Optimization Methods
Lecture 2

Solmaz S. Kia
Mechanical and Aerospace Engineering Dept.
University of California Irvine
solmaz@uci.edu

Reading: Sections 7.1-7.5, 8.6, 8.8 of Ref[2].

Unconstrained optimization

$$x^\star = \underset{x \in \mathbb{R}^n}{\text{argmin}}\ f(x)$$

- $x^\star \in \mathbb{R}^n$ **Unconstrained local minimum of** f **if**

$$\exists\, \epsilon > 0 \ \text{ s.t. } \ f(x^\star) \leqslant f(x), \qquad \forall x \text{ with } \|x - x^\star\| < \epsilon,$$

- $x^\star \in \mathbb{R}^n$ **Unconstrained global minimum of** f **if**

$$f(x^\star) \leqslant f(x), \qquad \forall x \in \mathbb{R}^n,$$

- $x^\star \in \mathbb{R}^n$ **Unconstrained strict local minimum of** f **if**

$$\exists\, \epsilon > 0 \ \text{ s.t. } \ f(x^\star) < f(x), \qquad \forall x \text{ with } \|x - x^\star\| < \epsilon,$$

- $x^\star \in \mathbb{R}^n$ **Unconstrained strict global minimum of** f **if**

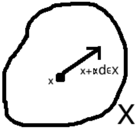
$$f(x^\star) < f(x), \qquad \forall x \in \mathbb{R}^n,$$

$$\text{OPT:} \quad x^\star = \underset{x \in \mathbb{R}^n}{\text{argmin}}\ f(x)$$

$$x \in X \quad (X \text{ is the set of constraints})$$

$$\text{for } X = \mathbb{R}^n \quad (\text{problem becomes unconstrained})$$

$D \in \mathbb{R}^n$ is a **feasible direction** at $x \in X$ for OPT if $(x + \alpha d) \in X$ for $\alpha \in [0, \bar{\alpha}]$



Proposition:

- **First order necessary condition (FONC)** consider OPT and let $f \in \mathcal{C}^1$ if $x^\star$ is a local minimizer for f then

$$\nabla f(x^\star)^\top d \geqslant 0, \quad \forall d \in \mathbb{R}^n, \quad d \text{ is a feasible direction}$$

- **Second order necessary condition (SONC)** let $f \in \mathcal{C}^2$ if $x^\star$ is a local minimizer for f then
  - **(i)** $\nabla f(x^\star)^\top d \geqslant 0$
  - **(ii)** if $\nabla f(x^\star) = 0 \quad \Rightarrow \quad d^\top \nabla^2 f(x^\star) d \geqslant 0\ \forall d \in \mathbb{R}^n, \quad d$ is a feasible direction

$$x^\star = \underset{x \in \mathbb{R}^n}{\text{argmin}}\ f(x)$$

Proposition (necessary optimality conditions)

Let $\mathbf{x}^\star$ be an unconstrained local minimum of $f : \mathbb{R}^n \to \mathbb{R}$ and assume that f is continuously differentiable in an open set S containing $\mathbf{x}^\star$, Then

$$\nabla f(\mathbf{x}^\star) = 0. \qquad \text{(First Order Necessary Condition)}$$

If in addition f is twice continuously differentiable within S, then

$$\nabla^2 f(\mathbf{x}^\star) : \text{positive semidefinite.} \qquad \text{(Second Order Necessary Condition)}$$

Proof: see page 13-14 of Ref[1].

Stationary point: Any point $\bar{\mathbf{x}} \in \mathbb{R}^n$ that satisfies $\nabla f(\bar{\mathbf{x}}) = 0$ is called a stationary point. A stationary point can be a minimum, maximum or saddle point of cost function f.

