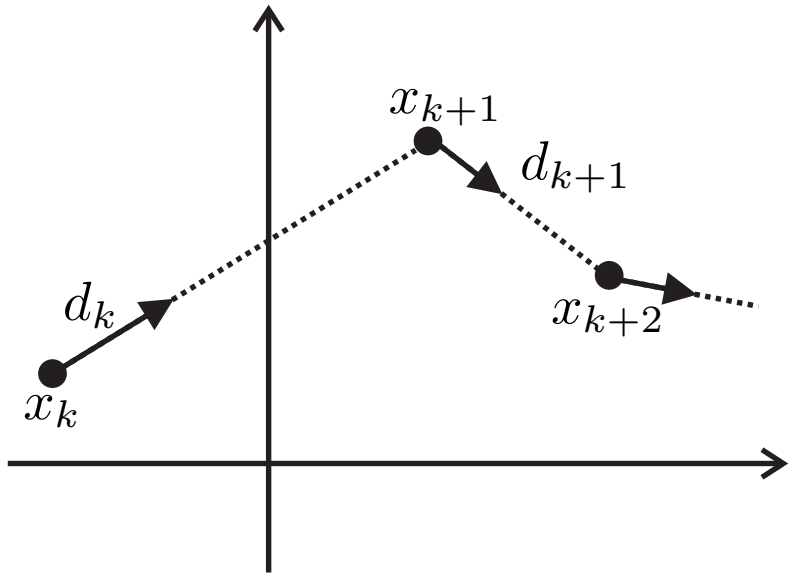
$$\text{minimize} \quad \sum_{i=1}^{m} -\log\left(b_i - a_i^\top x\right)$$

- dom $f = \{x \,:\, Ax < b\}$
- closed
- bounded below if there exists $\lambda > 0$, $\sum_k \lambda_k a_k = 0$

Algorithm produces a minimizing sequence: $x_1, x_2, x_3, \ldots$ such that

$$f(x_k) \downarrow p^\star$$

Basic iteration of line-search algorithms: $x_{k+1} = x_k + t_k d_k$



- d_k is the search direction
- $t_k > 0$ is the step size

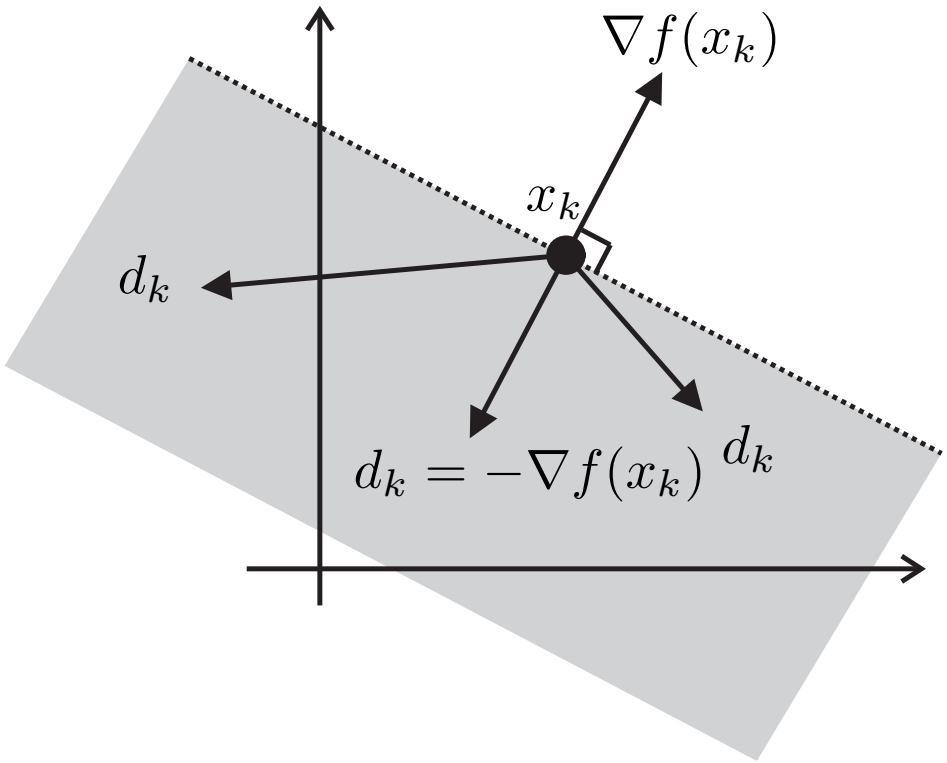
Definition (Descent direction) d_k is a descent direction for f at a non-stationary point $x_k \in \text{dom}\, f$ if

$$\langle \nabla f(x_k), d_k \rangle < 0$$

Example: $d_k = -\nabla f(x_k)$ is a descent direction because

$$\langle \nabla f(x_k), d_k \rangle = -\left\| \nabla f(x_k) \right\|^2 < 0$$

Infinite number of choices for descent directions d_k:



Important: if d_k is a descent direction, then

$$\exists_{\overline{t}>0} \; : \; f(x_k + t d_k) < f(x_k) \text{ for all } t \in]0, \overline{t}]$$

Prototype of a a line-search algorithm:

1. **initialization** ▶ choose $x_0 \in \mathrm{dom}\, f$ and tolerance $\epsilon > 0$
2. ▶ set $k = 0$
3. **loop** ▶ compute $g_k = \nabla f(x_k)$
4. ▶ if $\|g_k\| < \epsilon$ stop
5. ▶ compute descent direction d_k
6. ▶ compute step size t_k
7. ▶ update $x_{k+1} = x_k + t_k d_k$
8. ▶ $k \leftarrow k + 1$ and return to loop

Line 6 is called the line search subproblem

- exact line search:

$$t_k \in \arg\min_{t \geq 0} \phi(t) := f(x_k + t d_k)$$

- Armijo's rule or backtracking line search ($0 < \alpha < 0.5, 0 < \beta < 1$):

  $t = 1$ and while $f(x_k + t d_k) > f(x_k) + t\alpha \nabla f(x_k)^\top d_k$ do $t := \beta t$

Theorem (Convergence of steepest descent method) Let $x_0 \in \text{dom}\, f$ and suppose there exists $m > 0$ such that

$$\nabla^2 f(x) \succeq mI \quad \text{for all } x \in S_{f(x_0)} f.$$

Then,

$$x_k \longrightarrow x^\star.$$

Example:

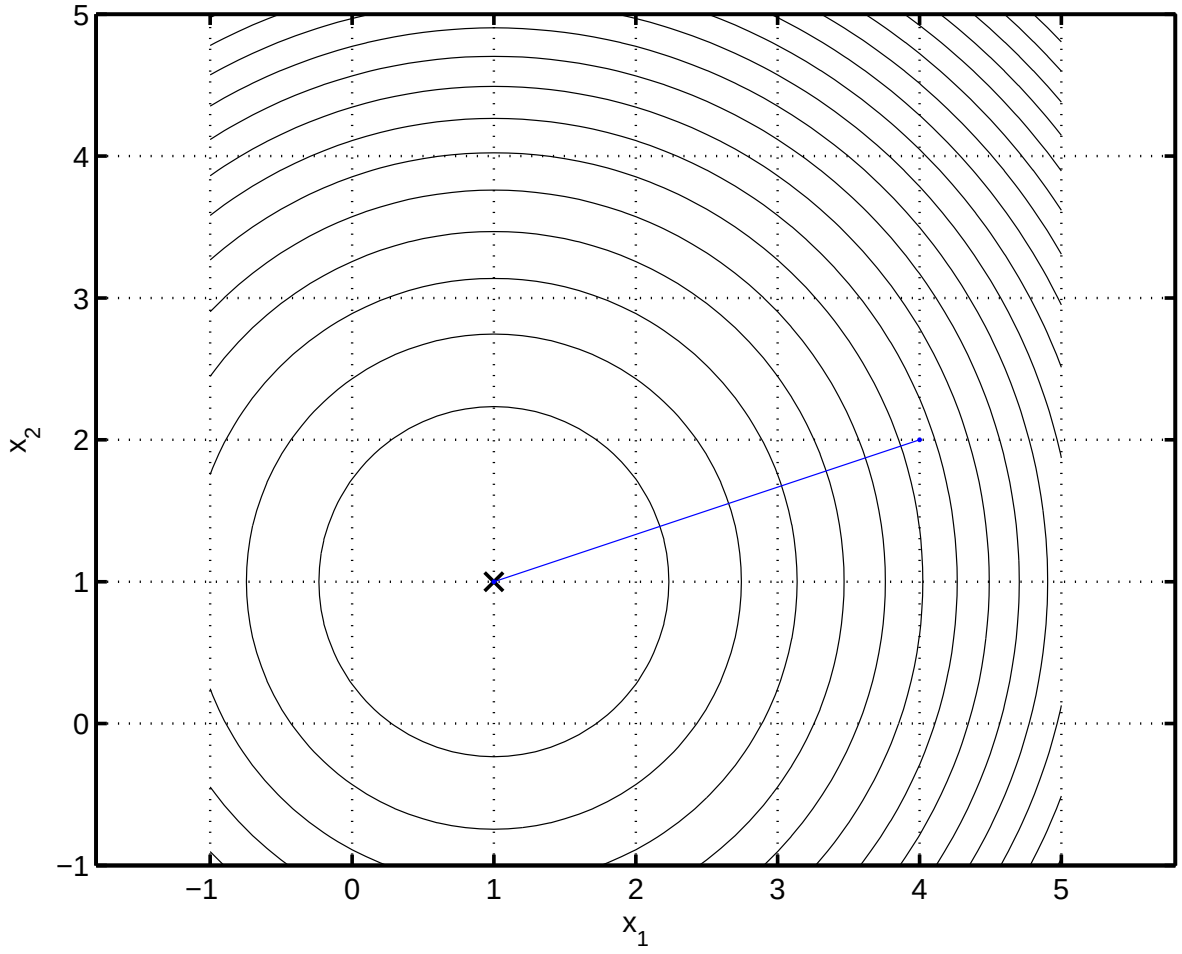
- $f : \mathbb{R}^2 \to \mathbb{R}$

$$f(x) = \frac{1}{2}(x-c)^\top A(x-c)$$

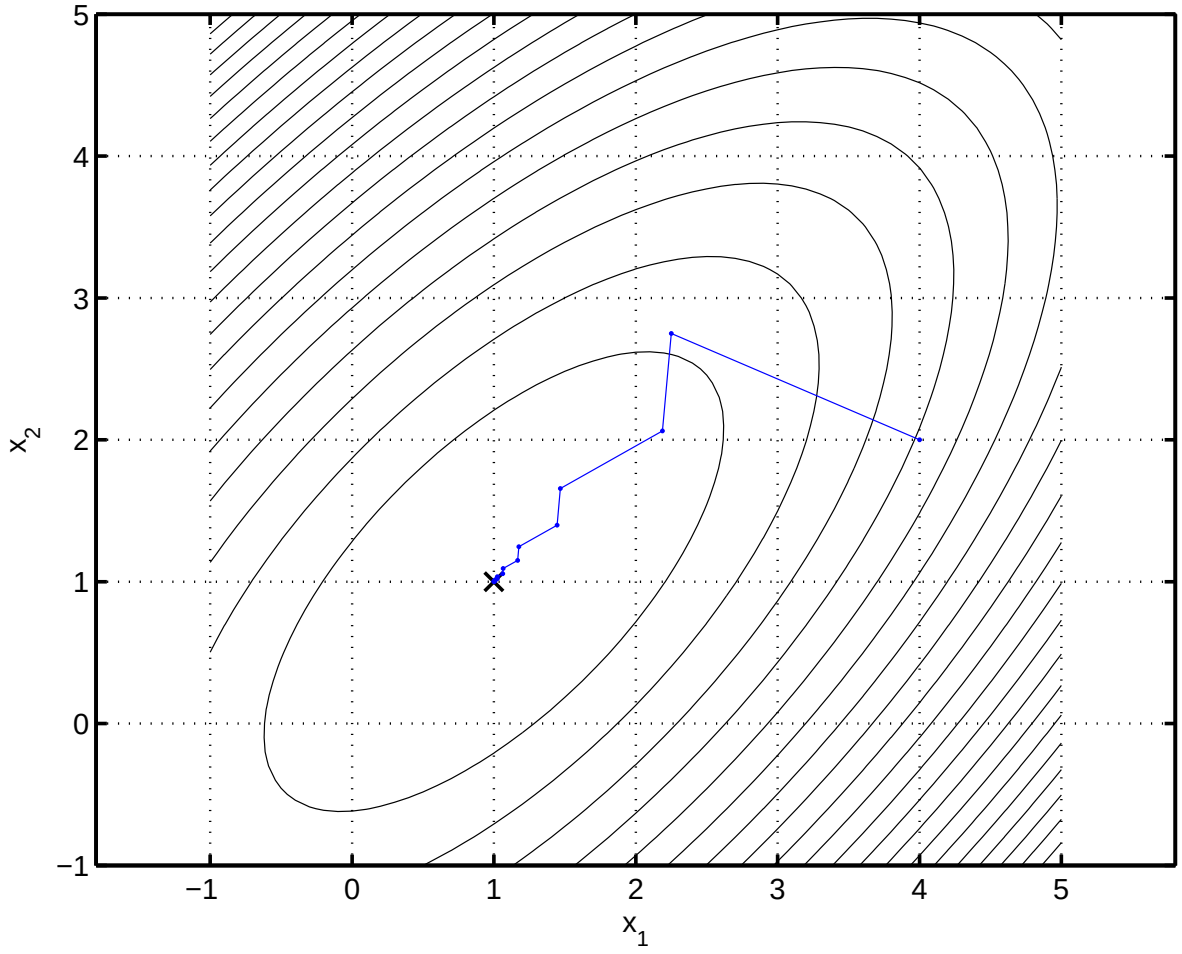
where

$$c = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \qquad A = Q \begin{bmatrix} 1 & 0 \\ 0 & \kappa \end{bmatrix} Q^\top \qquad Q = \begin{bmatrix} \cos\left(\frac{\pi}{4}\right) & -\sin\left(\frac{\pi}{4}\right) \\ \sin\left(\frac{\pi}{4}\right) & \cos\left(\frac{\pi}{4}\right) \end{bmatrix}$$

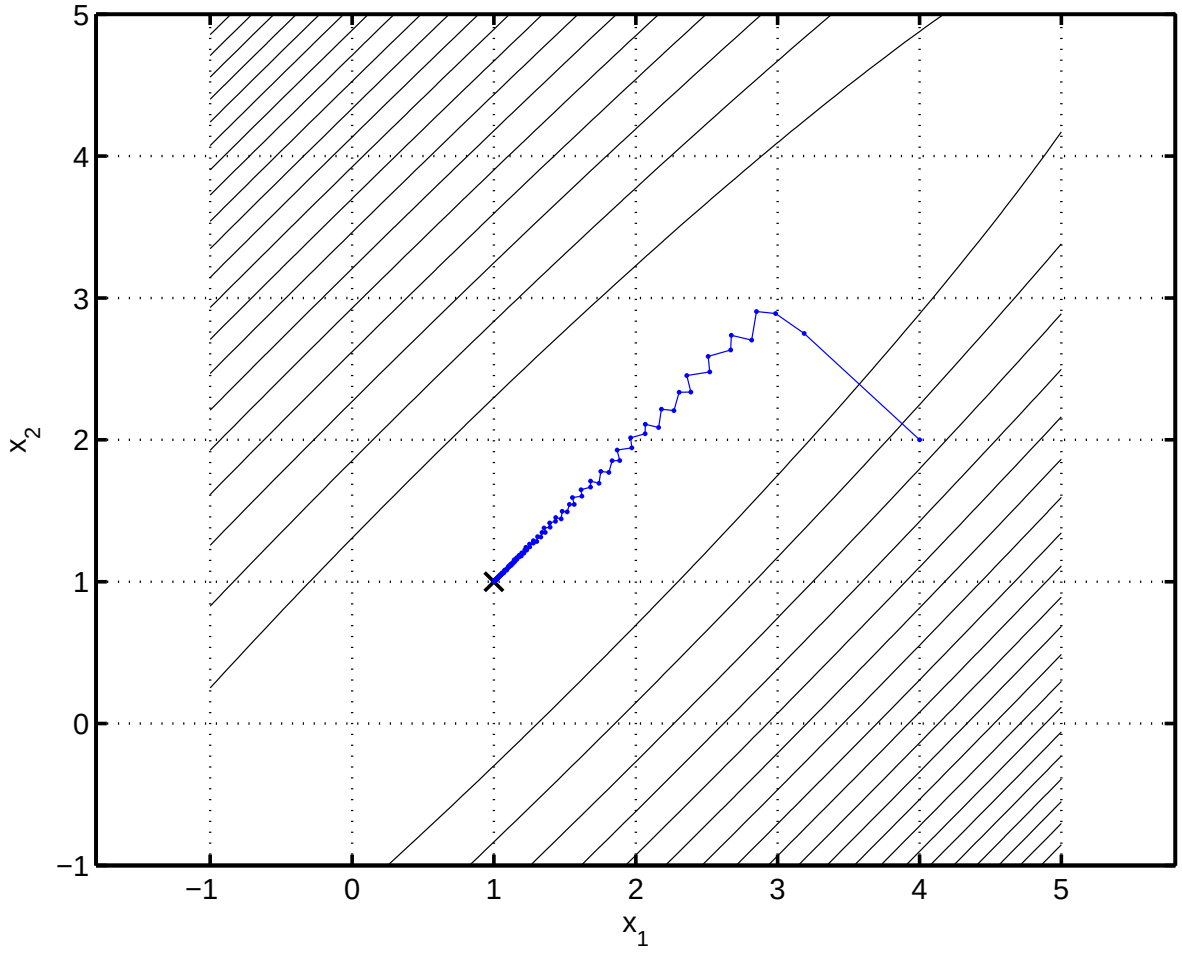
- $x^\star = (1,1)$ and $\kappa\left(\nabla^2 f(x^\star)\right) = \kappa \geq 1$
- $x_0 = (4,2)$, $\epsilon = 10^{-6}$, $\alpha = 10^{-4}$, $\beta = 0.5$



