6. Unconstrained optimization

- unconstrained minimization
- descent methods
- gradient descent method
- Newton method for unconstrained minimization

Unconstrained minimization

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

- $x \in \mathbb{R}^n$ is the optimization or decision *variable*
- $f : \mathbb{R}^n \to \mathbb{R}$ is the *objective function*
- f is assumed to be continuously differentiable (with open domain)
- we assume $x \in \operatorname{dom} f$ whenever $\operatorname{dom} f \neq \mathbb{R}^n$

Solution: $x^\star$ is a *minimizer (minimum point)* or *solution* of f if

$$f(x^\star) \leq f(x) \quad \text{for all } x \in \mathbb{R}^n$$

Optimal value and local minimizer

Optimal value: greatest ρ such that $\rho \leq f(x)$, denoted by $p^\star$

- if $x^\star$ is a minimizer of f, then $p^\star = f(x^\star)$ and optimal value is attained at $x^\star$
- if $p^\star = -\infty$, then we say that the function is unbounded below
- the optimal value is unique even though there could be multiple solutions

Local minimizer

- the minimizer $x^\star$ of f is also called a *global minimizer* of f
- x° is a *local minimizer* or *local minimum point* if there exists $r > 0$ such that

$$f(x^\circ) \leq f(x) \quad \text{for all} \quad \|x - x^\circ\| \leq r$$

- it is a *strict local minimizer* if $f(x^\circ) < f(x)$

First-order optimality condition

if the n-vector x° is a local minimizer of $f : \mathbb{R}^n \to \mathbb{R}$, then

$$\nabla f(x^\circ) = 0 \qquad \left(\frac{\partial f}{\partial x_i}(x^\circ) = 0, \quad i = 1, \ldots, n\right)$$

- reduces to $f'(x) = 0$ for single-variable case $n = 1$
- this condition is *necessary* but not sufficient
- points that satisfies $\nabla f(\hat{x}) = 0$ are called *stationary points* or *critical points*
- stationary points can be minimizers, maximizers, or neither (saddle points)
- minimizing $f(x)$ is the same as solving a nonlinear equation $h(x) = \nabla f(x) = 0$
- often difficult to solve and numerical algorithms are used

Intuition and proof for single-variable case

Intuition

- $f'(x) > 0$ implies f is increasing, so $\tilde{x}$ slightly less than x gives $f(\tilde{x}) < f(x)$
- $f'(x) < 0$ means f is decreasing, so $\tilde{x}$ slightly more than x gives $f(\tilde{x}) < f(x)$
- this means that x is not a minimizer of f

Proof

- if x° is a local minimizer, then $f(x^\circ) \le f(x^\circ + \epsilon)$ for sufficiently small ϵ
- when $\epsilon > 0$, the limit from the right is

$$f'(x^\circ) = \lim_{\epsilon \to 0^+} \frac{f(x^\circ + \epsilon) - f(x^\circ)}{\epsilon} \ge 0$$

- when $\epsilon < 0$, the limit from the left is

$$f'(x^\circ) = \lim_{\epsilon \to 0^-} \frac{f(x^\circ + \epsilon) - f(x^\circ)}{\epsilon} \le 0$$

- hence, $0 \le f'(x^\circ) \le 0 \Rightarrow f'(x^\circ) = 0$

Example

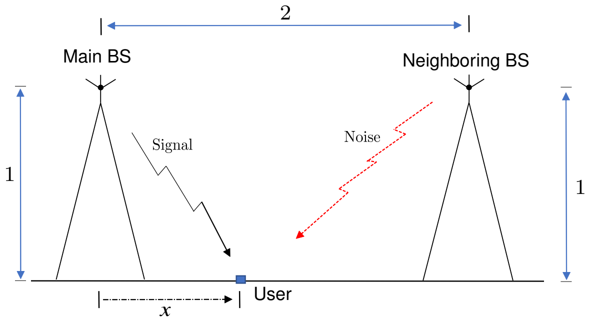
$$f(x) = 3x^4 - 20x^3 + 42x^2 - 36x$$

the optimality condition is

$$f'(x) = 12x^3 - 60x^2 + 84x - 36 = 12(x-1)^2(x-3) = 0$$

- the stationary points are $x = 1$ and $x = 3$
- $x = 1$ is not a local optima because $f'(x)$ does not change sign around $x = 1$
- $x = 3$ is a local minimizer since $f'(x)$ change from -ve to +ve around $x = 3$
- since $f(x) \to \infty$ as $|x| \to \infty$, the point $x = 3$ must be a global minimizer

