10. Unconstrained minimization

- terminology and assumptions
- gradient descent method
- steepest descent method
- Newton’s method
- self-concordant functions
- implementation
**Unconstrained minimization**

minimize \( f(x) \)

- \( f \) convex, twice continuously differentiable (hence \( \text{dom} \ f \) open)
- we assume optimal value \( p^* = \inf_x f(x) \) is attained (and finite)

**Unconstrained minimization methods**

- produce sequence of points \( x^{(k)} \in \text{dom} \ f, \ k = 0, 1, \ldots \), with
  \[
  f(x^{(k)}) \to p^*
  \]

- can be interpreted as iterative methods for solving optimality condition
  \[
  \nabla f(x^*) = 0
  \]
Initial point and sublevel set

algorithms in this chapter require a starting point \(x^{(0)}\) such that

- \(x^{(0)} \in \text{dom } f\)
- sublevel set \(S = \{x \mid f(x) \leq f(x^{(0)})\}\) is closed

2nd condition is hard to verify, except when all sublevel sets are closed:

- equivalent to condition that \(\text{epi } f\) is closed
- true if \(\text{dom } f = \mathbb{R}^n\)
- true if \(f(x) \to \infty\) as \(x \to \text{bd dom } f\)

examples of differentiable functions with closed sublevel sets:

\[
f(x) = \log\left(\sum_{i=1}^{m} \exp(a_i^T x + b_i)\right), \quad f(x) = -\sum_{i=1}^{m} \log(b_i - a_i^T x)
\]
Strong convexity and implications

$f$ is strongly convex on $S$ if there exists an $m > 0$ such that

$$\nabla^2 f(x) \succeq m I \quad \text{for all } x \in S$$

**Implications**

- for $x, y \in S$,
  $$f(y) \geq f(x) + \nabla f(x)^T (y - x) + \frac{m}{2} \|x - y\|_2^2$$
- $S$ is bounded
- $p^* > -\infty$ and for $x \in S$,
  $$f(x) - p^* \leq \frac{1}{2m} \|\nabla f(x)\|_2^2$$

useful as stopping criterion (if you know $m$)
Descent methods

\[ x^{(k+1)} = x^{(k)} + t^{(k)} \Delta x^{(k)} \quad \text{with } f(x^{(k+1)}) < f(x^{(k)}) \]

• other notations:
  \[ x^+ = x + t\Delta x, \quad x := x + t\Delta x \]

• \( \Delta x \) is the step, or search direction; \( t \) is the step size, or step length

• for convex \( f \): if \( f(x^+) < f(x) \) then \( \Delta x \) must be a descent direction:

\[ \nabla f(x)^T \Delta x < 0 \]

General descent method

given: a starting point \( x \in \text{dom } f \)
repeat
  1. determine a descent direction \( \Delta x \)
  2. line search: choose a step size \( t > 0 \)
  3. update: \( x := x + t\Delta x \)
until stopping criterion is satisfied
Line search types

**Exact line search:** \( t = \arg\min_{t>0} f(x + t\Delta x) \)

**Backtracking line search** (with parameters \( \alpha \in (0, 1/2), \beta \in (0, 1) \))

- starting at \( t = 1 \), repeat \( t := \beta t \) until
  \[
  f(x + t\Delta x) < f(x) + \alpha t \nabla f(x)^T \Delta x
  \]

- graphical interpretation: backtrack until \( t \leq t_0 \)
Gradient descent method

**Gradient descent:** general descent method with $\Delta x = -\nabla f(x)$

given: a starting point $x \in \text{dom } f$
repeat
1. $\Delta x := -\nabla f(x)$
2. *line search:* choose step size $t$ via exact or backtracking line search
3. *update:* $x := x + t\Delta x$
until stopping criterion is satisfied

- stopping criterion usually of the form $\|\nabla f(x)\|_2 \leq \epsilon$
- convergence result: for strongly convex $f$,

\[ f(x^{(k)}) - p^* \leq c^k (f(x^{(0)}) - p^*) \]

$c \in (0, 1)$ depends on $m, x^{(0)}$, line search type

- very simple, but often very slow
Quadratic problem in $\mathbb{R}^2$

$$f(x) = \frac{1}{2}(x_1^2 + \gamma x_2^2) \quad (\gamma > 0)$$

with exact line search, starting at $x^{(0)} = (\gamma, 1)$:

$$x_1^{(k)} = \gamma \left(\frac{\gamma - 1}{\gamma + 1}\right)^k, \quad x_2^{(k)} = \left(-\frac{\gamma - 1}{\gamma + 1}\right)^k$$