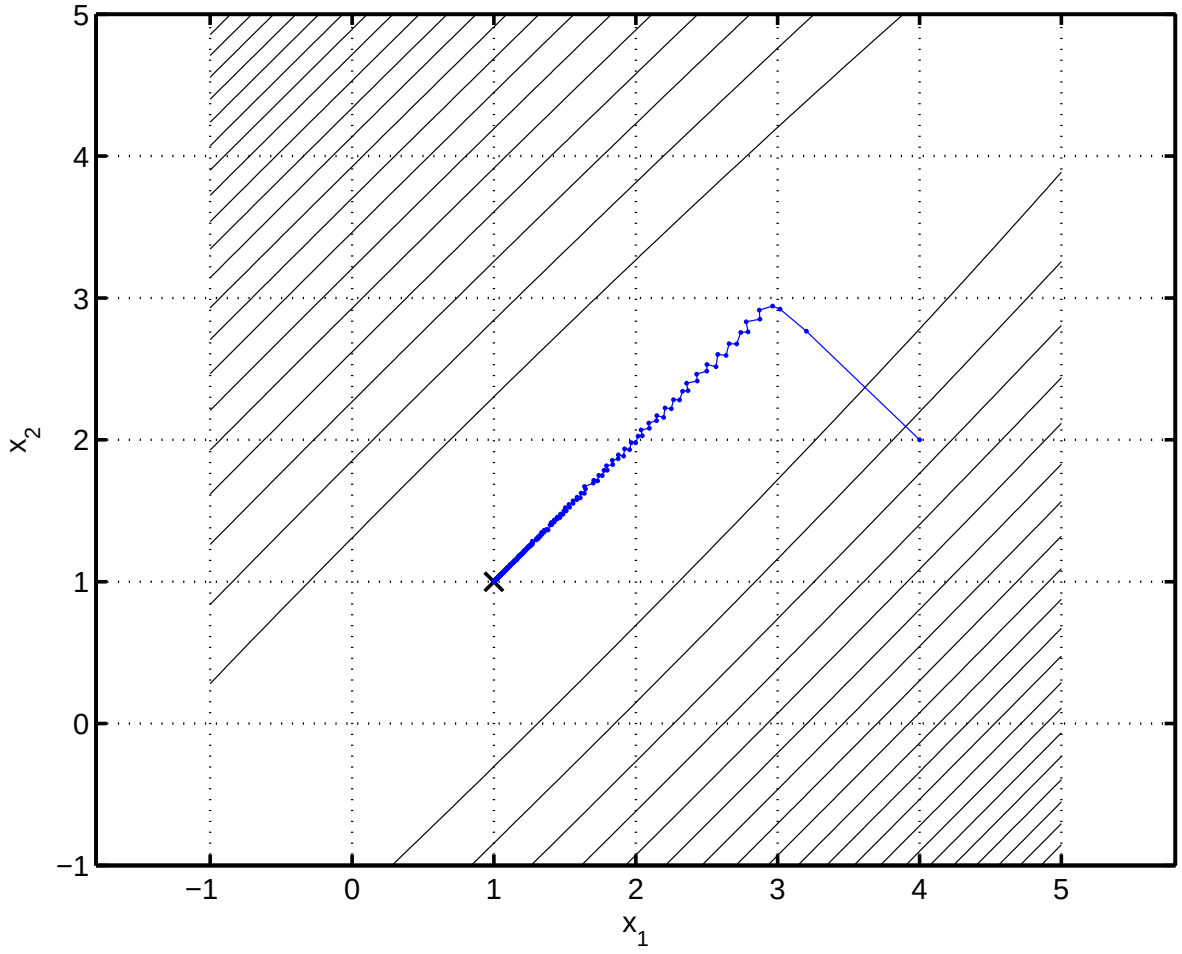
$\kappa = 1$ (number of iterations: 1)



$\kappa = 5$ (number of iterations: 32)

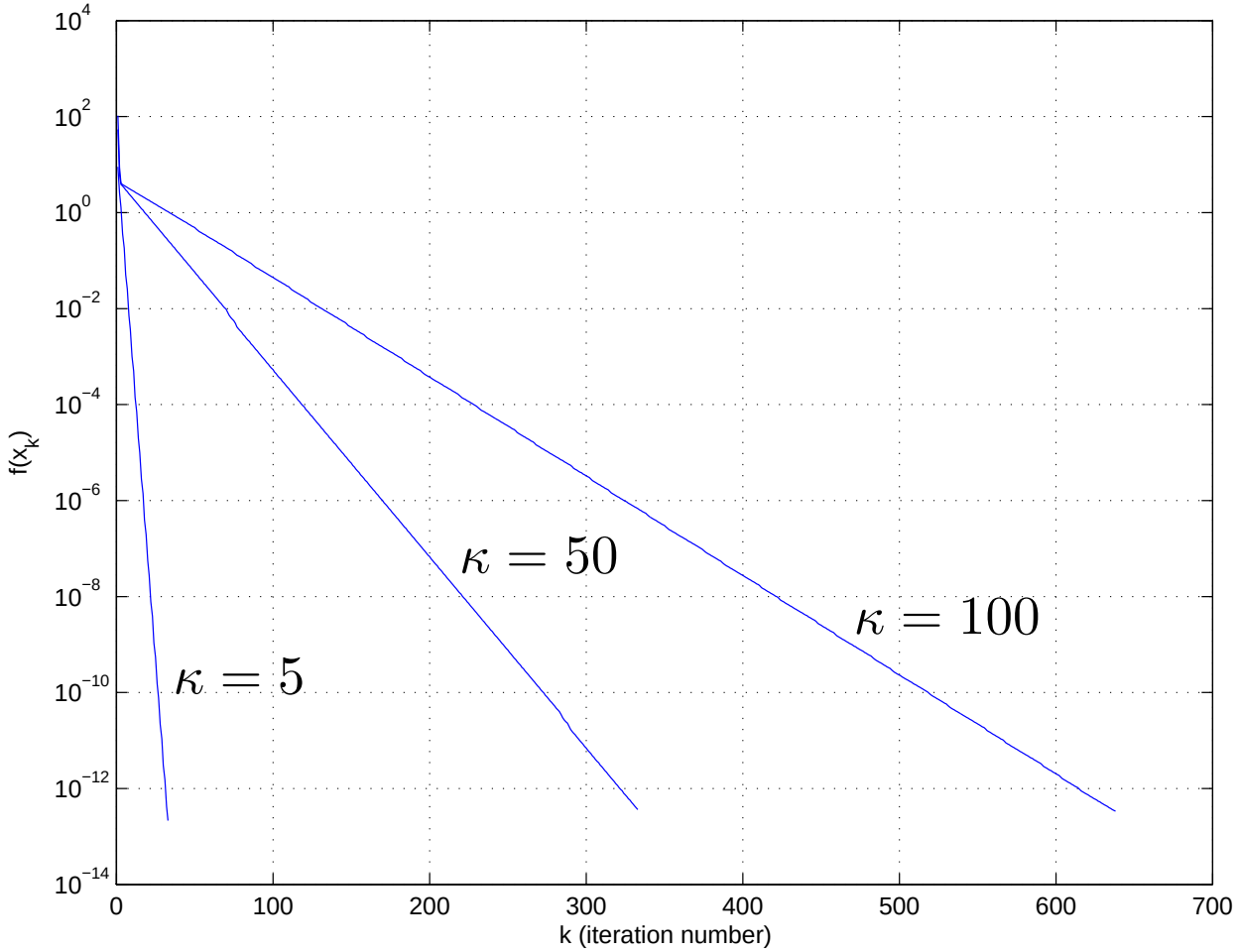


$\kappa = 50$ (number of iterations: 332)



$\kappa = 100$ (number of iterations: 637)

Sequence $\{f(x_k)\}_{k\geq 0}$



Definition (rates of convergence) Let $x_k \to x^\star$ $(x_k, x^\star \in \mathbb{R}^n)$ with $x_k \neq x^\star$. We say

- $x_k \to x^\star$ linearly if

$$\exists_{0<r<1} \ : \ \frac{\|x_{k+1} - x^\star\|}{\|x_k - x^\star\|} \leq r \text{ for all } k \text{ sufficiently large}$$

- $x_k \to x^\star$ superlinearly if

$$\frac{\|x_{k+1} - x^\star\|}{\|x_k - x^\star\|} \to 0$$

- $x_k \to x^\star$ quadratically if

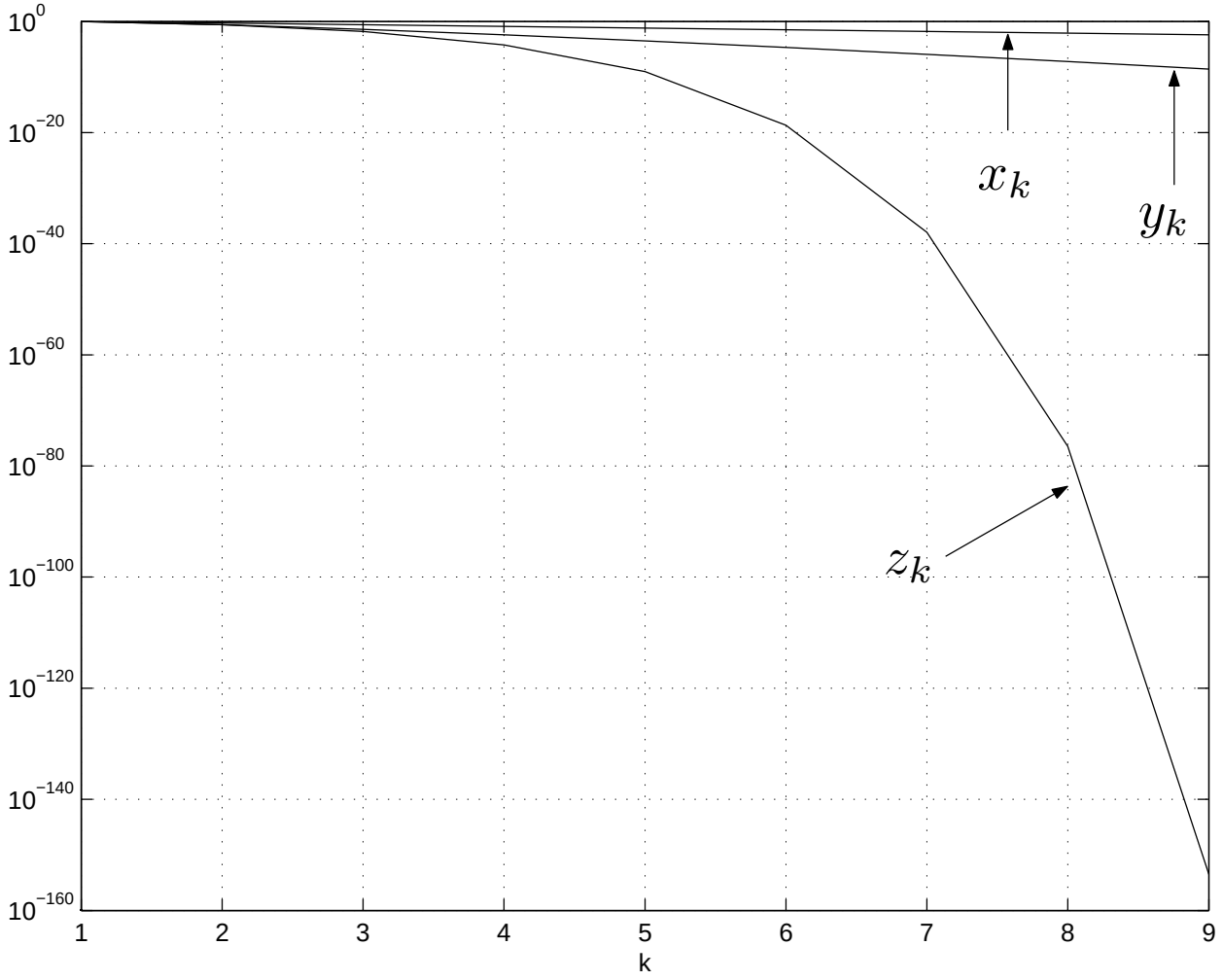
$$\exists_{r>0} \ : \ \frac{\|x_{k+1} - x^\star\|}{\|x_k - x^\star\|^2} \leq r \text{ for all } k \text{ sufficiently large}$$

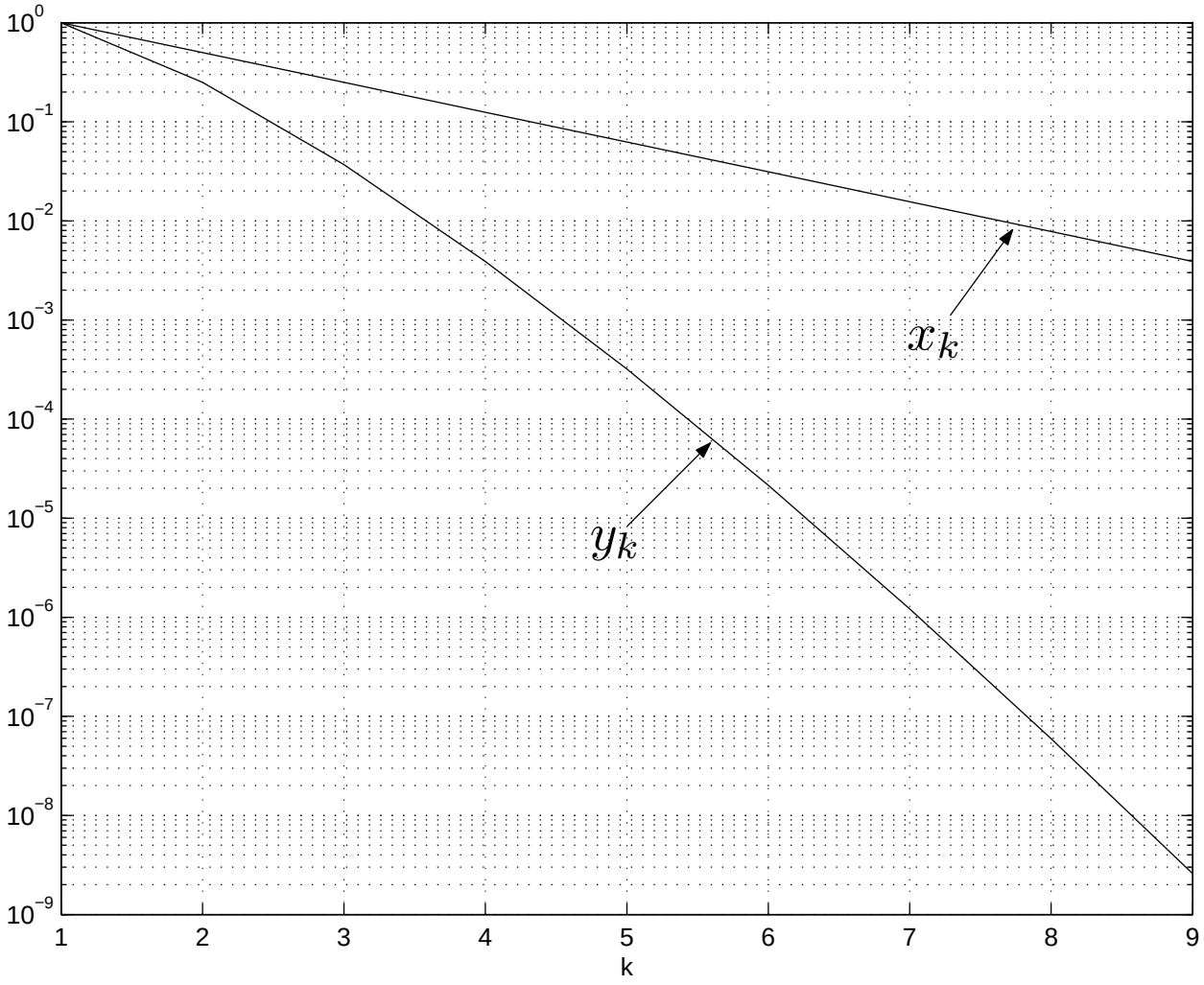
Examples:

$$
\begin{aligned}
x_k &= \left(\frac{1}{2}\right)^{k-1} \to 0 \text{ linearly} \\
y_k &= \frac{1}{k^k} \to 0 \text{ superlinearly} \\
z_k &= 4\left(\frac{1}{2}\right)^{2^k} \to 0 \text{ quadratically}
\end{aligned}
$$

First few terms of the sequences...

k	x_k	y_k	z_k
1	1	1	1
2	$5.0 \cdot 10^{-1}$	$2.5 \cdot 10^{-1}$	$2.5 \cdot 10^{-1}$
3	$2.5 \cdot 10^{-1}$	$3.7 \cdot 10^{-2}$	$1.6 \cdot 10^{-2}$
4	$1.3 \cdot 10^{-1}$	$3.9 \cdot 10^{-3}$	$6.1 \cdot 10^{-5}$
5	$6.3 \cdot 10^{-2}$	$3.2 \cdot 10^{-4}$	$9.3 \cdot 10^{-10}$
6	$3.1 \cdot 10^{-2}$	$2.1 \cdot 10^{-5}$	$2.2 \cdot 10^{-19}$
7	$1.6 \cdot 10^{-2}$	$1.2 \cdot 10^{-6}$	$1.2 \cdot 10^{-38}$
8	$7.8 \cdot 10^{-3}$	$6.0 \cdot 10^{-8}$	$3.5 \cdot 10^{-77}$
9	$3.9 \cdot 10^{-3}$	$2.6 \cdot 10^{-9}$	$3.0 \cdot 10^{-154}$