Example



- power of the received signal measured by the user from each antenna is the reciprocal of the squared distance from the corresponding antenna
- find position x of user (relative to main station) that maximizes signal-to-noise ratio

to solve this problem, we need to maximize the signal-to-noise ratio:

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

setting the derivative to zero:

$$f'(x) = \frac{-2(2-x)(1+x^2) - 2x(1+(2-x)^2)}{(1+x^2)^2} = \frac{4(x^2 - 2x - 1)}{(1+x^2)^2} = 0$$

- $f'(x) = 0$ at $x = 1 \pm \sqrt{2}$
- $x = 1 - \sqrt{2}$ gives larger objective ($f(1 - \sqrt{2}) \approx 5.828$)
- derivative changes its sign from +ve to -ve when passing through $x = 1 - \sqrt{2}$
- hence, $x^\circ = 1 - \sqrt{2}$ is a local maximizer
- it is a global maximizer since $f(x) \to 1 < f(x^\circ)$ as $|x| \to \infty$

Example

let us find the stationary points of

$$f(x) = x_1^3 - x_1^2 x_2 + 2x_2^2$$

- we set the gradient (partial derivatives) to zero to obtain optimality condition:

$$\frac{\partial f}{\partial x_1} = 3x_1^2 - 2x_1 x_2 = 0$$
$$\frac{\partial f}{\partial x_2} = -x_1^2 + 4x_2 = 0$$

- solving, we get two stationary points: $(0, 0)$ and $(6, 9)$

Deriving second-order conditions

- if $x^\star$ is a local minimum, then for any direction v we have

$$f(x^\star + v) = f(x^\star) + \nabla f(x^\star)^T v + (1/2) v^T \nabla^2 f(x^\star) v \geq f(x^\star)$$

- for a very small $\|v\|$, if $\nabla f(x^\star) \neq 0$, then we can find v such that $\nabla f(x^\star)^T v < 0$

- so we must have $\nabla f(x^\star) = 0$ at a minimum

- at a strict minimum we must also have for all v satisfying $0 < \|v\| \ll 1$

$$f(x^\star + v) = f(x^\star) + (1/2) v^T \nabla^2 f(x^\star) v > f(x^\star)$$

  this will happen if the Hessian matrix $\nabla^2 f(x^\star)$ is positive definite

- this implies that at a local minimizer, the function has an 'upward' curvature

Second-order optimality condition

Necessary condition: if x° is a local minimizer, then

$$\nabla f(x^\circ) = 0 \quad \text{and} \quad \nabla^2 f(x^\circ) \succeq 0$$

Sufficient condition: if x° satisfies

$$\nabla f(x^\circ) = 0 \quad \text{and} \quad \nabla^2 f(x^\circ) \succ 0$$

then x° is a (strict) local minimizer

Necessary and sufficient condition

- f is convex if $\nabla^2 f(x) \succeq 0$ for all x (positive semidefinite everywhere)
- for convex f, $x^\star$ is global minimizer if and only if $\nabla f(x^\star) = 0$

(we can find maximizers by finding minimizers of $-f$)

Example

a minimizer of $f(x) = e^x + e^{-x} - 3x^2$ must satisfy

$$f'(x) = e^x - e^{-x} - 6x = 0$$

- solving gives $\hat{x}_1 \approx 2.84$ and $\hat{x}_2 \approx -2.84$, and $\hat{x}_3 = 0$
- to find whether these points are local minimizer, we compute the second derivative

$$f''(x) = e^x + e^{-x} - 6$$

- $f''(2.84) > 0$, $f''(-2.84) > 0$, $f''(0) < 0$, so $\hat{x}_1$ and $\hat{x}_2$ are local minimizers
- checking the value of the functions, we see that $f(2.84) = f(-2.84)$; these two points are global minimizers since $f(x) \to \infty$ as $|x| \to \infty$