- very slow if $\gamma \gg 1$ or $\gamma \ll 1$
- example for $\gamma = 10$:
Nonquadratic example

\[ f(x_1, x_2) = e^{x_1 + 3x_2 - 0.1} + e^{x_1 - 3x_2 - 0.1} + e^{-x_1 - 0.1} \]
Example in $\mathbb{R}^{100}$

\[ f(x) = c^T x - \sum_{i=1}^{500} \log(b_i - a_i^T x) \]

'linear' convergence, \textit{i.e.}, a straight line on a semilog plot
Steepest descent method

Normalized steepest descent direction (at $x$, for norm $\| \cdot \|$):

$$\Delta x_{\text{nsd}} = \arg\min \{ \nabla f(x)^T v \mid \|v\| = 1 \}$$

interpretation: for small $v$,

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

direction $\Delta x_{\text{nsd}}$ is unit-norm step with most negative directional derivative

(Unnormalized) steepest descent direction

$$\Delta x_{\text{sd}} = \| \nabla f(x) \|_* \Delta x_{\text{nsd}}$$

satisfies $\nabla f(x)^T \Delta x_{\text{sd}} = -\| \nabla f(x) \|_*^2$

Steepest descent method

- general descent method with $\Delta x = \Delta x_{\text{sd}}$
- convergence properties similar to gradient descent
Examples

- Euclidean norm: $\Delta x_{sd} = -\nabla f(x)$
- Quadratic norm $\|x\|_P = (x^T P x)^{1/2}$ ($P \in S_+^n$):
  $$\Delta x_{sd} = -P^{-1} \nabla f(x)$$
- $\ell_1$-norm: $\Delta x_{sd} = -(\partial f(x)/\partial x_i) e_i$, where $|\partial f(x)/\partial x_i| = \|\nabla f(x)\|_\infty$

Unit balls, steepest descent directions for a quadratic norm and $\ell_1$-norm:
Choice of norm for steepest descent

- steepest descent with backtracking line search for two quadratic norms
- ellipses show \( \{ x \mid \|x - x^{(k)}\|_P = 1 \} \)
- equivalent interpretation of steepest descent with quadratic norm \( \| \cdot \|_P \): gradient descent after change of variables \( \bar{x} = P^{1/2}x \)

shows choice of \( P \) has strong effect on speed of convergence
Newton step

\[ \Delta x_{nt} = -\nabla^2 f(x)^{-1} \nabla f(x) \]

Interpretations

- \( x + \Delta x_{nt} \) minimizes second order approximation

\[ \tilde{f}(x + v) = f(x) + \nabla f(x)^T v + \frac{1}{2} v^T \nabla^2 f(x) v \]

- \( x + \Delta x_{nt} \) solves linearized optimality condition

\[ \nabla f(x + v) \approx \nabla \tilde{f}(x + v) = \nabla f(x) + \nabla^2 f(x) v = 0 \]
• \( \Delta x_{nt} \) is steepest descent direction at \( x \) in local Hessian norm

\[
\|u\|_{\nabla^2 f(x)} = (u^T \nabla^2 f(x) u)^{1/2}
\]

dashed lines are contour lines of \( f \); ellipse is \( \{x + v \mid v^T \nabla^2 f(x) v = 1\} \)

arrow shows \(-\nabla f(x)\)
Newton decrement

$$\lambda(x) = (\nabla f(x)^T \nabla^2 f(x)^{-1} \nabla f(x))^{1/2}$$

a measure of the proximity of $x$ to $x^*$

Properties

• gives an estimate of $f(x) - p^*$, using quadratic approximation $\tilde{f}$:

$$f(x) - \inf_y \tilde{f}(y) = \frac{1}{2} \lambda(x)^2$$

• equal to the norm of the Newton step in the quadratic Hessian norm

$$\lambda(x) = (\Delta x_{nt}^T \nabla^2 f(x) \Delta x_{nt})^{1/2}$$

• directional derivative in the Newton direction: $\nabla f(x)^T \Delta x_{nt} = -\lambda(x)^2$

• affine invariant (unlike $\|\nabla f(x)\|_2$)
Newton's method

given: a starting point $x \in \text{dom } f$, tolerance $\epsilon > 0$
repeat
1. compute the Newton step and decrement
   \[ \Delta x_{nt} := -\nabla^2 f(x)^{-1} \nabla f(x); \quad \lambda^2 := \nabla f(x)^T \nabla^2 f(x)^{-1} \nabla f(x) \]
2. stopping criterion: quit if $\lambda^2 / 2 \leq \epsilon$
3. line search: choose step size $t$ by backtracking line search
4. update: $x := x + t \Delta x_{nt}$