10^{0}
10^{-20}
10^{-40}
10^{-60}
10^{-80}
10^{-100}
10^{-120}
10^{-140}
10^{-160}
1
2
3
4
5
6
7
8
9
k
x_k
y_k
z_k


10^{0}
10^{-1}
10^{-2}
10^{-3}
10^{-4}
10^{-5}
10^{-6}
10^{-7}
10^{-8}
10^{-9}
1
2
3
4
5
6
7
8
9
k
x_k
y_k

Under mild assumptions, the steepest descent method $d_k = -\nabla f(x_k)$ yields linear convergence

Newton algorithm (first motivation):

- x_k is the current iterate
- the 2nd order model of f centered at x_k is

$$\widehat{f}_k(x) = f(x_k) + \nabla f(x_k)^\top (x - x_k) + \frac{1}{2}(x - x_k)^\top \nabla^2 f(x_k)(x - x_k)$$

- minimize 2nd order model to obtain the next iterate x_{k+1}:

$$x_{k+1} = \quad \arg\min_x \quad \widehat{f}_k(x) \quad = x_k + d_k \quad \text{where} \quad d_k = -\left[\nabla^2 f(x_k)\right]^{-1} \nabla f(x_k)$$

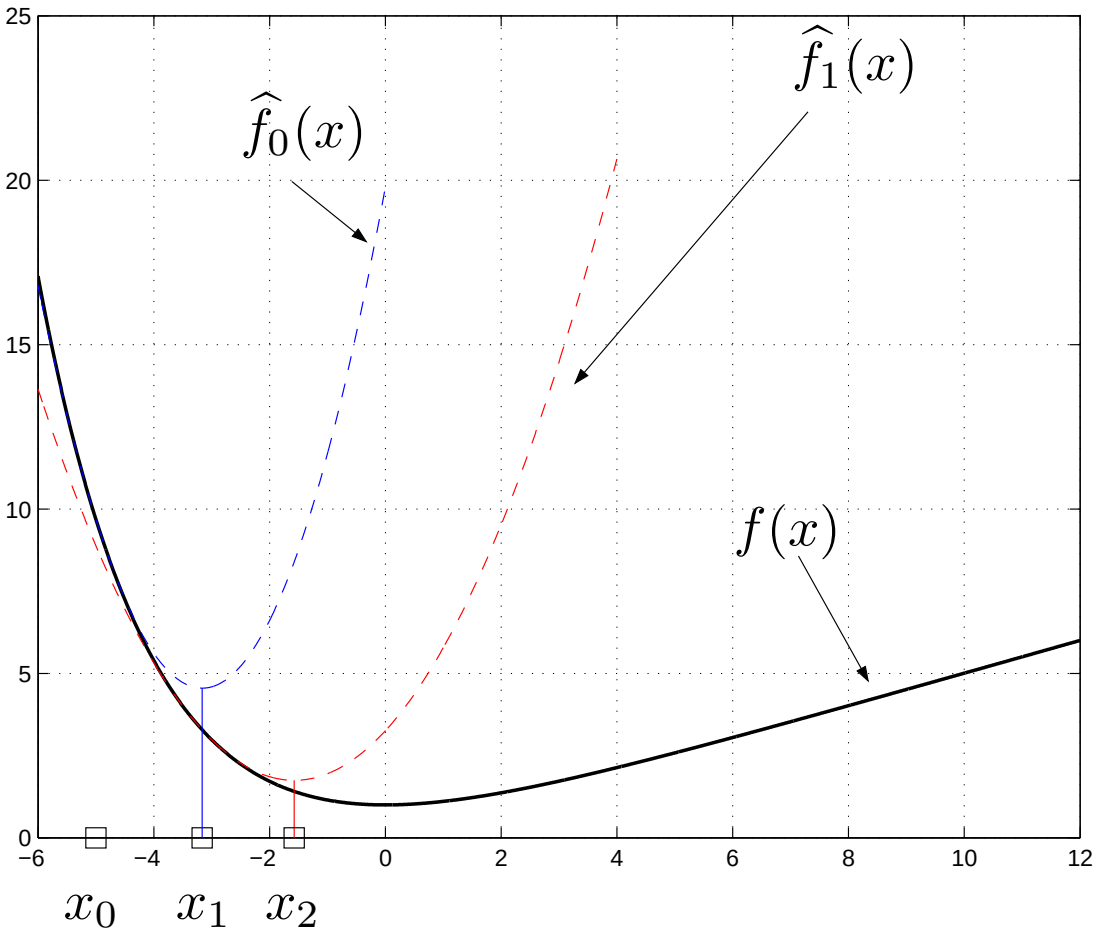
Note: it is assumed that $\nabla^2 f(x_k)$ is nonsingular

The algorithm

$$x_{k+1} = x_k + d_k \quad \text{with} \quad d_k = -\left[\nabla^2 f(x_k)\right]^{-1} \nabla f(x_k)$$

is called the pure Newton algorithm

Example: $f : \mathbb{R} \to \mathbb{R}$, $f(x) = e^{-\frac{1}{2}x} + \frac{1}{2}x$



Newton algorithm (second motivation):

- Newton method for solving a system of nonlinear equations

$$\begin{cases} F_1(x_1, \ldots, x_n) = 0 \\ F_2(x_1, \ldots, x_n) = 0 \\ \qquad\qquad \vdots \\ F_n(x_1, \ldots, x_n) = 0 \end{cases} \quad \Leftrightarrow \quad F(x) = 0$$

  is

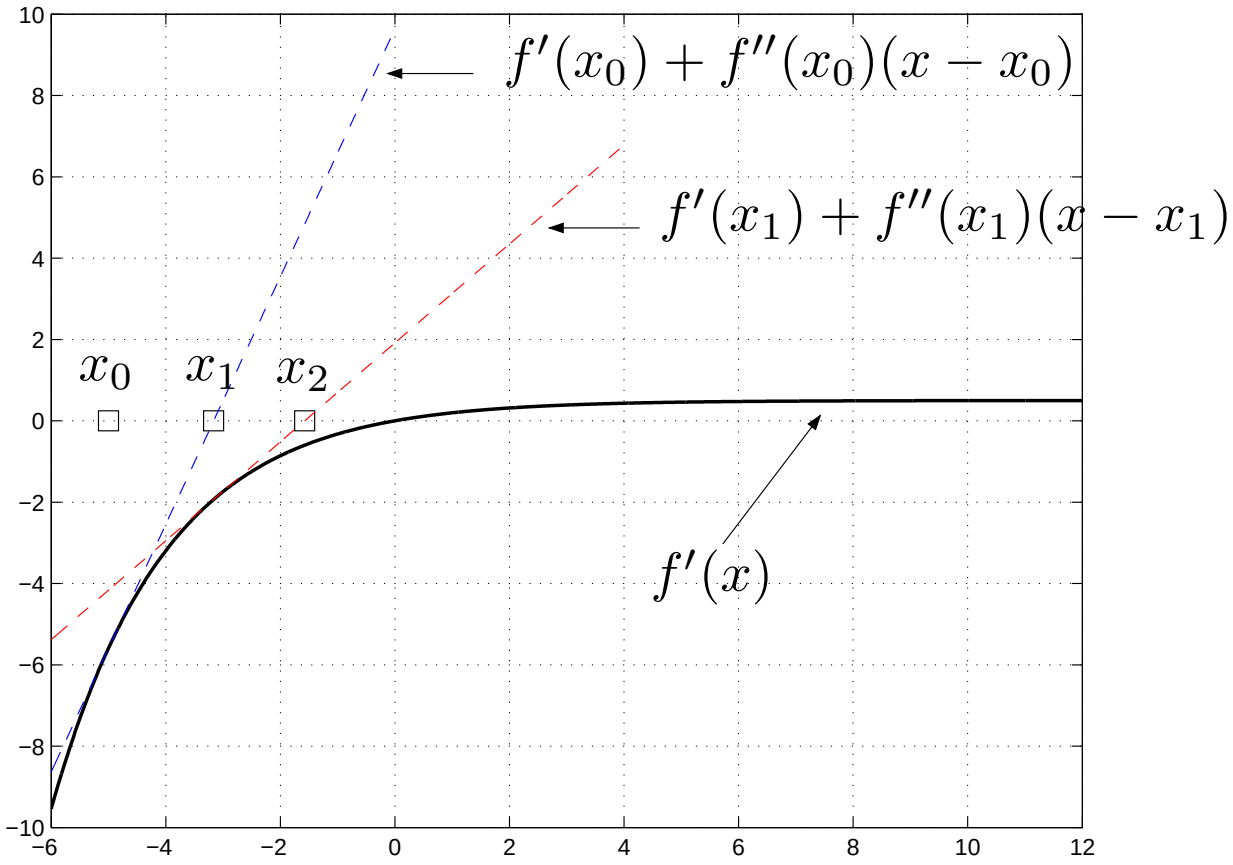
$$x_{k+1} = x_k - DF(x_k)^{-1} F(x_k).$$

  Note: the inverse $DF(x_k)^{-1}$ is supposed to exist

- the pure Newton algorithm corresponds to the special case $\nabla f(x) = 0$:

$$x_{k+1} = x_k - \left[\nabla^2 f(x_k)\right]^{-1} \nabla f(x_k)$$

Example: $f : \mathbb{R} \to \mathbb{R}$, $f(x) = e^{-\frac{1}{2}x} + \frac{1}{2}x$



- The pure Newton algorithm $x_{k+1} = x_k + d_k$ is convergent only near a solution
- The Newton direction

$$d_k = -\left[\nabla^2 f(x_k)\right]^{-1} \nabla f(x_k)$$

  is a descent direction for f at x_k:

$$\langle d_k, \nabla f(x_k) \rangle = -\nabla f(x_k)^\top \left[\nabla^2 f(x_k)\right]^{-1} \nabla f(x_k) < 0$$

- Use a line-search to make it globally convergent: $x_{k+1} = x_k + t_k d_k$

- The quantity

$$\lambda(x) := \sqrt{\nabla f(x)^\top \left[\nabla^2 f(x)\right]^{-1} \nabla f(x)} = \sqrt{-g_k^\top d_k}$$

  is called the Newton decrement at x

- Property of the Newton decrement:

$$\frac{1}{2}\lambda^2(x) = f(x) - \min\left\{ f(x) + \nabla f(x)^\top (y - x) + \frac{1}{2}(y - x)^\top \nabla^2 f(x)(y - x) \ : \ y \right\}$$

- Affine invariant stopping criterion:

$$\lambda^2(x) < \epsilon$$

- The Newton algorithm solves a sequence of unconstrained quadratic problems

1. **initialization** ▶ choose $x_0 \in \mathbf{dom}\, f$ and tolerance $\epsilon > 0$

2. ▶ choose $0 < \alpha < 0.5$ and $0 < \beta < 1$ (Armijo)

3. ▶ set $k = 0$

4. **loop** ▶ compute $g_k = \nabla f(x_k)$

5. ▶ compute $H_k = \nabla^2 f(x_k)$

6. ▶ solve the linear system $H_k d_k = -g_k$

7. ▶ if $g_k^\top H_k^{-1} g_k = -g_k^\top d_k < \epsilon$ stop

8. ▶ $t = 1$

9. ▶ while $f(x_k + td_k) > f(x_k) + t\alpha g_k^\top d_k$

$t \leftarrow \beta t$

10. ▶ $x_{k+1} = x_k + td_k$

11. ▶ $k \leftarrow k + 1$ and return to loop

Theorem (Convergence of Newton method) Let $x_0 \in \text{dom}\, f$ and suppose there exists $m > 0$ such that

$$\nabla^2 f(x) \succeq mI \quad \text{for all } x \in S_{f(x_0)}\, f.$$

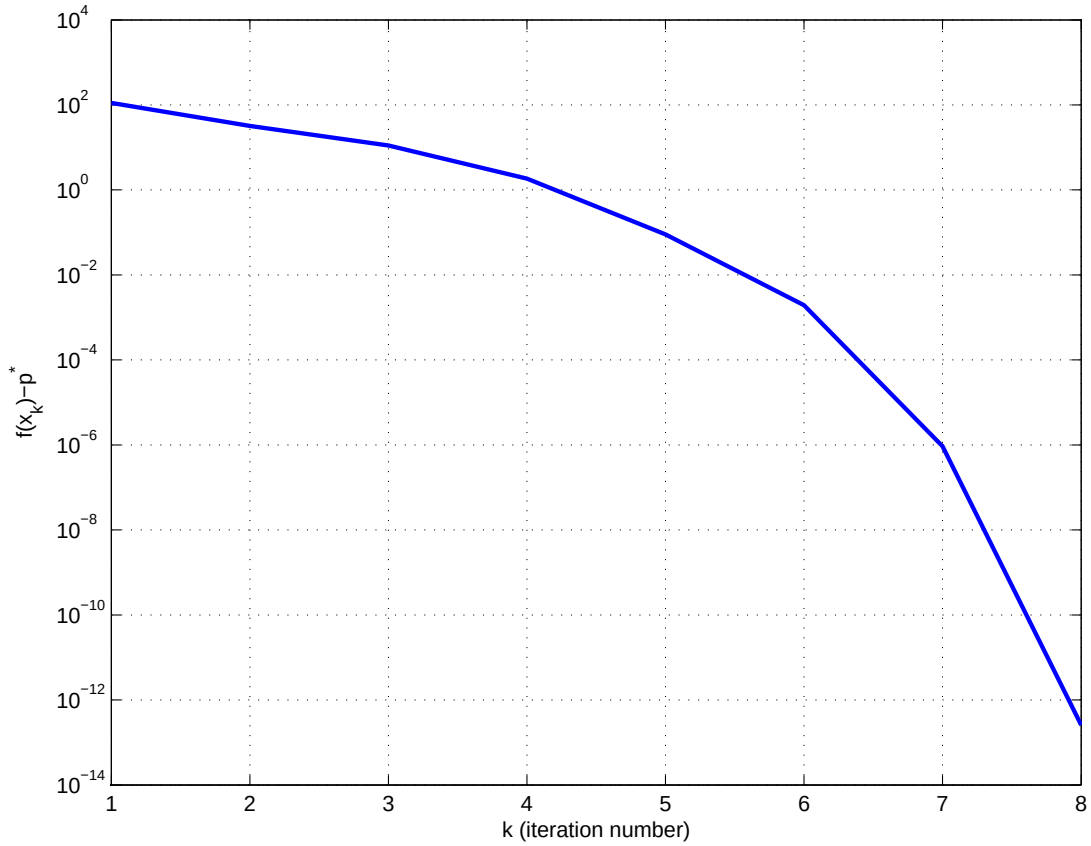
Also assume that $\nabla^2 f(x)$ is Lipschitz continuous on $S_{f(x_0)}\, f$. Then,

$$x_k \to x^\star \quad \text{quadratically.}$$

Example:

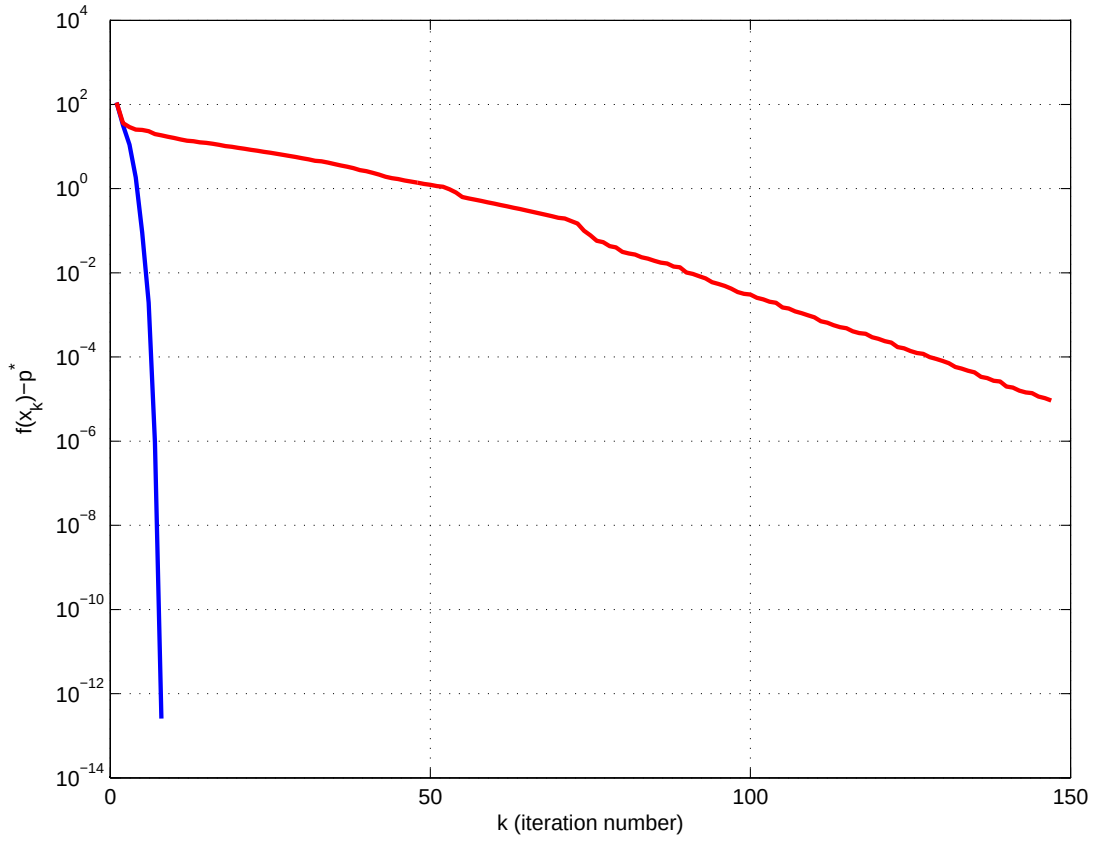
$$f(x) = c^\top x - \sum_{i=1}^{m} \log\left(b_i - a_i^\top x\right)$$

