10. Unconstrained minimization

- terminology and assumptions
- gradient descent method
- steepest descent method
- Newton’s method
- 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)

We will assume that \( x^* = \arg\min_x \, f(x) \) exists and is unique
Recall \( p^* = f(x^*) \)

unconstrained minimization methods

• produce sequence of points \( x^{(k)} \in \text{dom} \, f, \, k = 0, 1, \ldots \) with

\[
    f(x^{(k)}) \to p^* \quad \text{as} \, \, k \to \infty
\]

\( x^{(0)}, x^{(1)}, \ldots \) is a minimizing sequence to the problem
Algorithm stops when \( f(x^{(k)}) - p^* \leq \epsilon \), for some tolerance \( \epsilon > 0 \)

• can be interpreted as iterative methods for solving optimality condition

\[
    \nabla f(x^*) = 0
\]
Strong convexity and implications

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

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

implications

- for $x, y \in S$,

$$f(y) \geq f(x) + \nabla f(x)^T (y - x) + \frac{m}{2} \|x - y\|_2^2$$

hence, $S$ is bounded

Assume $f$ is twice differentiable

By Taylor’s theorem, there exists a $z$ in the line segment from $x$ to $y$ such that $f(y) = f(x) + df(x)'(y-x) + \frac{1}{2} (y-x)' d^2 f(z) (y-x) \\
\geq f(x) + df(x)'(y-x) + \frac{1}{2} (y-x)' (m I) (y-x) \quad \text{... since } f \text{ is strongly convex} \\
= f(x) + df(x)'(y-x) + \frac{1}{2} m |y-x|_2^2^2$$

(Taylor’s theorem is a generalization of the mean value theorem, and is very related to, but is not exactly the same as Taylor series)
Descent methods

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

- other notations: \( x^+ = x + t \Delta x \), \( x := x + t \Delta x \)
- \( \Delta x \) is the step, or search direction; \( t \) is the step size, or step length
- from convexity, \( f(x^+) < f(x) \) implies \( \nabla f(x)^T \Delta x < 0 \) (i.e., \( \Delta x \) is a descent direction)

General descent method.

given a starting point \( x \in \text{dom} \ f \).
repeat
1. Determine a descent direction \( \Delta x \). (Each algorithm has its own way for choosing \( \Delta x \))
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  

\[
f(x + t\Delta x) < f(x) + \alpha t \nabla f(x)^T \Delta x
\]

*(Armijo–Goldstein condition)*

Since \( \Delta x \) is a descent direction (see previous slide) then \( df(x)’\Delta x < 0 \) 
For small \( t \), we have:

\[
f(x + t \Delta x) \approx f(x) + t \, df(x)’\Delta x < f(x) + \alpha \, t \, df(x)’\Delta x
\]

Thus, the procedure will eventually terminate.
Gradient descent method

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 $f$,
  \[ f(x^{(k)}) - p^* \leq c^k (f(x^{(0)}) - p^*) \]  
  (linear convergence)
  
  $c \in (0, 1)$ depends on $m$, $x^{(0)}$, line search type
- very simple, but often very slow; rarely used in practice

Unconstrained minimization
quadratic problem in $\mathbb{R}^2$

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

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 $\gamma = 10$:

![Graph showing the iterative process for the quadratic problem with $\gamma = 10$. The graph illustrates the movement of the iterations from the initial point $x^{(0)}$ to $x^{(6)}$.](image-url)
nonquadratic example

\[ f(x_1, x_2) = e^{x_1 + 3x_2 - 0.1} + e^{x_1 - 3x_2 - 0.1} + e^{-x_1 - 0.1} \]

backtracking line search

exact line search
a problem in $\mathbf{R}^{100}$

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

Unconstrained minimization

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

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

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

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

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

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 and normalized steepest descent directions for a quadratic norm and the $\ell_1$-norm:
choice of norm for steepest descent

- steepest descent with backtracking line search for two quadratic norms (two different P’s)
- 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
(Uses the Hessian as a good ellipse, see previous slide)

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

interpretations

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

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

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

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

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

Let $H = d^2 f(x)$
$$d = df(x)$$
From slide 10-11 we have:

$$\min d'u$$
$$\text{s.t. } u^T H u = 1$$

Let $u = H^{-1/2} s$

$$\min (H^{-1/2} d)'s$$
$$\text{s.t. } s's = 1$$

$L(s,v) = (H^{-1/2} d)'s + v (s's - 1)$
$$dL/ds = H^{-1/2} d + 2 v s = 0$$
$$s^* = -1/(2v) H^{-1/2} d$$

Then:

$$u^* = H^{-1/2} s^*$$
$$= -1/(2v) H^{-1/2} d$$

which is the direction of $\Delta x_{nt}$!

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

\[ \lambda(x) = \left( \nabla f(x)^T \nabla^2 f(x)^{-1} \nabla f(x) \right)^{1/2} \]

a measure of the proximity of \( x \) to \( x^* \)

properties

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

\[ f(x) - \inf_y \hat{f}(y) = \frac{1}{2} \lambda(x)^2 \]

Let \( H = d^2 f(x) \)
\[ d = df(x) \]
\[ \lambda = \lambda(x) \]
\[ \Delta x = \Delta x_{nt} = -H^{-1} d \]

\[ \inf_y f^\wedge(y) = f^\wedge(x + \Delta x) \]
\[ = f(x) + d' \Delta x + \frac{1}{2} \Delta x' H \Delta x \]
\[ = f(x) - \frac{1}{2} d' H^{-1} d \]

\[ f(x) - \inf_y f^\wedge(y) = \frac{1}{2} d' H^{-1} d = \frac{1}{2} \lambda^2 \]

Thus \( \lambda = \sqrt{d' H^{-1} d} \)

Remember \( p^* = \inf_y f(y) \)
Newton’s method

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

affine invariant, \textit{i.e.}, independent of linear changes of coordinates:

Newton iterates for \( \tilde{f}(y) = f(Ty) \) with starting point \( y^{(0)} = T^{-1}x^{(0)} \) are

\[
y^{(k)} = T^{-1}x^{(k)}
\]

\[
x = Ty \quad y = T^{\text{-}1}x
\]
Let \( \text{Hf} \sim(y) = d^2 f \sim(y) \)
Let \( \text{Hf} \sim(Ty) = T' \text{Hf}(x) T \)
\[
\text{df} \sim(y) = T' \text{df}(Ty) = T' \text{df}(x) \quad \Delta y = - \text{Hf} \sim(y)^{\text{-}1} \text{df}(y) = - (T' \text{Hf}(x) T)^{\text{-}1} T' \text{df}(x)
\]
\[
\text{Hf} \sim(y) = T' \text{Hf}(Ty) T = T' \text{Hf}(x) T \quad = - T^{\text{-}1} \text{Hf}(x)^{\text{-}1} \text{df}(x) = T^{\text{-}1} \Delta x
\]
\[
y^{(k)}(k) = y + \Delta y = T^{\text{-}1} (x + \Delta x) = T^{\text{-}1} x^{(k)}
\]
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

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

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