Affine invariance

- Newton iterates for $\tilde{f}(y) = f(Ty)$ with starting point $y^{(0)} = T^{-1}x^{(0)}$ are
  \[ y^{(k)} = T^{-1}x^{(k)} \]
- independent of linear changes of coordinates
Classical convergence analysis

Assumptions

- \( f \) strongly convex on \( S \) with constant \( m \)
- \( \nabla^2 f \) is Lipschitz continuous on \( S \), with constant \( L > 0 \):
  \[
  \| \nabla^2 f(x) - \nabla^2 f(y) \|_2 \leq L \| x - y \|_2
  \]
  
  \( (L \) measures how well \( f \) can be approximated by a quadratic function)\n
Outline: there exist constants \( \eta \in (0, m^2/L), \gamma > 0 \) such that

- if \( \| \nabla f(x) \|_2 \geq \eta \), then \( f(x^{(k+1)}) - f(x^{(k)}) \leq -\gamma \)
- if \( \| \nabla f(x) \|_2 < \eta \), then
  \[
  \frac{L}{2m^2} \| \nabla f(x^{(k+1)}) \|_2 \leq \left( \frac{L}{2m^2} \| \nabla f(x^{(k)}) \|_2 \right)^2
  \]
Classical convergence analysis

**Damped Newton phase** \((\|\nabla f(x)\|_2 \geq \eta)\)

- most iterations require backtracking steps
- function value decreases by at least \(\gamma\)
- if \(p^* > -\infty\), this phase ends after at most \((f(x^{(0)}) - p^*)/\gamma\) iterations

**Quadratically convergent phase** \((\|\nabla f(x)\|_2 < \eta)\)

- all iterations use step size \(t = 1\)
- \(\|\nabla f(x)\|_2\) converges to zero quadratically: if \(\|\nabla f(x^{(k)})\|_2 < \eta\), then

\[
\frac{L}{2m^2}\|\nabla f(x^l)\|_2 \leq \left(\frac{L}{2m^2}\|\nabla f(x^k)\|_2\right)^{2^{l-k}} \leq \left(\frac{1}{2}\right)^{2^{l-k}}, \quad l \geq k
\]
Classical convergence analysis

Conclusion: number of iterations until $f(x) - p^* \leq \epsilon$ is bounded above by

$$\frac{f(x^{(0)}) - p^*}{\gamma} + \log_2 \log_2(\epsilon_0/\epsilon)$$

- $\gamma, \epsilon_0$ are constants that depend on $m, L, x^{(0)}$
- second term is small (of the order of 6) and almost constant for practical purposes
- in practice, constants $m, L$ (hence $\gamma, \epsilon_0$) are usually unknown
- provides qualitative insight in convergence properties (i.e., explains two algorithm phases)
Examples

Example in $\mathbb{R}^2$ (page 10.9)

- backtracking parameters $\alpha = 0.1$, $\beta = 0.7$
- converges in only 5 steps
- quadratic local convergence
Examples

Example in $\mathbb{R}^{100}$ (page 10.10)

- backtracking parameters $\alpha = 0.01$, $\beta = 0.5$
- backtracking line search almost as fast as exact l.s. (and much simpler)
- clearly shows two phases in algorithm
Examples

Example in $\mathbb{R}^{10000}$ (with sparse $a_i$)

$$f(x) = -\sum_{i=1}^{10000} \log(1 - x_i^2) - \sum_{i=1}^{10000} \log(b_i - a_i^T x)$$

- backtracking parameters $\alpha = 0.01$, $\beta = 0.5$
- performance similar as for small examples
Self-concordance

Shortcomings of classical convergence analysis

- depends on unknown constants \((m, L, \ldots)\)
- bound is not affinely invariant, although Newton’s method is

Convergence analysis via self-concordance (Nesterov and Nemirovski)

- does not depend on any unknown constants
- gives affine-invariant bound
- applies to special class of convex functions (‘self-concordant’ functions)
- developed to analyze polynomial-time interior-point methods for convex optimization
Self-concordant functions

Definition

- convex \( f : \mathbb{R} \rightarrow \mathbb{R} \) is self-concordant if
  \[
  |f'''(x)| \leq 2f''(x)^{3/2} \text{ for all } x \in \text{dom } f
  \]