Newton method



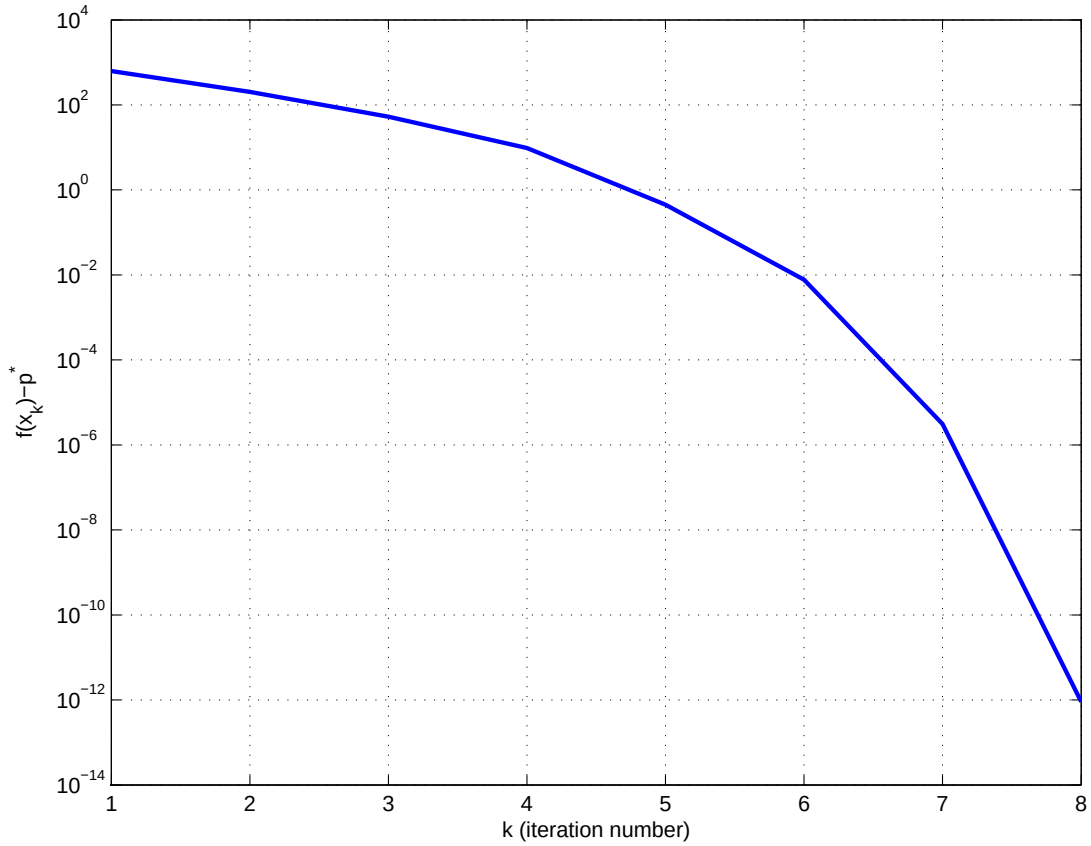
$$n = 8,\ m = 50,\ \kappa\left(\nabla^2 f(x^\star)\right) \simeq 42$$

Newton and gradient descent methods



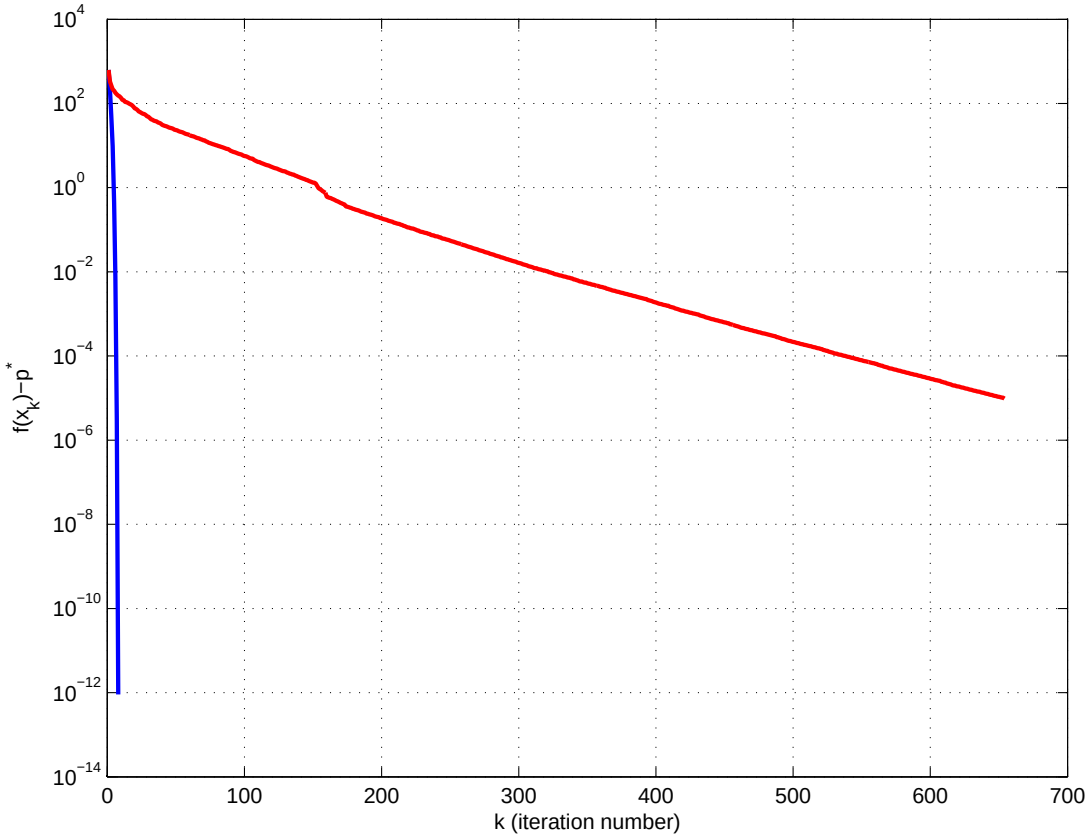
$$n = 8,\ m = 50,\ \kappa\left(\nabla^2 f(x^\star)\right) \simeq 42$$

Newton method



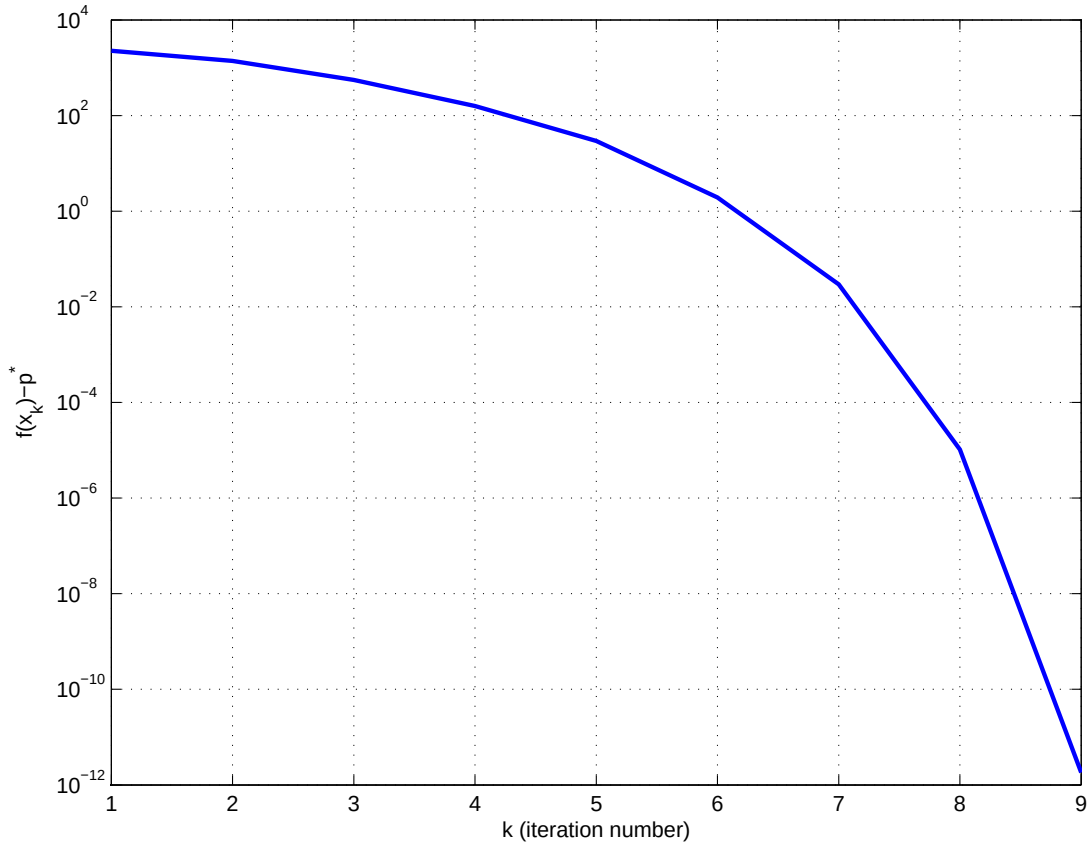
$$n = 50,\ m = 250,\ \kappa\left(\nabla^2 f(x^\star)\right) \simeq 253$$

Newton and gradient descent methods



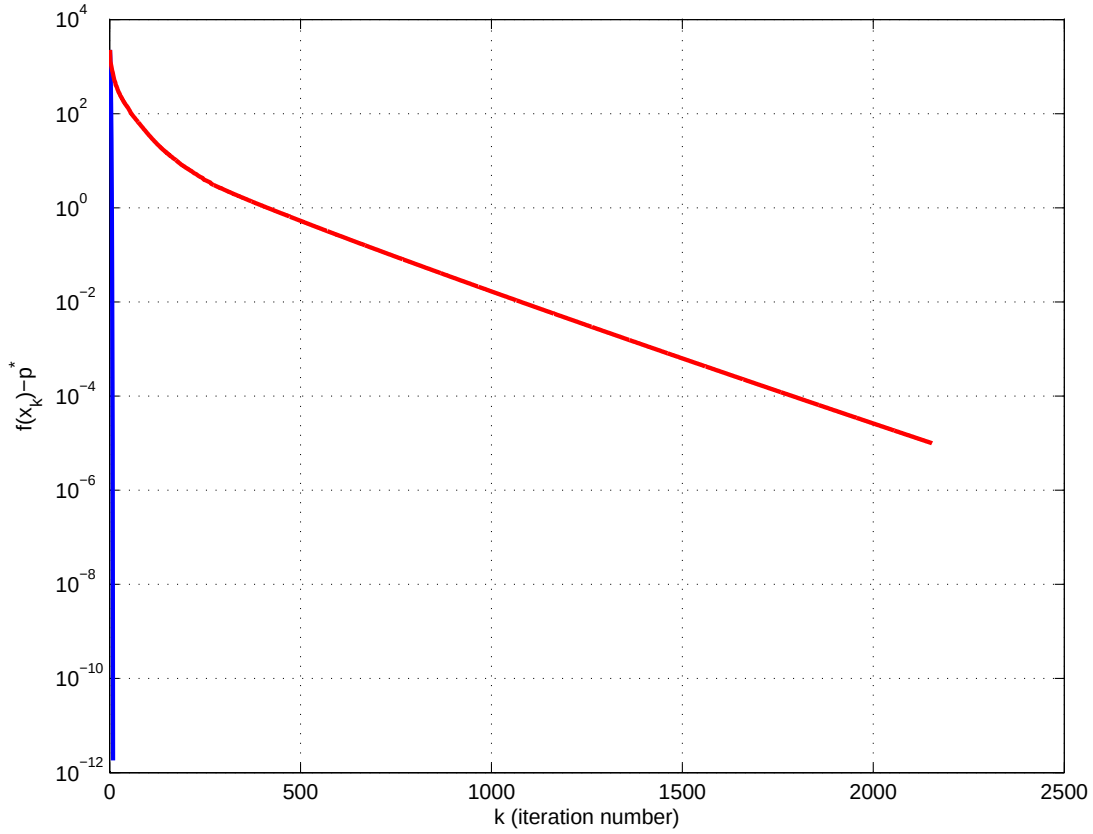
$$n = 50,\ m = 250,\ \kappa\left(\nabla^2 f(x^\star)\right) \simeq 253$$

Newton method



$$n = 200,\ m = 800,\ \kappa\left(\nabla^2 f(x^\star)\right) \simeq 777$$

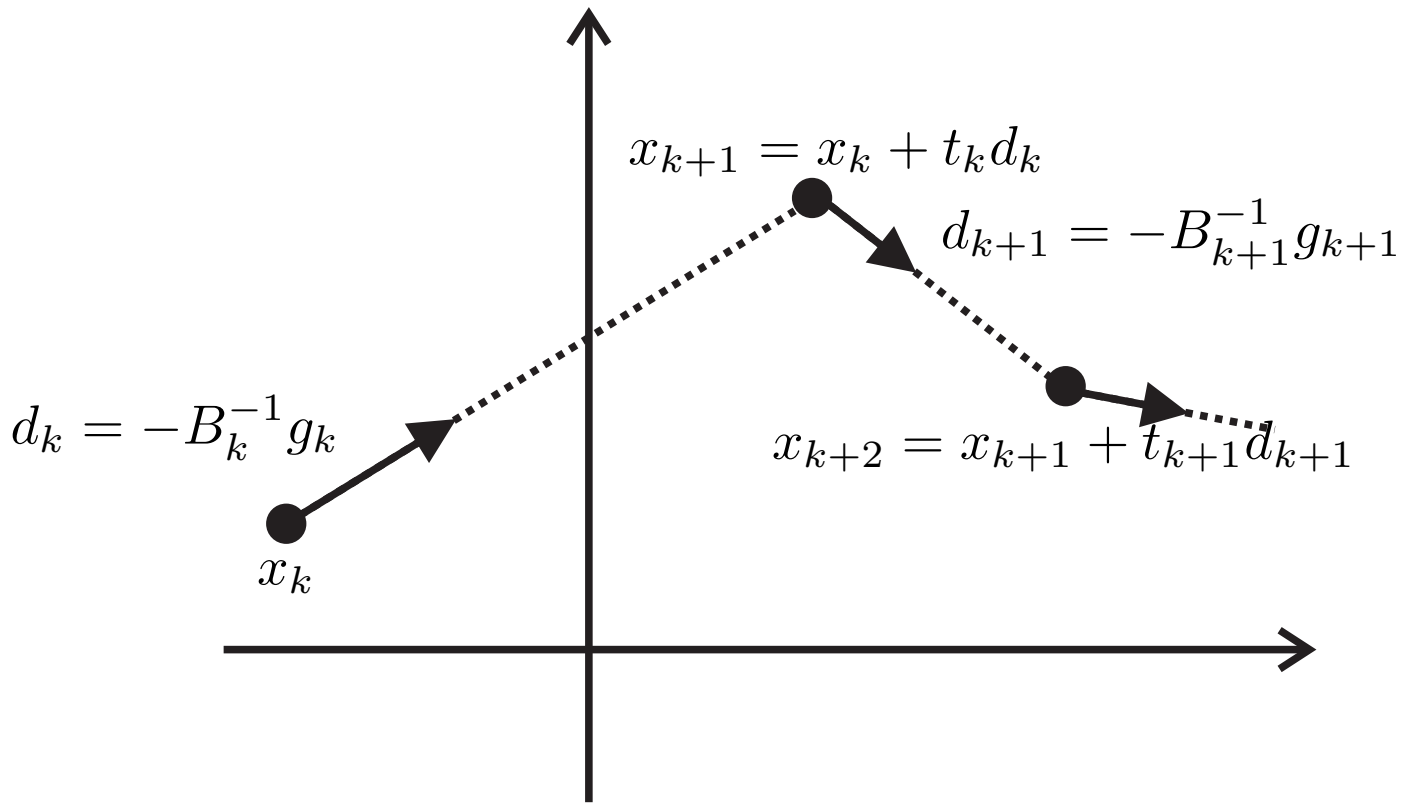
Newton and gradient descent methods



$$n = 200,\ m = 800,\ \kappa\left(\nabla^2 f(x^\star)\right) \simeq 777$$

Quasi-Newton BFGS algorithm

- BFGS = Broyden, Fletcher, Goldfarb and Shanno
- Newton method needs $H_k = \nabla^2 f(x_k)$ at each iteration
- idea: instead, use an approximation B_k



- **Initialization:** $B_0 = I_n$ or a better guess
- **Update:** how to update $B_k \to B_{k+1}$?