Examples

- for $f(x) = x^3$, we have

$$f'(x) = 3x^2 = 0 \Rightarrow \hat{x} = 0$$

  $f''(0) = 0$, but $\hat{x} = 0$ is not a local minimizer since $f(x) < f(0)$ for $x < 0$

  (condition $f''(x) \geq 0$ is not enough to characterize local minimizers)

- the first and second derivative of $f(x) = \log(e^x + e^{-x})$ are

$$f'(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}, \quad f''(x) = \frac{4}{(e^x + e^{-x})^2}$$

  unique stationary point $\hat{x} = 0$

  since $f''(x) > 0$ for all x, $\hat{x} = 0$ is a global minimizer

Example

$$f(x) = x_1^3 - x_1^2 x_2 + 2x_2^2$$

the stationary points are $(0,0)$ and $(6,9)$ (see page 6.9)

the Hessian is

$$\nabla^2 f(x) = \begin{bmatrix} 6x_1 - 2x_2 & -2x_1 \\ -2x_1 & 4 \end{bmatrix}$$

hence,

$$\nabla^2 f(0,0) = \begin{bmatrix} 0 & 0 \\ 0 & 4 \end{bmatrix}, \quad \nabla^2 f(6,9) = \begin{bmatrix} 18 & -12 \\ -12 & 4 \end{bmatrix}$$

- $\nabla^2 f(0,0) \succeq 0$, so it is still unclear whether $(0,0)$ is a local minimizer
- $\nabla^2 f(6,9)$ is indefinite, so $(6,9)$ is not a local minimizer/maximizer
- since $f(\epsilon,0) > 0$ for any $\epsilon > 0$ and $f(\epsilon,0) < 0$ for any $\epsilon < 0$, we conclude that the point $(0,0)$ is not a local minimizer/maximizer

Example

for $f(x) = \frac{1}{2}x_1^2 + x_1x_2 + 2x_2^2 - 4x_1 - 4x_2 - x_2^3$, the optimality condition is

$$\nabla f(x) = \begin{bmatrix} x_1 + x_2 - 4 \\ x_1 + 4x_2 - 4 - 3x_2^2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

solving, we get the stationary points $(4, 0)$ and $(3, 1)$; the Hessian is

$$\nabla^2 f(x) = \begin{bmatrix} 1 & 1 \\ 1 & 4 - 6x_2 \end{bmatrix}$$

thus,

$$\nabla^2 f(4, 0) = \begin{bmatrix} 1 & 1 \\ 1 & 4 \end{bmatrix}, \quad \nabla^2 f(3, 1) = \begin{bmatrix} 1 & 1 \\ 1 & -2 \end{bmatrix}$$

- $\nabla^2 f(4, 0) \succeq 0$ so $\hat{x} = (4, 0)$ is a local minimizer
- $\nabla^2 f(3, 1)$ is indefinite so $(3, 1)$ is not a minimizer/maximizer
- note that $\hat{x} = (4, 0)$ is not a global minimizer since $f(0, x_2) \to -\infty$ as $x_2 \to \infty$

Example

for

$$f(x) = x_1^2 - x_1 x_2 + x_2^2 - 3x_2$$

the optimality condition is

$$\nabla f(x) = \begin{bmatrix} 2x_1 - x_2 \\ -x_1 + 2x_2 - 3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

- has a unique solution $\hat{x}_1 = 1, \hat{x}_2 = 2$
- since the Hessian

$$\nabla^2 f(x) = \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix}$$

  is positive definite everywhere, the point $\hat{x} = (1, 2)$ is a global minimizer