Sufficient conditions for optimality

$$x^\star = \underset{x \in \mathbb{R}^n}{\text{argmin}}\ f(x)$$

Proposition (Second order sufficient optimality conditions)

Let $f : \mathbb{R}^n \to \mathbb{R}$ be twice continuously differentiable in an open set S. Suppose that a vector $\mathbf{x}^\star$ satisfies the conditions

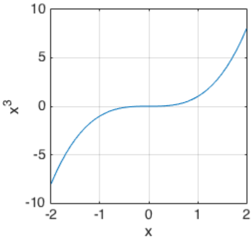
$$\nabla f(\mathbf{x}^\star) = 0, \qquad \nabla^2 f(\mathbf{x}^\star) : \text{positive definite.}$$

Then, $\mathbf{x}^\star$ is a strict unconstrained local minimum of f. In particular, there exist scalars $\gamma > 0$ and $\epsilon > 0$ such that

$$f(\mathbf{x}) \geqslant f(\mathbf{x}^\star) + \frac{\gamma}{2}\|\mathbf{x} - \mathbf{x}^\star\|^2, \qquad \forall \mathbf{x} \text{ with } \|\mathbf{x} - \mathbf{x}^\star\| < \epsilon.$$

Proof: see page 15 of Ref[1].

Stationary points: example



$f(x) = x^3$

$\nabla f(x) = 3x^2$

stationary point:

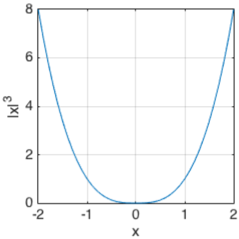
$\nabla f(0) = 0$

$x^\star = 0$ reflection point

— — — — —

$\nabla^2 f(x) = 6x$

$\nabla^2 f(0) = 0$



$f(x) = |x|^3$

$$\nabla f(x) = \begin{cases} 3x^2 & x > 0 \\ -3x^2 & x < 0 \end{cases}$$

stationary point:

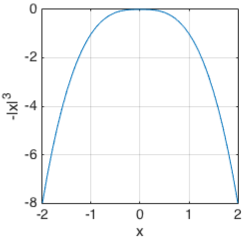
$\nabla f(0) = 0$

$x^\star = 0$ local minimizer

— — — — —

$$\nabla^2 f(x) = \begin{cases} 6x & x > 0 \\ -6x & x < 0 \end{cases}$$

$\nabla^2 f(0) = 0$



$f(x) = -|x|^3$

$$\nabla f(x) = \begin{cases} -3x^2 & x > 0 \\ 3x^2 & x < 0 \end{cases}$$

stationary point:

$\nabla f(0) = 0$

$x^\star = 0$ local maximizer

— — — — —

$$\nabla^2 f(x) = \begin{cases} -6x & x > 0 \\ 6x & x < 0 \end{cases}$$

$\nabla^2 f(0) = 0$

Note here that in all three of these cases $x^\star$ satisfies FONC and SONC, but satisfying necessary conditions does not mean that these points are minimizers. Note that $x^\star$ does not satisfy the second order sufficient conditions either.

Singular and non-singular local minimum

- Local minimum point that does not satisfy the sufficiency condition $\nabla f(x^\star) = 0$, $\nabla f(x^\star) > 0$ is called <u>singular</u> otherwise it is called <u>nonsingular</u>.

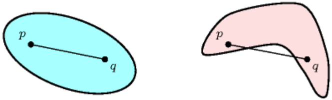
  Singular local minima are harder to deal with

  - In the absence of convexity of f, their optimality cannot be ascertained using easily verifiable sufficient conditions
  - In their neighborhood, the behavior of most commonly used optimization algorithms tends to be slow and /or erratic

Convex sets and convex functions (see Appendix B of Ref[1])

- Convex set Ω: The line connecting any point $p, q \in \Omega$ belongs to Ω:

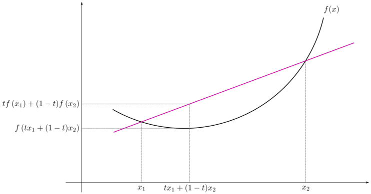
$$\forall p, q \in C : \quad (t\,p + (1-t)\,q) \in \Omega \text{ for } t \in [0, 1].$$