Quadratic model $m(x)$ for $f(x)$ centered at x_{k+1}:

$$m(x) = f(x_{k+1}) + \nabla f(x_{k+1})^\top (x - x_{k+1}) + \frac{1}{2}(x - x_{k+1})^\top B_{k+1}(x - x_{k+1})$$

"Nice" properties that B_{k+1} should possess:

- $\nabla m(x_k) = g_k$ which leads to the secant equation

$$B_{k+1} \underbrace{(x_{k+1} - x_k)}_{s_k} = \underbrace{(g_{k+1} - g_k)}_{y_k}$$

- $B_{k+1} \succ 0$ to ensure that $d_{k+1} = -B_{k+1}^{-1} g_k$ is a descent direction
- $B_{k+1} \simeq B_k$ (f is C^2 implies $\nabla^2 f(x)$ changes continuously with x)

First two properties might conflict: for given s_k, y_k there exist no

$$B_{k+1} \succ 0 \text{ such that } B_{k+1} s_k = y_k$$

Example: $s_k = 1$, $y_k = -1$ implies $B_{k+1} = y_k / s_k = -1$

Fact. Let $s, y \in \mathbb{R}^n$ be given with $s \neq 0$. Then,

$$\exists B \succ 0 \, : \, y = Bs \quad \Leftrightarrow \quad y^\top s > 0.$$

Proof ($\Leftarrow$): $B = I_n - \dfrac{ss^\top}{s^\top s} + \dfrac{yy^\top}{y^\top s}$ works.

Note: if $\nabla^2 f(x) \succ 0$ for all x then $y_k^\top s_k > 0$

- Translate desired properties into an optimization problem:

$$\begin{aligned} \text{minimize} \quad & \phi(B) := \text{tr}\left(B_k^{-1/2} B B_k^{-1/2}\right) - \log\det\left(B_k^{-1/2} B B_k^{-1/2}\right) - n \\ \text{subject to} \quad & Bs_k = y_k \\ & B \succ 0 \end{aligned}$$

- $\phi(B)$ is a "distance" measure between B and B_k:

$$\phi(B) \geq 0 \text{ with equality if and only if } B = B_k$$

- Solution is

$$B_{k+1} = B_k - \frac{B_k s_k s_k^\top B_k}{s_k^\top B_k s_k} + \frac{y_k y_k^\top}{y_k^\top s_k}.$$

  This rank-2 update is known as the **BFGS update**

We can propagate directly the inverse $W_{k+1} = B_{k+1}^{-1}$:

$$W_{k+1} = \left(I - \rho_k s_k y_k^\top\right) W_k \left(I - \rho_k y_k s_k^\top\right) + \rho_k s_k s_k^\top \quad \text{where} \quad \rho_k = \frac{1}{y_k^\top s_k}$$

1. **initialization** ▶ choose $x_0 \in \mathbf{dom}\, f$, $W_0 \succ 0$ and $\epsilon > 0$

2. ▶ choose $0 < \alpha < 0.5$ and $0 < \beta < 1$ (Armijo)

3. ▶ set $k = 0$

4. **loop** ▶ compute $g_k = \nabla f(x_k)$

5. ▶ if $\|g_k\| < \epsilon$ stop

6. ▶ compute search direction $d_k = -W_k g_k$

7. ▶ $t = 1$

8. ▶ while $f(x_k + td_k) > f(x_k) + t\alpha g_k^\top d_k$

$$t \leftarrow \beta t$$

9. ▶ $x_{k+1} = x_k + td_k$

10. ▶ $W_{k+1} = \left(I - \rho_k s_k y_k^\top\right) W_k \left(I - \rho_k y_k s_k^\top\right) +$

$$\rho_k s_k s_k^\top \text{ with } \rho_k = 1/y_k^\top s_k$$

11. ▶ $k \leftarrow k + 1$ and return to loop

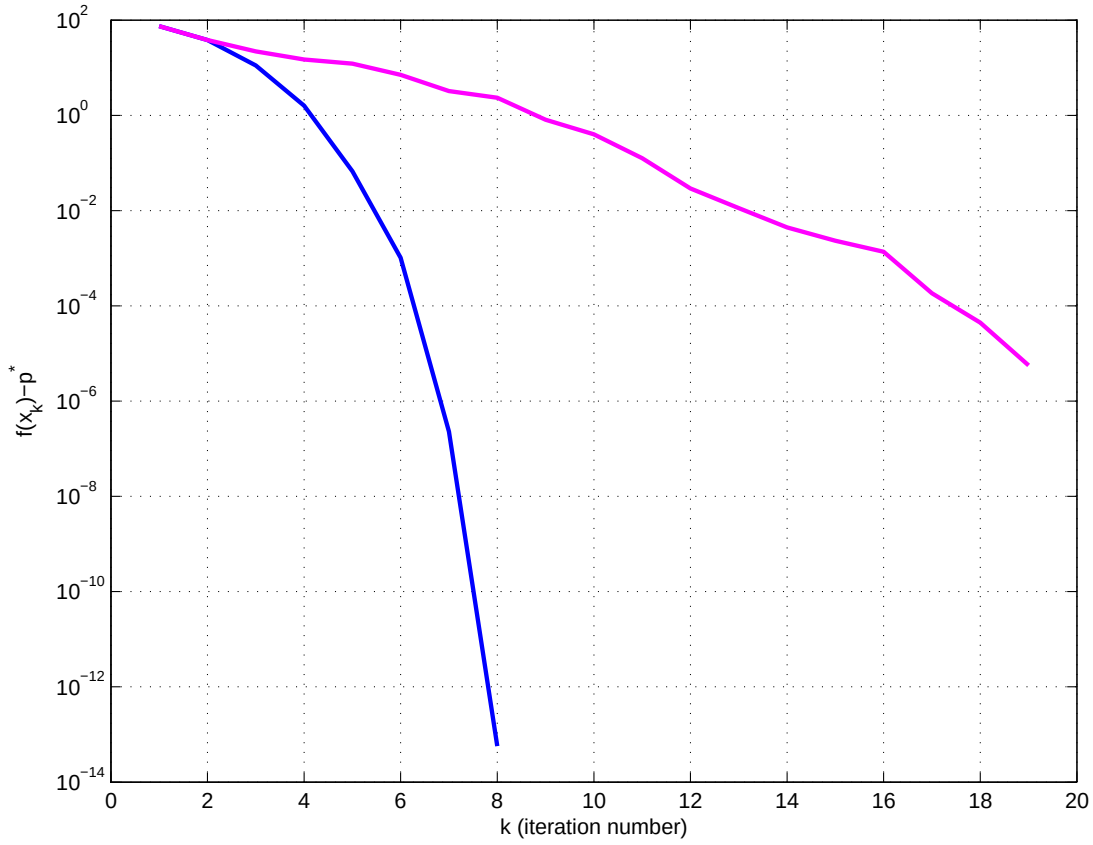
Theorem (Convergence of Quasi-Newton BFGS method) Let $x_0 \in \mathrm{dom}\, f$ and suppose there exists $m > 0$ such that

$$\nabla^2 f(x) \succeq mI \quad \text{for all } x \in S_{f(x_0)}\, f.$$

Also assume that $\nabla^2 f(x)$ is Lipschitz continuous on $S_{f(x_0)}\, f$. Then,

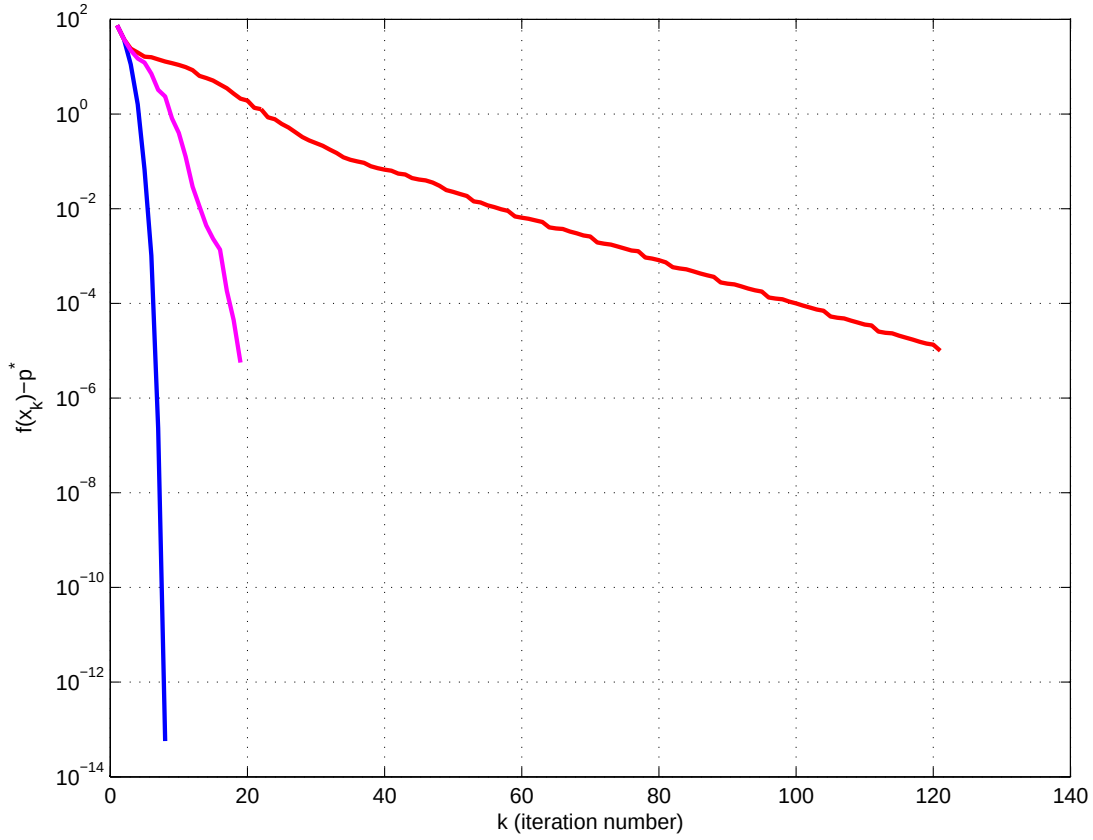
$$x_k \to x^\star \quad \text{superlinearly.}$$

Newton and Quasi-Newton BFGS methods



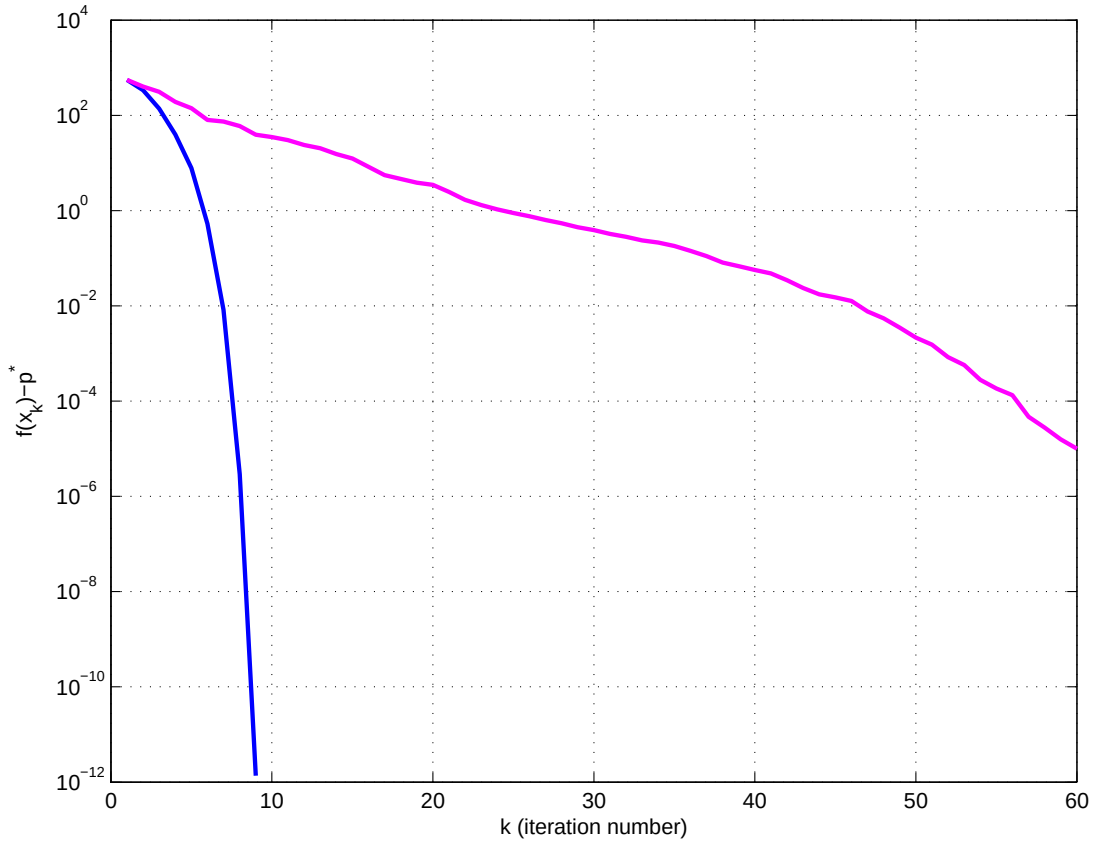
$$n = 8,\ m = 50,\ \kappa\left(\nabla^2 f(x^\star)\right) \simeq 786$$

Newton, Quasi-Newton BFGS and gradient descent methods



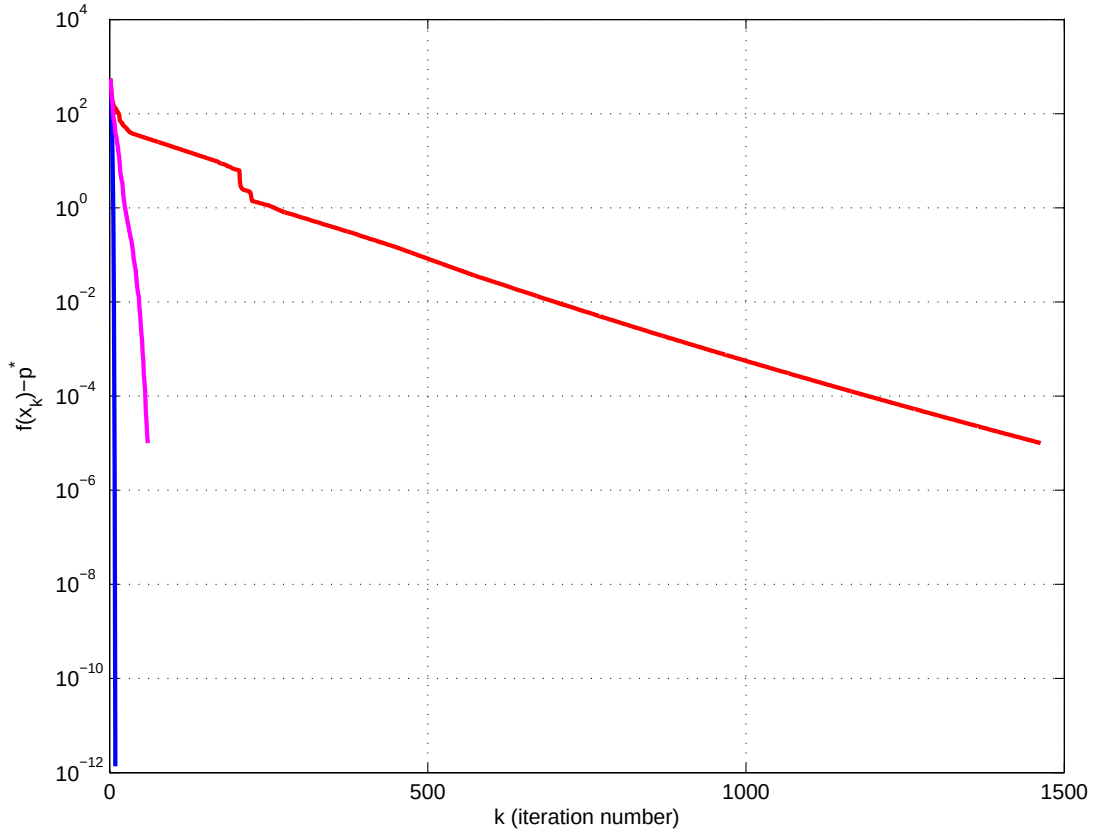
$$n = 8,\ m = 50,\ \kappa\left(\nabla^2 f(x^\star)\right) \simeq 786$$