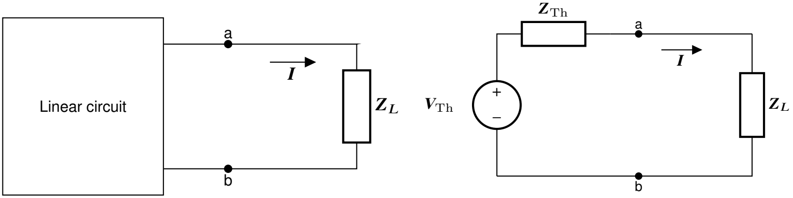
Quadratic functions

$$f(x) = \tfrac{1}{2}x^T Q x + r^T x + s, \qquad Q \in \mathbb{S}^n$$

Optimality condition: $\nabla f(x) = Qx + r = 0$ with Hessian $\nabla^2 f(x) = Q$

- if $Q \succeq 0$, then $x^\star$ is a global minimizer iff $Qx^\star + r = 0$
  - if $Q \succ 0$, then there is a unique minimizer $x^\star = -Q^{-1}r$
- if Q is singular and $r \in \mathrm{range}(Q)$, then there exists multiple stationary points
- if $r \notin \mathrm{range}(Q)$, then there is no solution and f is unbounded below
- if Q is indefinite, then any stationary point is a saddle-point
- if Q is invertible, then there is a unique stationary point: $\hat{x} = -Q^{-1}r$

Example: maximum power transfer



- V_{Th} is the Thevenin voltage
- $Z_{\mathrm{Th}} = R_{\mathrm{Th}} + \mathrm{j}X_{\mathrm{Th}}$ ($\mathrm{j} = \sqrt{-1}$) is the Thevenin impedance
- $Z_L = R_L + \mathrm{j}X_L$ is the impedance of the load
- find load impedance (*i.e.*, R_L and X_L) such that average power delivered to load

$$P = |I|^2 R_L, \qquad I = \frac{V_{\mathrm{Th}}}{R_{\mathrm{Th}} + R_L + j(X_{\mathrm{Th}} + X_L)}$$

is maximized; (assume $V_{\mathrm{Th}} = 1$ and $R_{\mathrm{Th}} > 0$)

problem is

$$\text{maximize} \quad f(x) = \frac{x_1}{(R_{\mathrm{Th}} + x_1)^2 + (X_{\mathrm{Th}} + x_2)^2}$$

with variables $x_1 = R_L$, $x_2 = X_L$; setting the gradient (partial derivatives) to zero:

$$\nabla_{x_1} f(x) = \frac{\partial f}{\partial x_1} = \frac{(R_{\mathrm{Th}} + x_1)^2 + (X_{\mathrm{Th}} + x_2)^2 - 2x_1(R_{\mathrm{Th}} + x_1)}{\left((R_{\mathrm{Th}} + x_1)^2 + (X_{\mathrm{Th}} + x_2)^2\right)^2} = 0$$

$$\nabla_{x_2} f(x) = \frac{\partial f}{\partial x_2} = \frac{-2x_1(X_{\mathrm{Th}} + x_2)}{\left((R_{\mathrm{Th}} + x_1)^2 + (X_{\mathrm{Th}} + x_2)^2\right)^2} = 0$$

- from 2nd equation, we have $x_1 = 0$ or $x_2 = -X_{\mathrm{Th}}$
- note that $x_1 = 0$ does not satisfy the 1st condition
- plugging $x_2 = -X_{\mathrm{Th}}$ into the 1st condition and simplifying, we get

$$(R_{\mathrm{Th}} + x_1)^2 - 2x_1(R_{\mathrm{Th}} + x_1) = 0 \Longrightarrow x_1 = R_{\mathrm{Th}}$$

- hence, the stationary point is $x = (R_{\mathrm{Th}}, -X_{\mathrm{Th}})$

we now check the second-order conditions

- to simplify derivation of Hessian, let $f(x) = g(Ax + b)$ where

$$g(y_1, y_2, y_3) = \frac{y_1}{y_2^2 + y_3^2}, \quad A = \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad b = \begin{bmatrix} 0 \\ R_{\mathrm{Th}} \\ X_{\mathrm{Th}} \end{bmatrix}$$