- \( f : \mathbb{R}^n \rightarrow \mathbb{R} \) is self-concordant if \( g(t) = f(x + tv) \) is s.c. for all \( x \in \text{dom } f \) and \( v \)

Examples on \( \mathbb{R} \)

- linear and quadratic functions
- negative logarithm \( f(x) = -\log x \)
- negative entropy plus negative logarithm: \( f(x) = x \log x - \log x \)

Affine invariance: if \( f : \mathbb{R} \rightarrow \mathbb{R} \) is s.c., then \( \tilde{f}(y) = f(ay + b) \) is s.c.:

\[
\tilde{f}'''(y) = a^3 f'''(ay + b), \quad \tilde{f}''(y) = a^2 f''(ay + b)
\]
Self-concordant calculus

Properties

• preserved under sums and positive scaling by factor $\geq 1$
• preserved under composition with affine function
• if $g$ is convex with $\text{dom } g = \mathbb{R}_{++}$ and $|g'''(x)| \leq 3g''(x)/x$ then

$$f(x) = \log(-g(x)) - \log x$$

is self-concordant

Examples: properties can be used to show that the following are s.c.

• $f(x) = -\sum_{i=1}^{m} \log(b_i - a_i^T x)$ on $\{x \mid a_i^T x < b_i, \ i = 1, \ldots, m\}$
• $f(X) = -\log \det X$ on $\mathbb{S}_{++}^{n}$
• $f(x) = -\log(y^2 - x^T x)$ on $\{(x, y) \mid \|x\|_2 < y\}$
Convergence analysis for self-concordant functions

**Summary:** there exist constants $\eta \in (0, 1/4]$, $\gamma > 0$ such that

- if $\lambda(x) > \eta$, then
  \[ f(x^{(k+1)}) - f(x^{(k)}) \leq -\gamma \]
- if $\lambda(x) \leq \eta$, then
  \[ 2\lambda(x^{(k+1)}) \leq \left(2\lambda(x^{(k)})\right)^2 \]

($\eta$ and $\gamma$ only depend on backtracking parameters $\alpha$, $\beta$)

**Complexity bound:** number of Newton iterations bounded by

\[ \frac{f(x^{(0)}) - p^*}{\gamma} + \log_2 \log_2(1/\epsilon) \]

for $\alpha = 0.1$, $\beta = 0.8$, $\epsilon = 10^{-10}$, bound evaluates to $375(f(x^{(0)}) - p^*) + 6$
Numerical example

150 randomly generated instances of

\[ \text{minimize } f(x) = - \sum_{i=1}^{m} \log(b_i - a_i^T x) \]

○: \( m = 100, n = 50 \)
□: \( m = 1000, n = 500 \)
◇: \( m = 1000, n = 50 \)

- number of iterations much smaller than \( 375(f(x^{(0)}) - p^*) + 6 \)
- bound of the form \( c(f(x^{(0)}) - p^*) + 6 \) with smaller \( c \) (empirically) valid
Implementation

main effort in each iteration: evaluate derivatives and solve Newton system

\[ H \Delta x = -g \]

where \( H = \nabla^2 f(x), \ g = \nabla f(x) \)

Via Cholesky factorization

\[ H = LL^T, \quad \Delta x_{nt} = -L^{-T}L^{-1}g, \quad \lambda(x) = \|L^{-1}g\|_2 \]

• cost \((1/3)n^3\) flops for unstructured system
• cost \(\ll (1/3)n^3\) if \(H\) sparse, banded
Example of dense Newton system with structure

\[ f(x) = \sum_{i=1}^{n} \psi_i(x_i) + \psi_0(Ax + b), \quad H = D + A^T H_0 A \]

- assume \( A \in \mathbb{R}^{p \times n} \), dense, with \( p \ll n \)
- \( D \) diagonal with diagonal elements \( \psi_i''(x_i) \); \( H_0 = \nabla^2 \psi_0(Ax + b) \)

**Method 1**: form \( H \), solve via dense Cholesky factorization (cost \((1/3)n^3\))

**Method 2** (page 9.15): factor \( H_0 = L_0 L_0^T \); write Newton system as

\[ D \Delta x + A^T L_0 w = -g, \quad L_0^T A \Delta x - w = 0 \]

eliminate \( \Delta x \) from first equation; compute \( w \) and \( \Delta x \) from

\[ (I + L_0^T A D^{-1} A^T L_0)w = -L_0^T A D^{-1} g, \quad D \Delta x = -g - A^T L_0 w \]

cost: \( 2p^2 n \) (dominated by computation of \( L_0^T A D^{-1} A^T L_0 \)