Newton and Quasi-Newton BFGS methods



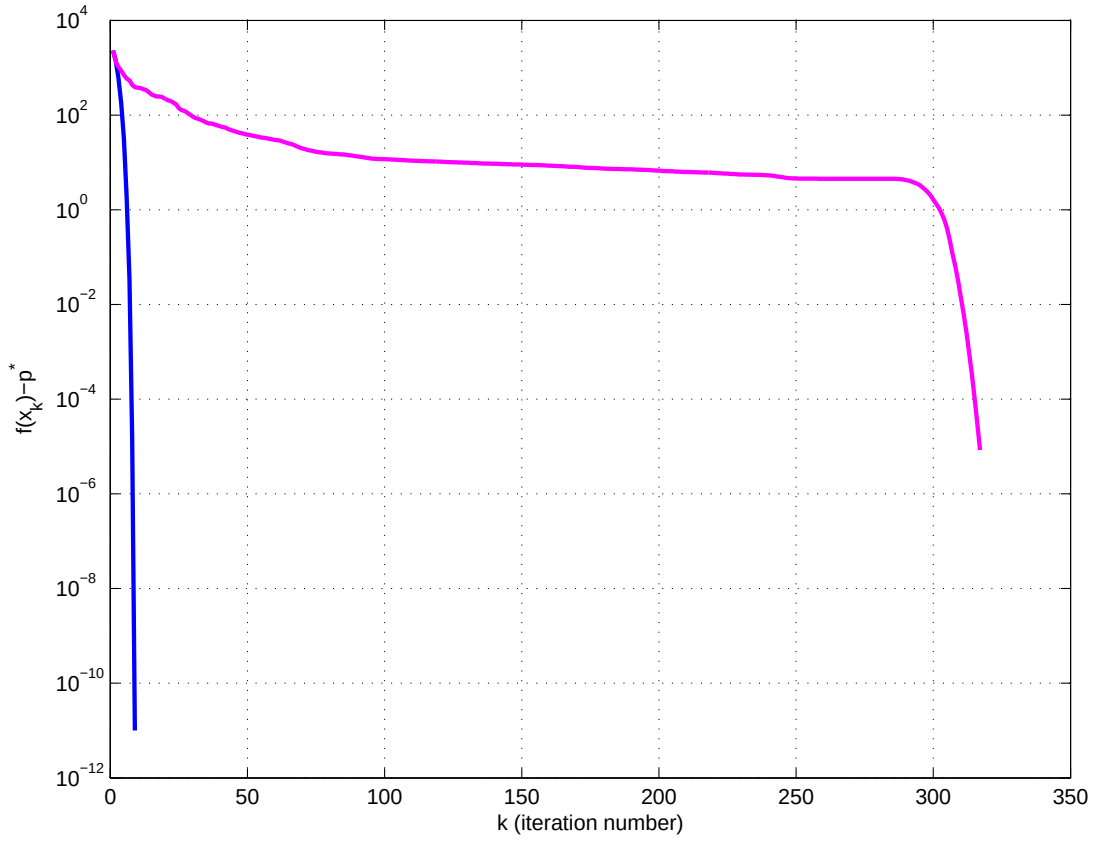
$$n = 50,\ m = 250,\ \kappa\left(\nabla^2 f(x^\star)\right) \simeq 851$$

Newton, Quasi-Newton BFGS and gradient descent methods



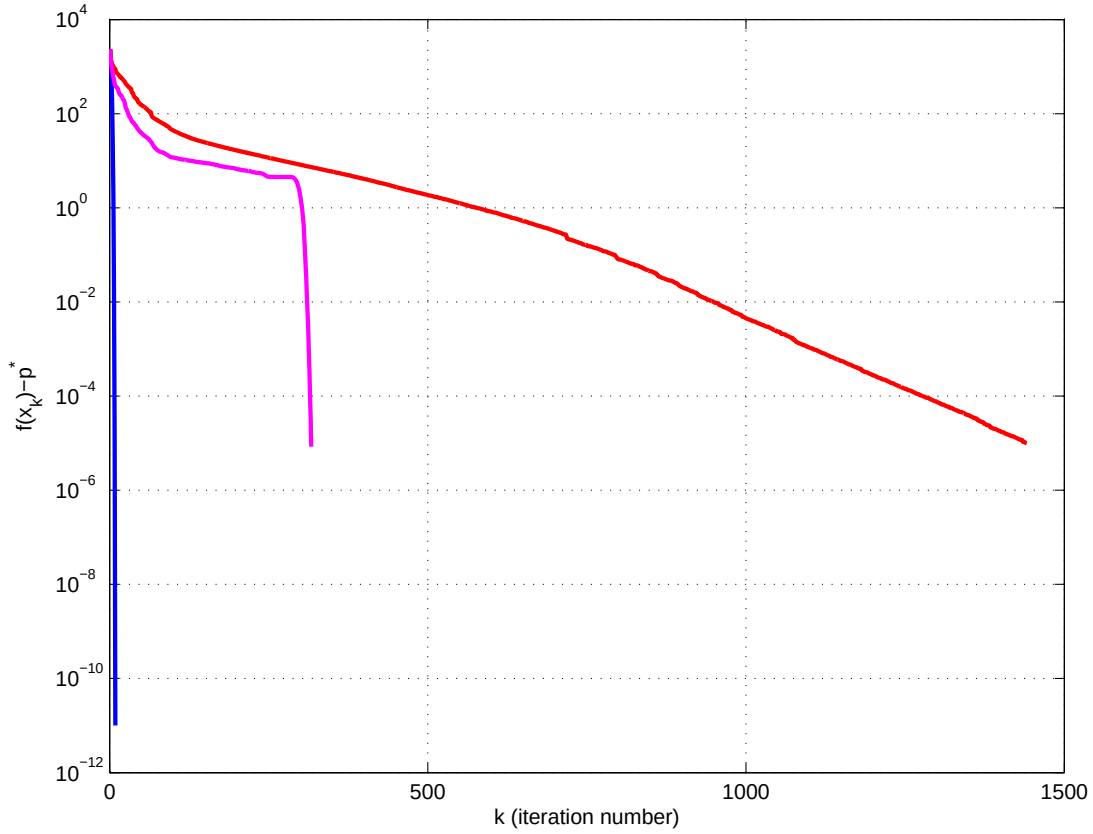
$$n = 50,\ m = 250,\ \kappa\left(\nabla^2 f(x^\star)\right) \simeq 851$$

Newton and Quasi-Newton BFGS methods



$$n = 200,\ m = 800,\ \kappa\left(\nabla^2 f(x^\star)\right) \simeq 873$$

Newton, Quasi-Newton BFGS and gradient descent methods



$$n = 200,\ m = 800,\ \kappa\left(\nabla^2 f(x^\star)\right) \simeq 873$$

Constrained minimization (equality constraints)

Optimization problem:

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & Ax = b \end{array}$$

Assumptions:

- $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ is convex
- dom $f = \{x : f(x) < +\infty\}$ is open
- f is closed
- f is twice continuously differentiable
- problem is solvable: $\exists\, x^\star : f(x^\star) = p^\star := \inf\{f(x) : Ax = b\}$
- matrix A has full row rank (otherwise, drop redundant eqs)

Two main approaches:

- eliminate equality constraints and switch to unconstrained problem
- maintain equality constraints

Eliminate equality constraints:

- parameterize constraint set as

$$\{x \,:\, Ax = b\} = \{x_0 + By \,:\, y\}$$

  where $Ax_0 = b$ and B contains a basis for Ker A

- solve equivalent unconstrained optimization problem

$$y^\star = \quad \arg\min \quad f(x_0 + By)$$

- recover solution

$$x^\star = x_0 + By^\star$$

- sparsity structure in A might be lost

Maintain equality constraints (first motivation):

- x_k is the current iterate
- the 2nd order model of f centered at x_k is

$$\widehat{f}_k(x) = f(x_k) + \nabla f(x_k)^\top (x - x_k) + \frac{1}{2}(x - x_k)^\top \nabla^2 f(x_k)(x - x_k)$$

- minimize 2nd order model to obtain the next iterate x_{k+1}:

$$\begin{aligned} x_{k+1} = \quad & \arg\min \quad \widehat{f}_k(x) \quad = x_k + d_k \\ & \text{subject to} \quad Ax = b \end{aligned}$$

where

$$\begin{bmatrix} \nabla^2 f(x_k) & A^\top \\ A & 0 \end{bmatrix} \begin{bmatrix} d_k \\ \lambda_k \end{bmatrix} = - \begin{bmatrix} \nabla f(x_k) \\ 0 \end{bmatrix}$$

Maintain equality constraints (second motivation):

- KKT system is

$$\begin{cases} \nabla f(x) + A^\top \lambda = 0 \\ Ax - b = 0 \end{cases} \quad \Leftrightarrow \quad r(x, \lambda) = 0$$

- linearize around the current iterate x_k:

$$r(x_k + d_k, \lambda_k) \simeq \big(\nabla f(x_k) + \nabla^2 f(x_k) d_k + A^\top \lambda_k, A(x_k + d_k) - b\big)$$

- solve for (d_k, λ_k):

$$\begin{bmatrix} \nabla^2 f(x_k) & A^\top \\ A & 0 \end{bmatrix} \begin{bmatrix} d_k \\ \lambda_k \end{bmatrix} = - \begin{bmatrix} \nabla f(x_k) \\ 0 \end{bmatrix}$$

- $x_{k+1} = x_k + d_k$

1. **initialization** ▶ choose $x_0 \in \mathbf{dom}\, f$ $(Ax_0 = b)$ and $\epsilon > 0$

2. ▶ choose $0 < \alpha < 0.5$ and $0 < \beta < 1$ (Armijo)

3. ▶ set $k = 0$

4. **loop** ▶ compute $g_k = \nabla f(x_k)$

5. ▶ compute $H_k = \nabla^2 f(x_k)$

6. ▶ solve $\begin{bmatrix} H_k & A^\top \\ A & 0 \end{bmatrix} \begin{bmatrix} d_k \\ \lambda_k \end{bmatrix} = \begin{bmatrix} -g_k \\ 0 \end{bmatrix}$

7. ▶ if $-g_k^\top d_k < \epsilon$ stop

8. ▶ $t = 1$

9. ▶ while $f(x_k + td_k) > f(x_k) + t\alpha g_k^\top d_k$

   $t \leftarrow \beta t$

10. ▶ $x_{k+1} = x_k + td_k$

11. ▶ $k \leftarrow k + 1$ and return to loop

Constrained minimization (inequality constraints)

Optimization problem:

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & Ax = b \\ & g_i(x) \leq 0 \quad i = 1, \ldots, m \end{array}$$

Assumptions:

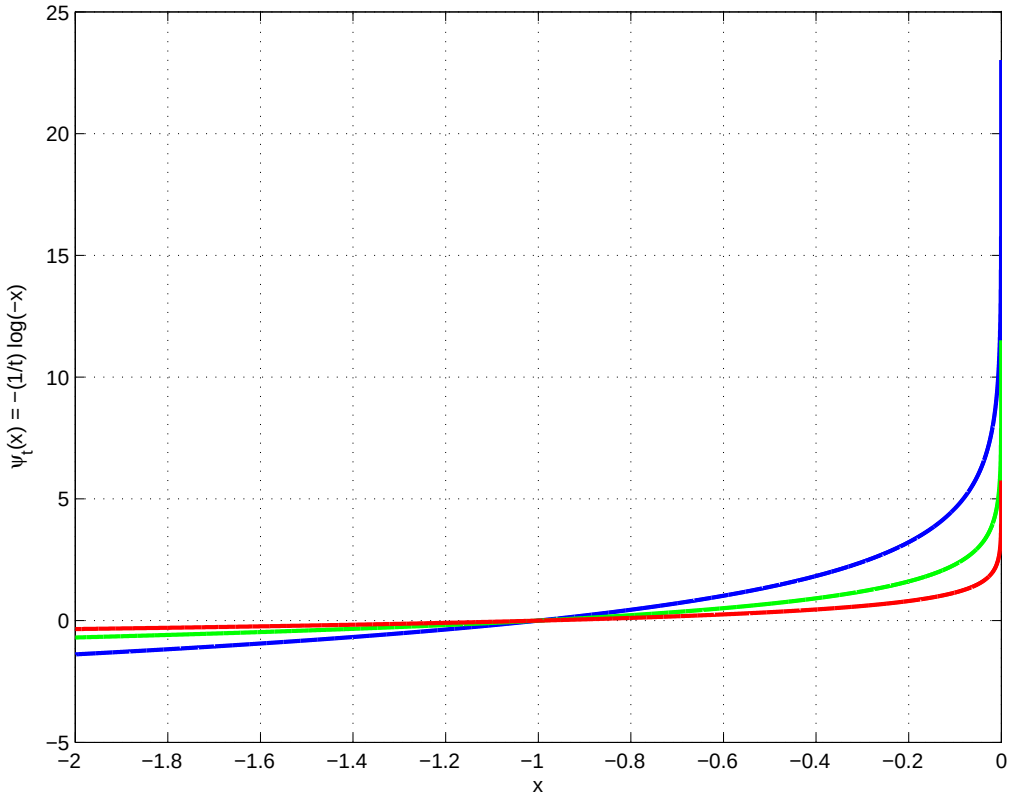
- $f, g_i : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ are convex
- dom f, dom g_i are open
- f, g_i are closed and twice continuously differentiable
- problem is solvable:

  $\exists\, x^\star : f(x^\star) = p^\star := \inf\{f(x) : Ax = b,\, g(x) = (g_1(x), \ldots, g_m(x)) \leq 0\}$

- matrix A has full row rank
- there is a Slater point: $\exists_{x \in \text{dom}\, f} : Ax = b,\, g(x) < 0$

Logarithmic barrier:

$$\psi_t : \mathbb{R} \to \mathbb{R} \cup \{+\infty\} \quad \psi_t(x) = -\frac{1}{t}\log(-x) \quad \text{dom}\, \psi_t = -\mathbb{R}_{++}$$



Plot of ψ_t for $t = 0.5$, $t = 1$ and $t = 2$

- Consider the approximation

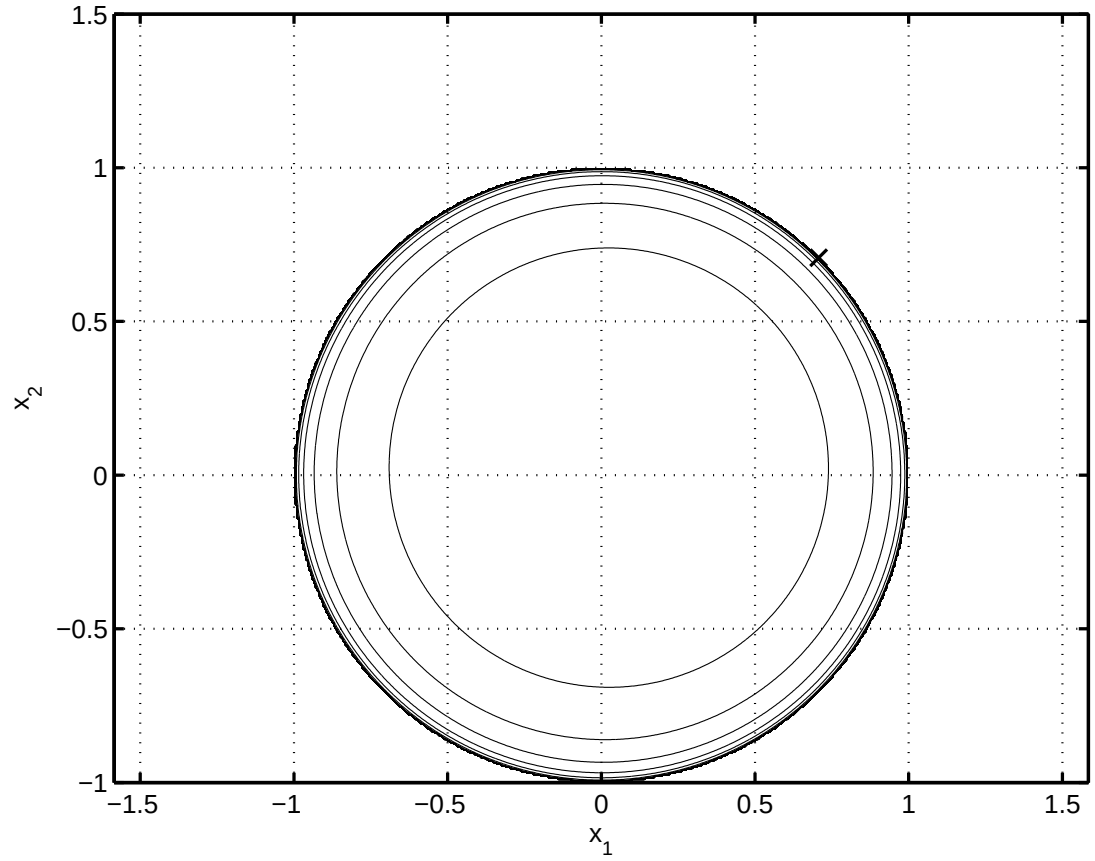
$$\begin{array}{ll} \text{minimize} & f(x) + \sum_{i=1}^{m} -(1/t)\log\left(-g_i(x)\right) \\ \text{subject to} & Ax = b \end{array}$$

- Example:

$$\begin{array}{ll} \text{minimize} & -x_1 - x_2 \\ \text{subject to} & x_1^2 + x_2^2 \leq 1 \end{array}$$