- by composition rule, the Hessian of f is $A^T \nabla^2 g(Ax + b) A$

- thus, we need to find the Hessain of h; the gradient of g is

$$\nabla g(y) = \begin{bmatrix} \frac{1}{y_2^2 + y_3^2} \\ \frac{-2y_1 y_2}{(y_2^2 + y_3^2)^2} \\ \frac{-2y_1 y_3}{(y_2^2 + y_3^2)^2} \end{bmatrix}$$

- the Hessian of g is

$$\nabla^2 g(y) = \begin{bmatrix} 0 & \frac{-2y_2}{(y_2^2+y_3^2)^2} & \frac{-2y_3}{(y_2^2+y_3^2)^2} \\ \frac{-2y_2}{(y_2^2+y_3^2)^2} & \frac{-2y_1(y_2^2+y_3^2)+8y_1y_2^2}{(y_2^2+y_3^2)^3} & \frac{8y_1y_2y_3}{(y_2^2+y_3^2)^3} \\ \frac{-2y_3}{(y_2^2+y_3^2)^2} & \frac{8y_1y_2y_3}{(y_2^2+y_3^2)^3} & \frac{-2y_1(y_2^2+y_3^2)+8y_1y_3^2}{(y_2^2+y_3^2)^3} \end{bmatrix}$$

$$= \frac{2}{(y_2^2+y_3^2)^2} \begin{bmatrix} 0 & -y_2 & -y_3 \\ -y_2 & \frac{-y_1(y_2^2+y_3^2)+4y_1y_2^2}{(y_2^2+y_3^2)} & \frac{4y_1y_2y_3}{(y_2^2+y_3^2)} \\ -y_3 & \frac{4y_1y_2y_3}{(y_2^2+y_3^2)} & \frac{-y_1(y_2^2+y_3^2)+4y_1y_3^2}{(y_2^2+y_3^2)} \end{bmatrix}$$

- at $x = (R_{\mathrm{Th}}, -X_{\mathrm{Th}})$, we have

$$Ax + b = \begin{bmatrix} R_{\mathrm{Th}} \\ 2R_{\mathrm{Th}} \\ 0 \end{bmatrix}$$

- hence, at $x = (R_{\rm Th}, -X_{\rm Th})$, we have

$$\nabla^2 g(Ax+b) = \frac{2}{(2R_{\rm Th})^4}\begin{bmatrix} 0 & -2R_{\rm Th} & 0 \\ -2R_{\rm Th} & 3R_{\rm Th} & 0 \\ 0 & 0 & -R_{\rm Th} \end{bmatrix} = \frac{1}{(2R_{\rm Th})^3}\begin{bmatrix} 0 & -2 & 0 \\ -2 & 3 & 0 \\ 0 & 0 & -1 \end{bmatrix}$$

- the Hessian of f at $x = (R_{\rm Th}, -X_{\rm Th})$ is

$$\begin{aligned} \nabla^2 f(x) &= A^T \nabla^2 g(Ax+b) A \\ &= \frac{1}{(2R_{\rm Th})^3}\begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} 0 & -2 & 0 \\ -2 & 3 & 0 \\ 0 & 0 & -1 \end{bmatrix}\begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \\ &= \frac{1}{(2R_{\rm Th})^3}\begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} \end{aligned}$$

- since $R_{\rm Th} > 0$, the Hessian is negative definite and $x = (R_{\rm Th}, -X_{\rm Th})$ is a local maximum; because it is the only point stationary point, it is a global maximum

Outline

- unconstrained minimization
- **descent methods**
- gradient descent method
- Newton method for unconstrained minimization

Descent methods

Descent direction: a vector $v \in \mathbb{R}^n$ is called a *descent direction* for f if

$$f(x + \alpha v) < f(x) \quad \text{for sufficiently small } \alpha > 0$$

---

choose a starting point $x^{(0)}$, a solution tolerance $\epsilon > 0$, and a stopping criteria
repeat for $k \geq 0$