(A) Convex set (B) Non-convex set

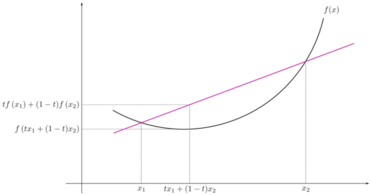
- Convex function: f is convex over convex set Ω iff

$$f(t\,x_1 + (1-t)\,x_2) \leqslant t\,f(x_1) + (1-t)\,f(x_2), \quad \forall x_1, x_2 \in \Omega \text{ for } t \in [0, 1].$$



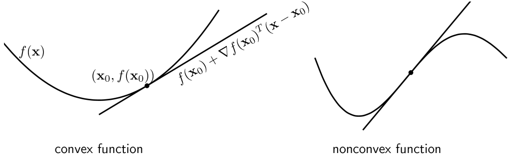
- Convex function: f is convex over convex set Ω iff

$$f(t\,x_1 + (1-t)\,x_2) \leqslant t\,f(x_1) + (1-t)\,f(x_2), \quad \forall x_1, x_2 \in \Omega \text{ for } t \in [0,1].$$



- When f is differentiable, it is convex over convex set Ω iff

$$f(x) \geqslant f(x_0) + \nabla f(x_0)(x - x_0), \quad \forall x_0, x \in \Omega.$$



- When f is twice differentiable, it is convex over convex set Ω iff

$$\nabla^2 f(x) \geqslant 0, \quad \forall x_0, x \in \Omega.$$

Optimality conditions for convex functions

Proposition (Optimality conditions for convex functions)

Let $f : X \to \mathbb{R}$ be a convex function over the convex set X.

(a) A local minimum of f over X is also a global minimum over X. If in addition f is strictly convex, then there exists at most one global minimum of f.

(b) If f is convex and the set X is open, then $\nabla f(\mathbf{x}^\star) = 0$ is a necessary and sufficient condition for a vector $\mathbf{x} \in X$ to be a global minimum of f over X.

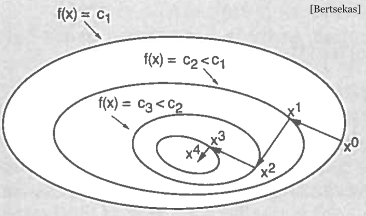
Proof: see page 14 of Ref[1]

- for part (a) use $f(\alpha \mathbf{x}^\star + (1-\alpha)\bar{\mathbf{x}}) \leqslant \alpha f(\mathbf{x}^\star) + (1-\alpha) f(\bar{\mathbf{x}})$
- for part (b) use $f(\mathbf{x}) \geqslant f(\mathbf{x}^\star) + \nabla f(\mathbf{x}^\star)^\top (\mathbf{x} - \mathbf{x}^\star)$, $\forall x \in X$.

Iterative descent methods

- start from $x_0 \in \mathbb{R}^n$ (initial guess)
- successively generate vectors $x_1, x_2, \cdots$ such that

$$f(x_{k+1}) < f(x_k), \qquad k = 0, 1, 2, \cdots$$



$$x_{k+1} = x_k + \alpha_k \, d_k$$

Design factors in iterative descent algorithms:

- what direction to move: descent direction
- how far move in that direction: step size

Successive descent method

$$x_{k+1} = x_k + \alpha_k \, d_k$$

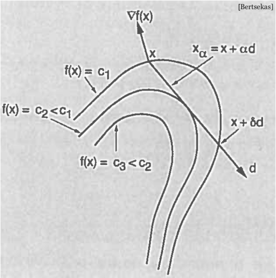
1st order Taylor series : $f(x_{k+1}) = f(x_k + \alpha_k \, d_k) \approx f(x_k) + \alpha_k \nabla f(x_k)^\top \, d_k$

for successive reduction: $\alpha_k \nabla f(x_k)^\top \, d_k < 0$

If $\nabla f(x_k) \neq 0$

- $90° < \angle(d_k, \nabla f(x_k)) < 270°$: $\nabla f(x_k)^\top \, d < 0$
- by appropriate choice of step size α_k we can achieve $f(x_{k+1}) < f(x_k)$