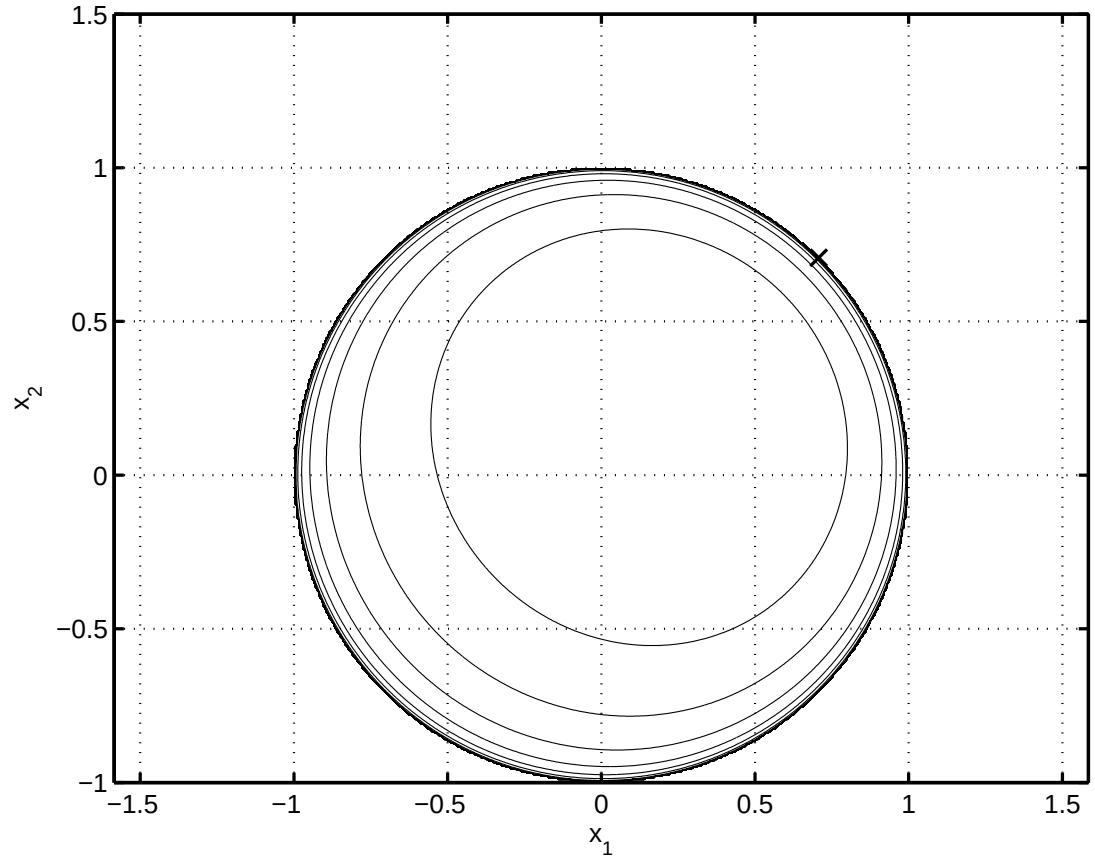
Solution is $(x_1^\star, x_2^\star) = (\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})$

Level sets for $\phi_t(x_1, x_2) = -x_1 - x_2 - (1/t)\log\left(1 - x_1^2 - x_2^2\right)$



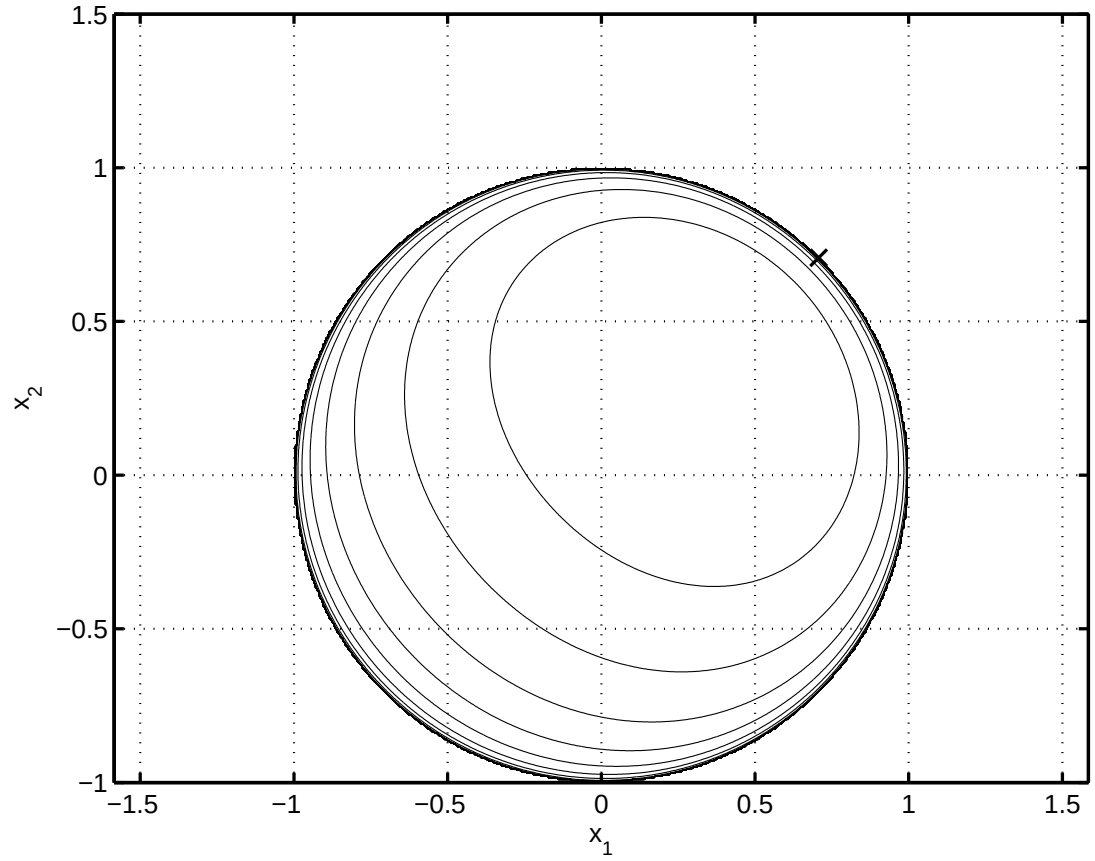
$t = 0.1$

Level sets for $\phi_t(x_1, x_2) = -x_1 - x_2 - (1/t)\log\left(1 - x_1^2 - x_2^2\right)$



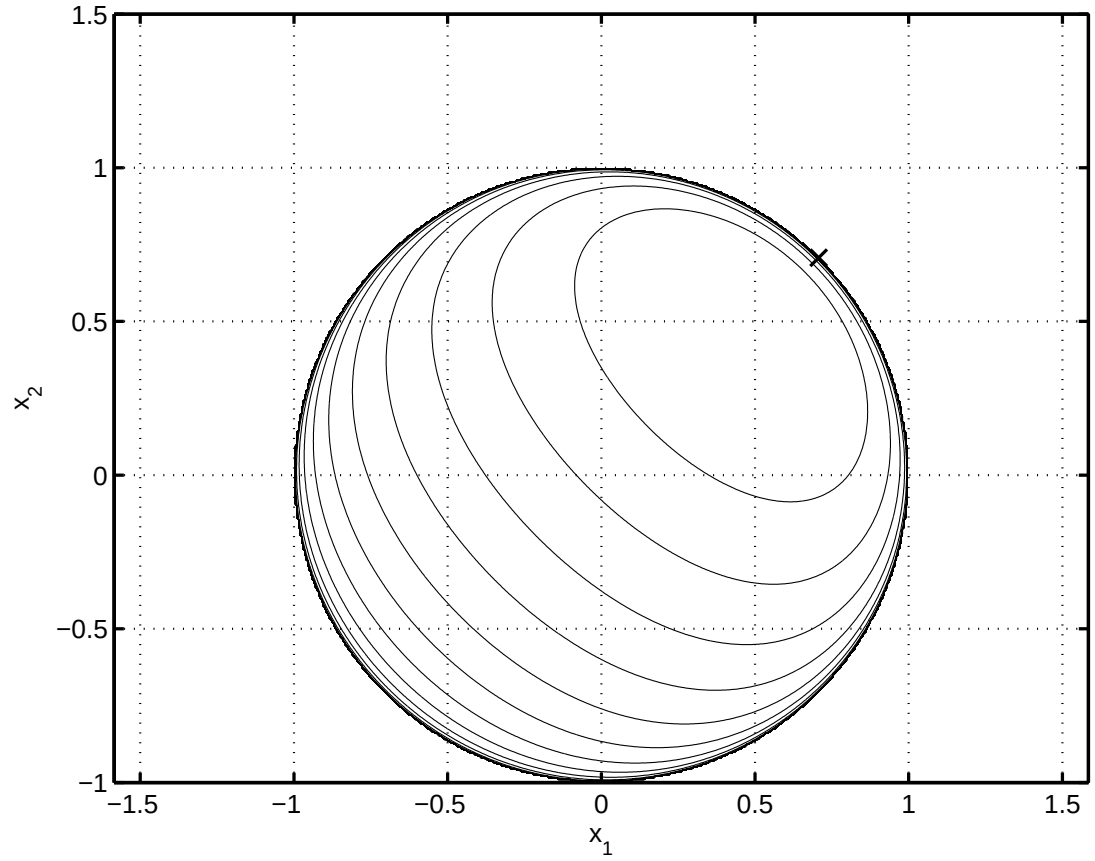
$t = 0.5$

Level sets for $\phi_t(x_1, x_2) = -x_1 - x_2 - (1/t)\log\left(1 - x_1^2 - x_2^2\right)$



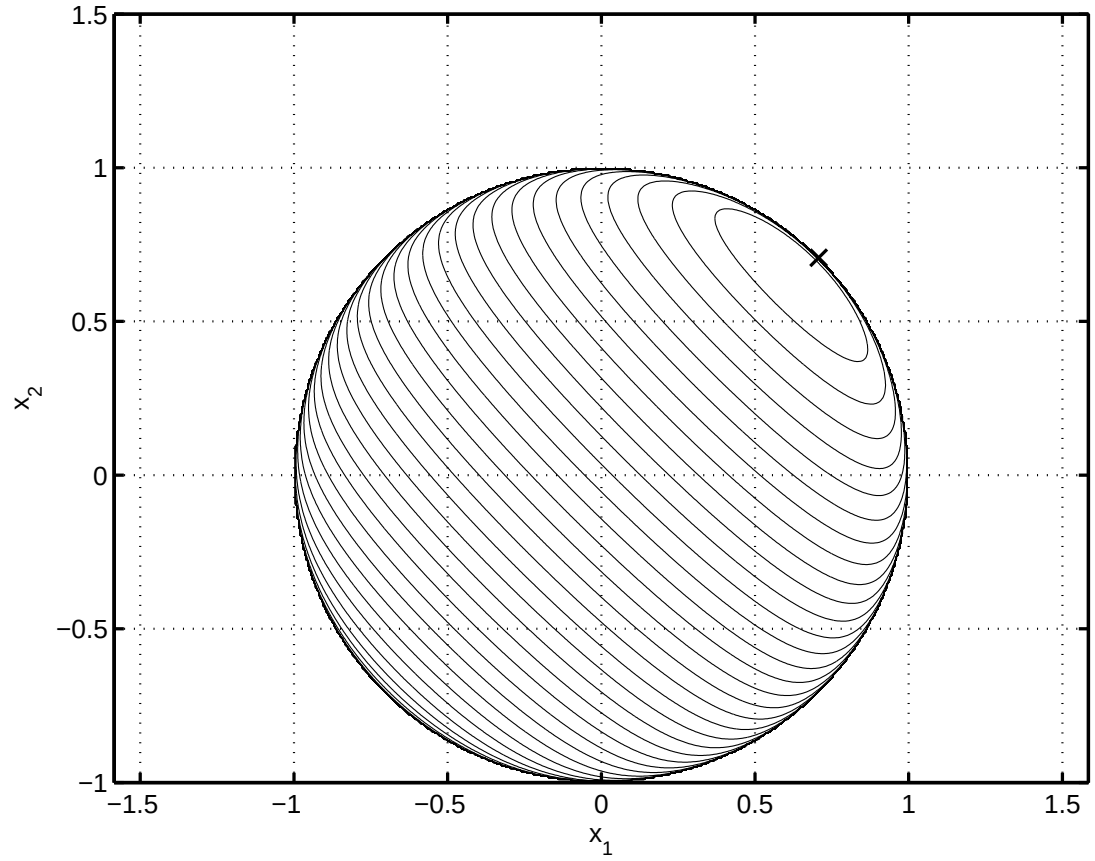
$t = 1$

Level sets for $\phi_t(x_1, x_2) = -x_1 - x_2 - (1/t)\log\left(1 - x_1^2 - x_2^2\right)$



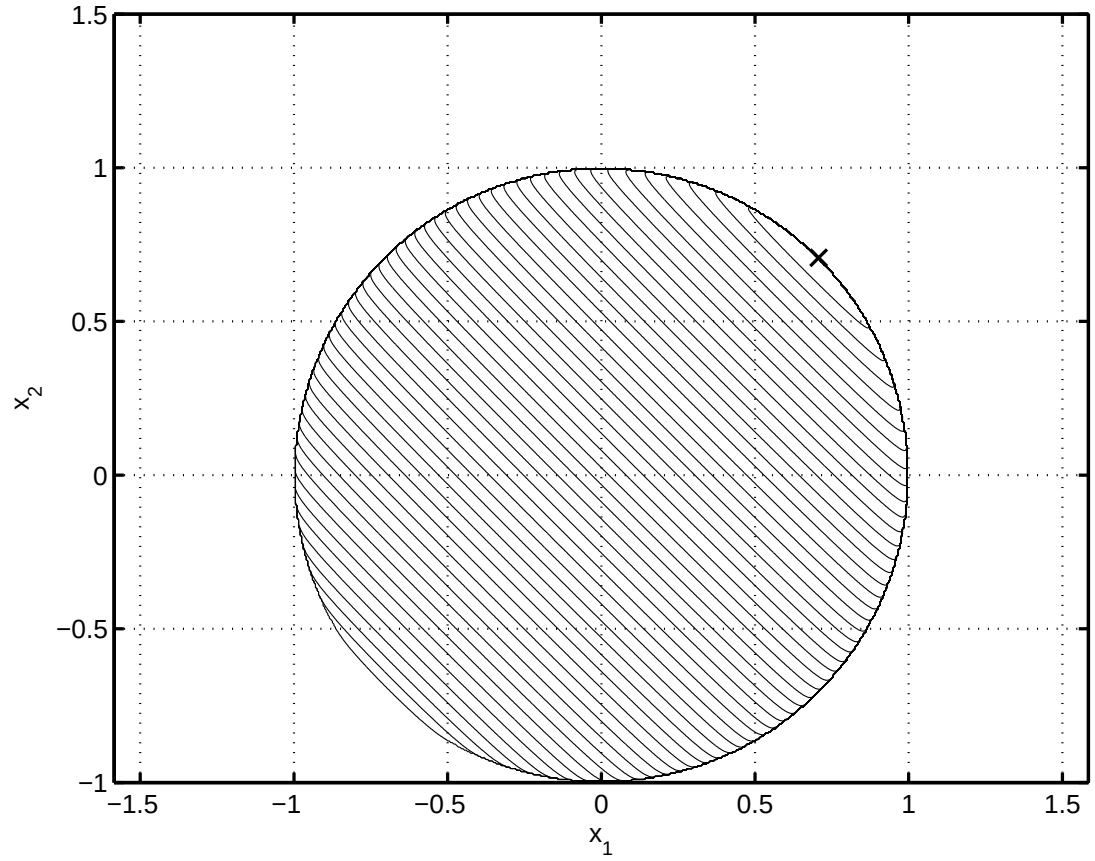
$t = 2$

Level sets for $\phi_t(x_1, x_2) = -x_1 - x_2 - (1/t)\log\left(1 - x_1^2 - x_2^2\right)$



$t = 10$

Level sets for $\phi_t(x_1, x_2) = -x_1 - x_2 - (1/t)\log\left(1 - x_1^2 - x_2^2\right)$



$t = 100$

- Assume an unique minimizer $x^\star(t)$ for

$$\begin{array}{ll} \text{minimize} & f(x) + \sum_{i=1}^{m} -(1/t)\log\left(-g_i(x)\right) \\ \text{subject to} & Ax = b \end{array}$$

- $x^\star(t)$ is (at most) m/t suboptimal:

$$f\left(x^\star(t)\right) - p^\star \leq \frac{m}{t}$$

- Central path $= \{x^\star(t) \,:\, t > 0\}$
- Following the central path leads to a solution

1. **initialization** ▶ $t_0 > 0$, strictly feasible x_0, $\mu > 1$, $\epsilon > 0$

2. ▶ initialize $k = 0$

3. **loop** ▶ solve $\min_{Ax=b} t_k f(x) + \sum_{i=1}^{m} -\log(-g_i(x))$
   starting from x_k

4. ▶ $x_{k+1} = x^\star(t_k)$

5. ▶ if $m/t_k < \epsilon$ then stop

6. ▶ $t_{k+1} = \mu t_k$

7. ▶ $k \leftarrow k + 1$ and return to loop

- To construct a strictly feasible point x_0 use a "Phase I" problem:

$$\begin{array}{ll} \text{minimize} & s \\ \text{subject to} & Ax = b \\ & g_i(x) \leq s \quad i = 1, \ldots, m \end{array}$$

- Can start from $x \in \bigcap_{i=1}^m \text{dom}\, f_i$ with $Ax = b$ and

$$s > \max\{g_i(x) \,:\, i = 1, \ldots, m\}$$

Optimization problem:

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & Ax = b \\ & g_i(x) \preceq_{K_i} 0 \quad i = 1, \ldots, m \end{array}$$

Assumptions:

- K_i are proper cones
- f is convex and g_i is K_i-convex
- dom f, dom g_i are open and f, g_i are twice continuously differentiable
- problem is solvable:

  $\exists\, x^\star \,:\, f(x^\star) = p^\star := \inf\{f(x) \,:\, Ax = b,\, g_i(x) \preceq_{K_i} 0,\, i = 1, \ldots, m\}$

- matrix A has full row rank
- there is a Slater point:
  $\exists_{x \in \text{dom}\, f} \,:\, Ax = b,\, g_i(x) \prec_{K_i} 0,\, i = 1, \ldots, m$

A generalized logarithm for a proper cone $K \subset \mathbb{R}^n$ is a function

$$\psi : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\} \quad \mathsf{dom}\, \psi = \mathsf{int}\, K$$

such that

- ψ is closed, concave, twice continuously differentiable with

$$\nabla^2 \psi(x) \prec 0 \quad \text{for all } x \succ_K 0$$

- there is $\theta > 0$ (degree of ψ) such that

$$\psi(tx) = \psi(x) + \theta \log(t) \text{ for all } x \succ_K 0 \text{ and } t > 0$$

Examples:

- nonnegative orthant: $K = \mathbb{R}^n_+$

$$\psi(x_1, \ldots, x_n) = \sum_{i=1}^{n} \log(x_i)$$

  degree $\theta = n$

- second-order cone: $K = \{(x, t) \in \mathbb{R}^n \times \mathbb{R} \,:\, \|x\| \leq t\}$

$$\psi(x, t) = \log\left(t^2 - x^\top x\right)$$

  degree $\theta = 2$

- positive semidefinite cone: $K = \mathsf{S}^n_+$

$$\psi(X) = \log\det(X)$$

  degree $\theta = n$

- Consider the approximation

$$\begin{array}{ll} \text{minimize} & tf(x) + \sum_{i=1}^{m} -\psi_i\left(-g_i(x)\right) \\ \text{subject to} & Ax = b \end{array}$$

- Assume an unique minimizer $x^\star(t)$
- $x^\star(t)$ is (at most) $(\sum_{i=1}^{m} \theta_i)/t$ suboptimal:

$$f\left(x^\star(t)\right) - p^\star \leq \frac{\sum_{i=1}^{m} \theta_i}{t}$$

- Central path $= \{x^\star(t) \,:\, t > 0\}$
- Following the central path leads to a solution

- Example:

$$\begin{array}{ll} \text{minimize} & \mathsf{tr}(AX) \\ \text{subject to} & \mathsf{diag}(X) = 1 \\ & X \succeq 0 \end{array}$$

- Log barrier subproblem:

$$\begin{array}{ll} \text{minimize} & t\,\mathsf{tr}(AX) - \log\det(X) \\ \text{subject to} & \mathsf{diag}(X) = 1 \end{array}$$

  with implicit constraint $X \succ 0$

- KKT conditions for subproblem:

$$\begin{cases} tA - X^{-1} + \mathsf{Diag}(\lambda) = 0 \\ \mathsf{diag}(X) = 1 \end{cases}$$

  with $X \succ 0$

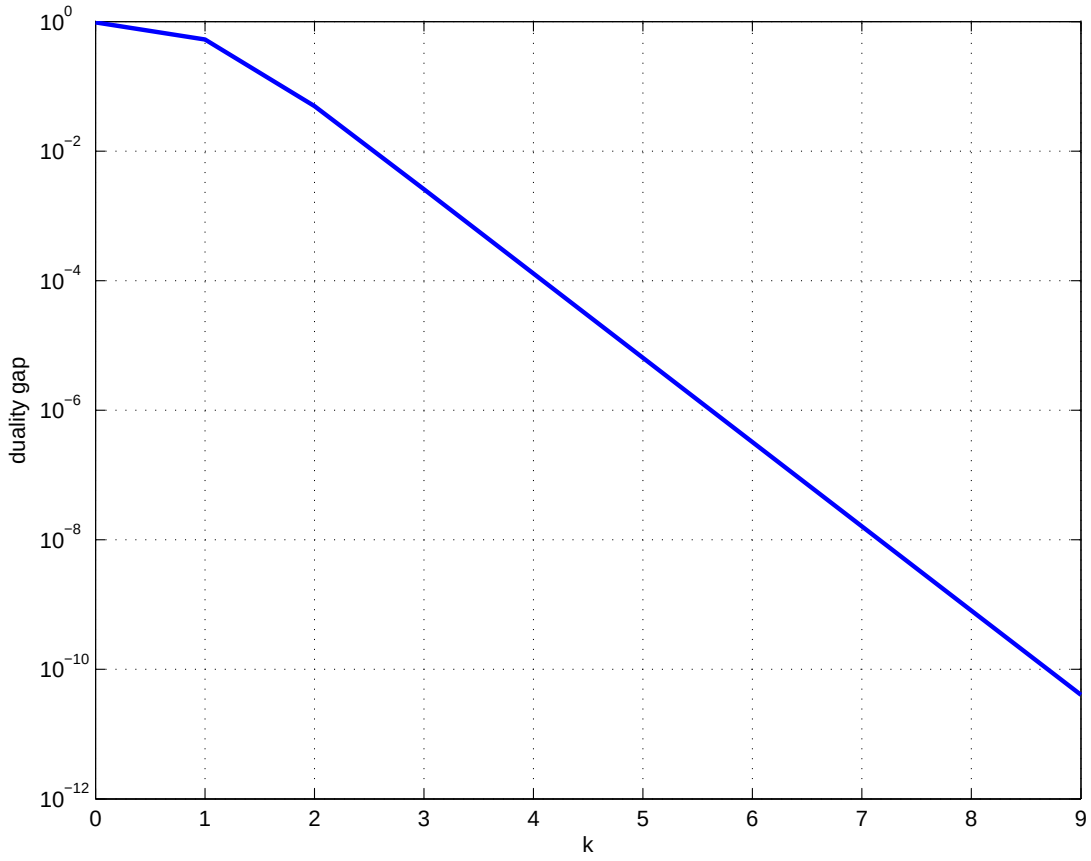
- Linearizing around a feasible X:

$$\begin{cases} tA - X^{-1} + X^{-1}\Delta X^{-1} + \mathsf{Diag}(\lambda) = 0 \\ \mathsf{diag}(\Delta) = 0 \end{cases}$$

- Solution:

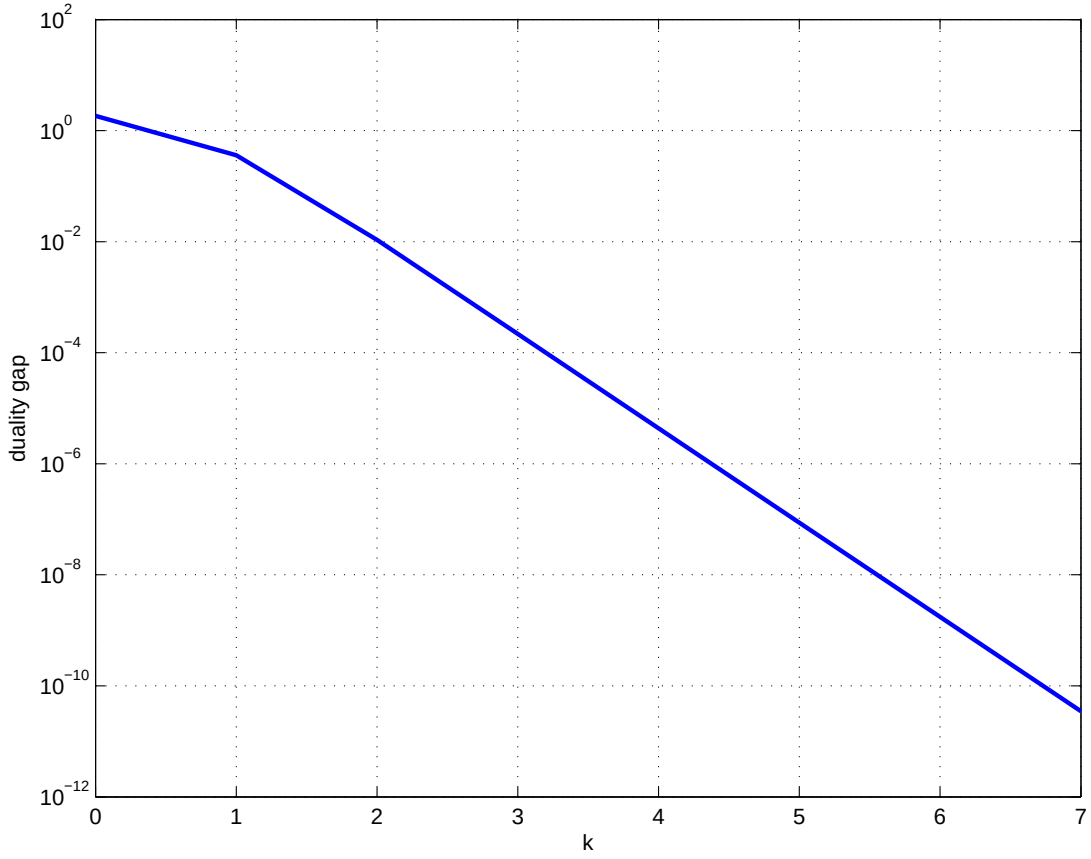
$$\begin{cases} (X \odot X)\,\lambda = X - tXAX \\ \Delta = X - tXAX - X\mathsf{Diag}(\lambda)X \end{cases}$$

Problem size $n = 20$ $(\mu = 20)$



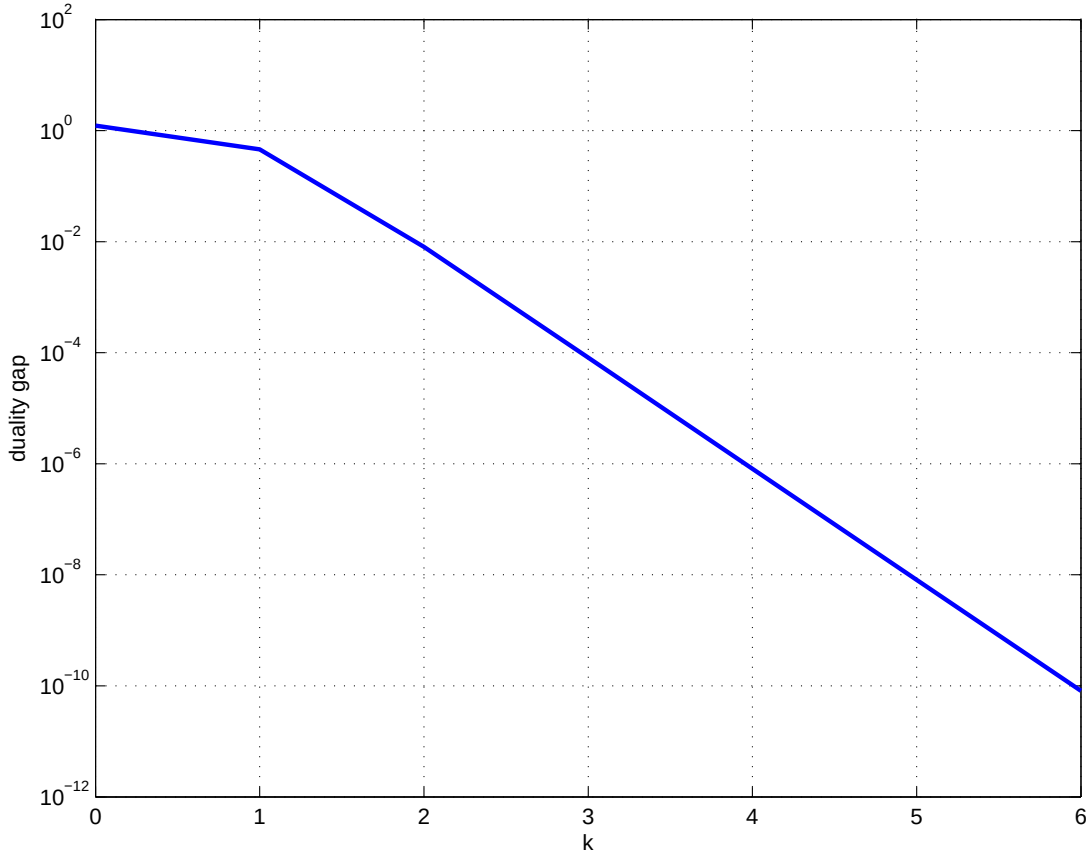
Time $\simeq 0.06$ sec

Problem size $n = 20$ ($\mu = 50$)



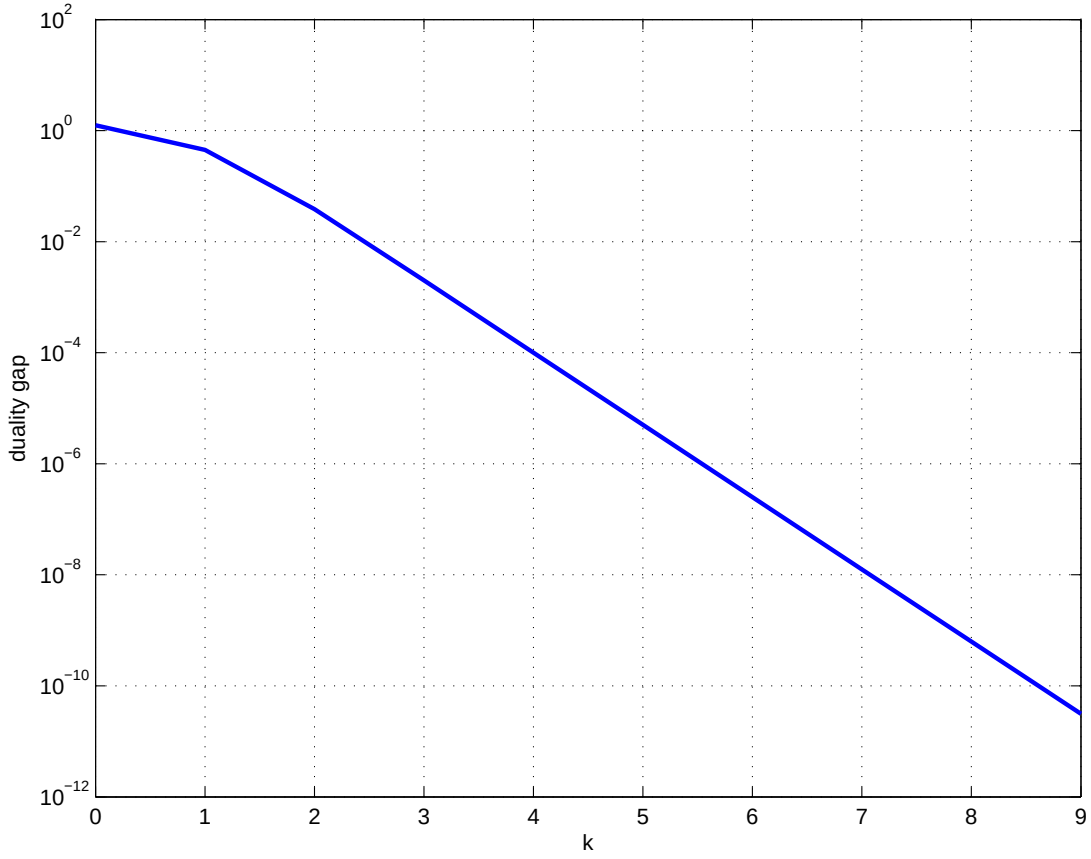
Time $\simeq 0.06$ sec

Problem size $n = 20$ ($\mu = 100$)



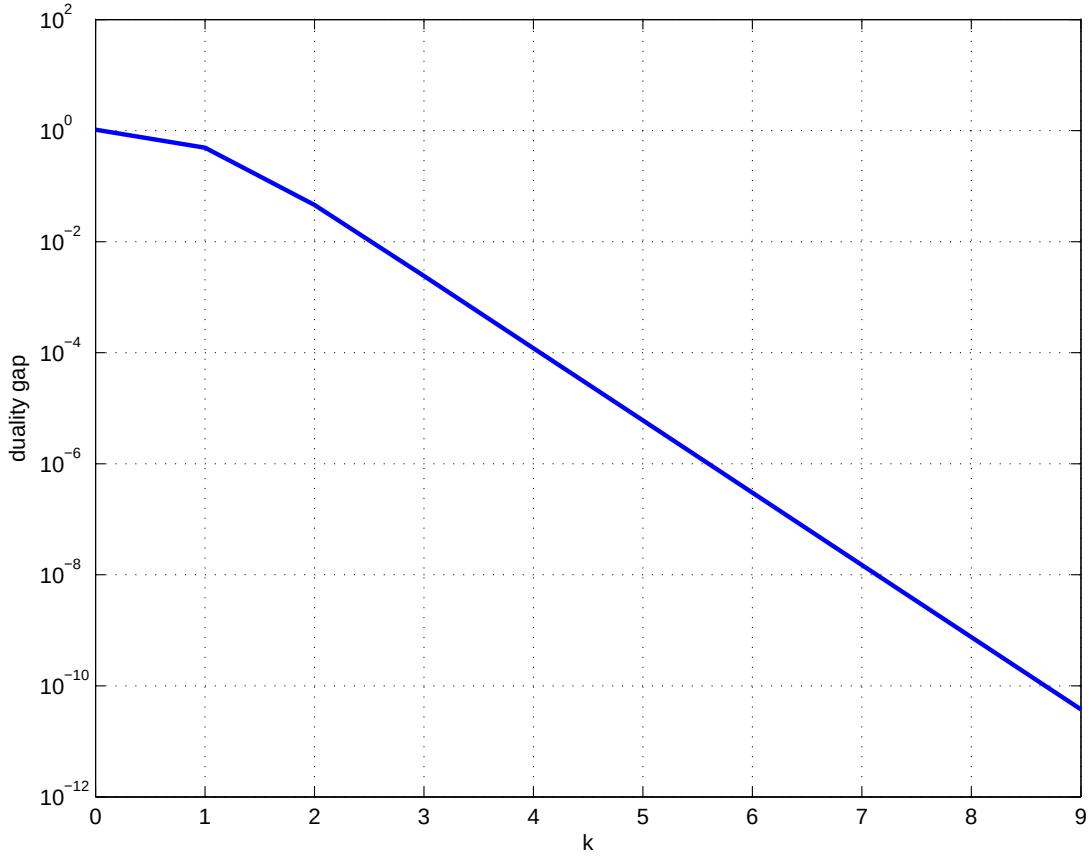
Time $\simeq 0.06$ sec

Problem size $n = 100$ ($\mu = 20$)



Time $\simeq 1.2$ sec

Problem size $n = 150$ ($\mu = 20$)



Time $\simeq 3.4$ sec

Thank you !

Special thanks to **João Paulo** and **Rob**