1. determine a decent direction $v^{(k)}$
2. **if** stopping criteria is satisfied, then stop and output $x^{(k)}$
3. select a stepsize α_k
4. update $x^{(k+1)} = x^{(k)} + \alpha_k v^{(k)}$

until maximum number of iterations reached

- v is a descent direction if the *directional derivative* of f at x in the direction v is

$$f'(x; v) = \lim_{\alpha \to 0} \frac{f(x + \alpha v) - f(x)}{\alpha} = \nabla f(x)^T v < 0$$

- $\nabla f(x)^T v$ gives an approximate rate of change (increase) of f in direction v at x

Determining the stepsize

Constant stepsize: set $\alpha_k = \alpha$ for all k

Exact line search

$$\alpha_k = \underset{\alpha \geq 0}{\operatorname{argmin}} \quad f(x^{(k)} + \alpha v^{(k)})$$

it is not always possible to actually find the exact minimizer α

Backtracking line search

- choose $\beta \in (0, 1/2)$, and $\gamma \in (0, 1)$ and initial guess α_k (*e.g.*, $\alpha_k = 1$)
- set $\alpha_k := \beta\alpha_k$ until

$$f(x^{(k)} + \alpha_k v^{(k)}) < f(x^{(k)}) + \gamma\alpha_k \nabla f(x^{(k)})^T v^{(k)}$$

  this method is a compromise between the above two methods

- simple backtracking algorithm is to set

$$\alpha_k = 1, 0.5, 0.5^2, 0.5^3, \ldots$$

  until the above is satisfied or until $f(x^{(k)} + \alpha_k v^{(k)}) < f(x^{(k)})$

Stopping criteria

1. $|f(x^{(k+1)}) - f(x^{(k)})| < \epsilon$
2. $\|x^{(k+1)} - x^{(k)}\| < \epsilon$
3. $|f(x^{(k+1)}) - f(x^{(k)})|/|f(x^{(k)})| < \epsilon$
4. $\|x^{(k+1)} - x^{(k)}\|/\|x^{(k)}\| < \epsilon$
5. $\|\nabla f(x^{(k)})\| < \epsilon$

- the above conditions do not necessarily imply that $x^{(k)}$ is a good solution since it can be a local minimizer/maximizer or a saddle-point (unless f is convex)
- it is common to run the algorithm from different starting points and choose the best solution of these multiple runs

Outline

- unconstrained minimization
- descent methods
- **gradient descent method**
- Newton method for unconstrained minimization

Negative gradient direction

the directional derivative in the direction $v = -\nabla f(x)$ is

$$v^T \nabla f(x) = -\|\nabla f(x)\|^2 < 0 \quad \text{for any } x \text{ with } \nabla f(x) \neq 0$$

thus, $-\nabla f(x)$ is a descent direction

- suppose $\|v\| = 1$, then by Cauchy-Schwarz, we have

$$-\|\nabla f(x)\| \leq \nabla f(x)^T v$$

- equality holds only if $v = -\nabla f(x)/\|\nabla f(x)\|$
- so $-\nabla f(x)$ point in *steepest descent* (maximum rate of decrease) direction at x
- setting $v^{(k)} = -\nabla f(x^{(k)})$ in the descent method gives the *gradient method* or *gradient descent method*

Gradient descent method

---

given a starting point $x^{(0)}$ and a solution tolerance $\epsilon > 0$

repeat for $k \geq 0$

1. **if** $\|\nabla f(x^{(k)})\| \leq \epsilon$ stop and output $x^{(k)}$
2. choose a stepsize α_k
3. update
$$x^{(k+1)} = x^{(k)} - \alpha_k \nabla f(x^{(k)})$$

- for α_k small enough, the algorithm is a descent method
- when α_k is large, the algorithm may not be a descent method and may fail
- called *the method of steepest descent* with exact line search