Observations above lead to a set of gradient based algorithms



$$x_{k+1} = x_k + \alpha_k \, d_k$$

$$\text{1st order Taylor series : } f(x_{k+1}) = f(x_k + \alpha_k \, d_k) \approx f(x_k) + \alpha_k \nabla f(x_k)^\top \, d_k$$

$$\text{for successive reduction: } \; \alpha_k \nabla f(x_k)^\top \, d_k < 0$$

$$d_k = -\nabla f(x_k) : \quad -\nabla f(x_k)^\top \nabla f(x_k) < 0, \quad \nabla f(x_k) \neq 0$$

Proposition $d_k = -\nabla f(x_k)$ is a descent direction, i.e., $f(x_k + \alpha_k d_k) < f(x_k)$ for all sufficiently small values of $\alpha_k > 0$.

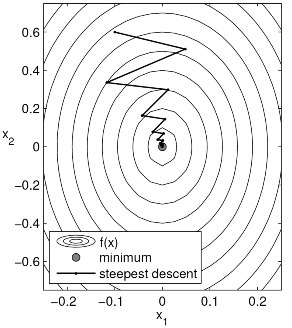
Steepest Descent Algorithm

- **Step 0**. Given x_0, set $k := 0$
- **Step 1**. $d_k := -\nabla f(x_k)$. If $d_k = 0$, then stop.
- **Step 2**. Solve $\alpha_k = \underset{\alpha}{\operatorname{argmin}} f(x_k + \alpha d_k)$ for the stepsize α_k (chosen by an exact or inexact linesearch)
- **Step 3**. Set $x_{k+1} \leftarrow x_k + \alpha_k d_k$, $k \leftarrow k + 1$. Go to **Step 1**.

Note: from Step 2 and the fact that $d_k = -\nabla_k f(x_k)$ is a descent direction it follows that $f(x_{k+1}) < f(x_k)$.

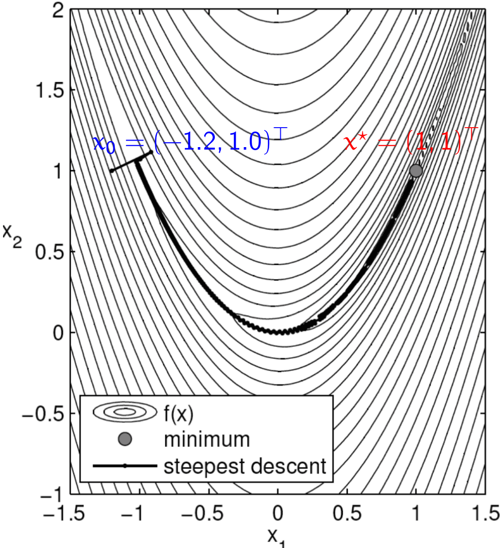
- Steepest descent method can have slow convergence

$f(x_1, x_2) = 1 - e^{-(10x_1^2 + x_2^2)}$



Rosenbrock function:

$f(x_1, x_2) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2$



Newton's method

$$x_{k+1} = x_k + \underbrace{\alpha_k\, d_k}_{\Delta x_k}$$

2nd order Taylor series:

$$f(x_{k+1}) = f(x_k + \Delta x_k) \approx h(\Delta x_k) = f(x_k) + \nabla f(x_k)^\top \Delta x_k + \frac{1}{2}\Delta x_k^\top \nabla^2 f(x_k)\Delta x_k$$

For successive reduction: find the Δx_k from $\underset{\Delta x_k}{\text{minimize}}\ h(\Delta x_k)$

$$\nabla h(\Delta x) = 0 \Rightarrow \ \nabla^2 f(x_k)\Delta x_k + \nabla f(x_k) = 0 \Rightarrow \Delta x_k = -(\nabla^2 f(x_k))^{-1}\nabla f(x_k)$$

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

Newton's method

- **Step 0**. Given x_0, set $k := 0$
- **Step 1**. $d_k := -(\nabla^2 f(x_k))^{-1}\nabla f(x_k)$. If $d_k = 0$, then stop.
- **Step 2**. Solver $\alpha_k = 1$
- **Step 3**. Set $x_{k+1} \leftarrow x_k + \alpha_k d_k$, $k \leftarrow k + 1$. Go to **Step 1**.