Example

$$f(x_1, x_2, x_3) = (x_1 - 4)^4 + (x_2 - 3)^2 + 4(x_3 + 5)^4$$

the gradient of this function is

$$\nabla f(x) = \begin{bmatrix} 4(x_1 - 4)^3 \\ 2(x_2 - 3) \\ 16(x_3 + 5)^3 \end{bmatrix}$$

applying one iteration of the gradient descent with $x^{(0)} = (4, 2, -1)$, $\alpha = 0.002$ gives

$$x^{(1)} = \begin{bmatrix} 4 \\ 2 \\ -1 \end{bmatrix} - 0.002 \begin{bmatrix} 4(4-4)^3 \\ 2(2-3) \\ 16(-1+5)^3 \end{bmatrix} = \begin{bmatrix} 4.000 \\ 2.004 \\ -3.048 \end{bmatrix}$$

the new objective value is

$$59.06 = f(4, 2.004, -3.048) < f(4, 2, -1) = 1025,$$

which shows that $\alpha = 0.002$ is a good choice if we use exact line search, then

$$\begin{aligned}\alpha_0 &= \underset{\alpha>0}{\operatorname{argmin}}\, f(x^{(0)} - \alpha\nabla f(x^{(0)})) \\ &= \underset{\alpha>0}{\operatorname{argmin}}(0 + (2 + 2\alpha - 3)^2 + 4(-1 - 1024\alpha + 5)^4) \\ &= 3.967 \times 10^{-3}\end{aligned}$$

hence,

$$x^{(1)} = x^{(0)} - \alpha_0\nabla f(x^{(0)}) = (4.000, 2.008, -5.062)$$

Example

$$f(x_1, x_2) = \frac{x_1^2}{5} + x_2^2$$

- the gradient is $\nabla f(x) = (\frac{2}{5}x_1, 2x_2)$
- we have
$$f(x - \alpha \nabla f(x)) = \tfrac{1}{5}(x_1 - \tfrac{2}{5}\alpha x_1)^2 + (x_2 - 2\alpha x_2)^2$$
- using exact line search in the gradient method, we have
$$\begin{aligned}\alpha &= \underset{\alpha>0}{\operatorname{argmin}}\, f(x - \alpha \nabla f(x)) \\ &= \underset{\alpha>0}{\operatorname{argmin}} \left(\tfrac{1}{5}(x_1 - \tfrac{2}{5}\alpha x_1)^2 + (x_2 - 2\alpha x_2)^2\right)\end{aligned}$$

- setting the derivative with respect to α to zero, we get

$$-\tfrac{4}{25}x_1(x_1 - \tfrac{2}{5}\alpha x_1) - 4x_2(x_2 - 2\alpha x_2) = 0$$

- solving for α, gives

$$\alpha = \frac{\frac{4}{25}x_1^2 + 4x_2^2}{\frac{8}{125}x_1^2 + 8x_2^2} > 0$$

- hence, the method of steepest descent is

$$\begin{bmatrix} x_1^{(k+1)} \\ x_2^{(k+1)} \end{bmatrix} = \begin{bmatrix} x_1^{(k)} \\ x_2^{(k)} \end{bmatrix} - \frac{\frac{4}{25}(x_1^{(k)})^2 + 4(x_2^{(k)})^2}{\frac{8}{125}(x_1^{(k)})^2 + 8(x_2^{(k)})^2} \begin{bmatrix} \frac{2}{5}x_1^{(k)} \\ 2x_2^{(k)} \end{bmatrix}$$

Exact line search for quadratic functions

$$f(x) = \frac{1}{2}x^TQx - r^Tx$$

- $Q \in \mathbb{S}^n_{++}$ is positive definite
- gradient method with exact line search requires solving:

$$\alpha_k = \operatorname*{argmin}_{\alpha>0} f(x^{(k)} + \alpha v^{(k)})$$

where $v^{(k)} = -\nabla f(x^{(k)}) = -(Qx^{(k)} - r)$

Update form

$$x^{(k+1)} = x^{(k)} - \frac{\|\nabla f(x^{(k)})\|^2}{\nabla f(x^{(k)})^T Q \nabla f(x^{(k)})} \nabla f(x^{(k)})$$