2nd order Taylor series:

$$f(x_{k+1}) = f(x_k + \Delta x_k) \approx h(\Delta x_k) = f(x_k) + \nabla f(x_k)^\top \Delta x_k + \Delta x_k^\top \nabla^2 f(x_k) \Delta x_k$$

$$x_{k+1} = x_k - (\nabla^2 f(x_k))^{-1} \nabla f(x_k),$$

Note the following:

- $f(x_{k+1}) < f(x_k)$ is not necessarily guaranteed
- Algorithm can be modified to be $x_{k+1} = x_k - \alpha_k (\nabla^2 f(x_k))^{-1} \nabla f(x_k)$,
- Step 2 the should be modified to be
    - **Step 2**. Solve $\alpha_k = \underset{\alpha}{\operatorname{argmin}} f(x_k - \alpha (\nabla^2 f(x_k))^{-1} \nabla f(x_k))$ for the stepsize α_k (chosen by an exact or inexact linesearch)

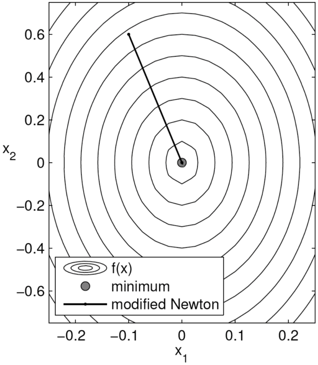
Proposition If $H(x_k) = \nabla^2 f(x_k)$ is a symmetric positive definite matrix, then $d_k := -H(x)^{-1} \nabla f(x_k))$ is a descent direction, i.e., $f(x_k + \alpha_k d_k) < f(x_k)$ for all sufficiently small values of $\alpha_k > 0$.

proof: for d_k to be a descent direction we should show that $\nabla f(x_k)^\top d_k < 0$. here: $\nabla f(x_k)^\top d_k = -\nabla f(x_k)^\top H(x)^{-1} \nabla f(x_k)$. Because $H(x_k)$ is positive definite, it follows that $\nabla f(x_k)^\top d_k = -\nabla f(x_k)^\top H(x)^{-1} \nabla f(x_k) < 0$. Here we used the fact that if a matrix is positive definite, its inverse is also positive definite

Newton and modified Newton methods

- Newton method typically converges very fast asymptotically
- Does not exhibit the zig-zagging behavior of the steepest descent
- on the down side: Newton's method needs to compute not only the gradient, but also the Hessian, which contains $n(n+1)/2$ second order derivatives (numerically expensive).

Example: $f(x_1, x_2) = 1 - e^{-(10x_1^2 + x_2^2)}$



Practical Stopping Conditions for Iterative Optimization Algorithms for Unconstrained Optimization

In iterative algorithms typically the initial point is picked randomly, or if we have a guess for the location of local minima, we pick close to them.

Stopping Criteria: The stoping condition is related to the first order optimality condition of $\nabla f(x) = 0$. The followings are common practical stopping conditions for iterative unconstrained optimization algorithms. Let $\epsilon > 0$:

- $\|f(x_k)\| \leqslant \epsilon$
  - close to satisfying first order necessary condition $\nabla f(x) = 0$.
- $|f(x_{k+1}) - f(x_k)| \leqslant \epsilon$
  - Improvements in function value are saturating.
- $\|x_{k+1} - x_k\| \leqslant \epsilon$
  - Movement between iterates has become small.
- $\frac{|f(x_{k+1}) - f(x_k)|}{\max\{1, |f(x_k)|\}} \leqslant \epsilon$
  - A "relative" measure -removes dependence on the scale of f.
  - The max is taken to avoid dividing by small numbers.
- $\frac{\|x_{k+1} - x_k\|}{\max\{1, \|x_k\|\}} \leqslant \epsilon$
  - A "relative" measure -removes dependence on the scale of $x(k)$
  - The max is taken to avoid dividing by small numbers.

References

[1] Nonlinear Programming: 3rd Edition, by D. P. Bertsekas

[2] Linear and Nonlinear Programming, by D. G. Luenberger, Y. Ye