Derivation

- let $v = v^{(k)} = -\nabla f(x^{(k)}) = -(Qx^{(k)} - r)$
- using the chain rule, we have

$$\begin{aligned} g'(\alpha) &= v^T \nabla f(x^{(k)} + \alpha v) \\ &= v^T \big(Q(x^{(k)} + \alpha v) - r\big) \\ &= \alpha v^T Q v + v^T (Qx^{(k)} - r) \\ &= \alpha v^T Q v - v^T v \end{aligned}$$

- setting to zero and solving for α, we get

$$\alpha_k = \frac{v^T v}{v^T Q v}$$

Convergence

under mild assumptions, $\{x^{(k)}\}$ of gradient method converge to a stationary point:

$$\lim_{k\to\infty} \nabla f(x^{(k)}) = 0$$

- converges to a global minimizer for convex f (*e.g.*, $\nabla^2 f(x) \succeq 0$ for all x)
- the rate of convergence is sublinear (slow) in general and linear if $\mu I \preceq \nabla^2 f(x)$ for all x and some constant $\mu > 0$

Outline

- unconstrained minimization
- descent methods
- gradient descent method
- **Newton method for unconstrained minimization**

Newton method

consider n nonlinear equations in n variables

$$h_1(x) = 0, \quad h_2(x) = 0, \quad \ldots, \quad h_n(x) = 0$$

where $x = (x_1, \ldots, x_n) \in \mathbb{R}^n$; we let $h(x) = (h_1(x), \ldots, h_n(x))$

Newton method: choose $x^{(0)}$ and repeat for $k \geq 0$

$$x^{(k+1)} = x^{(k)} - Dh(x^{(k)})^{-1} h(x^{(k)})$$

assumes $Dh(x^{(k)})$ exists and nonsingular

Unconstrained optimization: if $h(x) = \nabla f(x)$, we get

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

Interpretation of Newton update

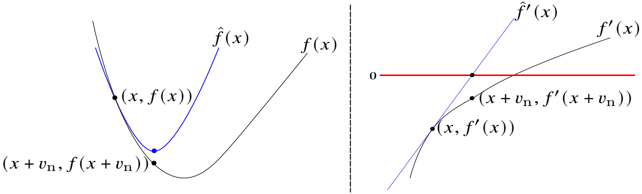
$$x = x^{(k)} - \nabla^2 f(x^{(k)})^{-1} \nabla f(x^{(k)})$$

1. minimizing the quadratic approximation of f around $x^{(k)}$:

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

2. solve approximate optimality condition around $x^{(k)}$:

$$\widehat{\nabla f}(x) = \nabla f(x^{(k)}) + \nabla^2 f(x^{(k)})(x - x^{(k)}) = 0$$



Damped Newton method

given a starting point $x^{(0)}$, a solution tolerance $\epsilon > 0$

repeat for $k \geq 0$

1. **if** stopping criteria is met (*e.g.*, $\|\nabla f(x^{(k)})\| \leq \epsilon$), stop and return $x^{(k)}$
2. select a step-size α_k
3. solve $\nabla^2 f(x^{(k)})v^{(k)} = -\nabla f(x^{(k)})$ for $v^{(k)}$
4. update:

$$x^{(k+1)} = x^{(k)} + \alpha_k v^{(k)}$$

- assumes $\nabla^2 f(x)$ exists and is invertible
- $v_{\mathrm{n}} = -\nabla^2 f(x^{(k)})^{-1}\nabla f(x^{(k)})$ is called *Newton step* at $x^{(k)}$
- similar stepsize selection and stopping criteria as before can be used
- single-variable update

$$x^{(k+1)} = x^{(k)} - \alpha_k \frac{f'(x^{(k)})}{f''(x^{(k)})}$$

Example

$$\text{minimize} \quad f(x) = \tfrac{1}{2}x^2 - \sin x$$

given $x^{(0)} = 0.5$, $\alpha = 1$, $\epsilon = 10^{-5}$ with stopping criteria $|x^{(k+1)} - x^{(k)}| < \epsilon$

- applying Newton's method, we have

$$\begin{aligned} x^{(1)} = x^{(0)} - \frac{f'(x^{(0)})}{f''(x^{(0)})} &= 0.5 - \frac{0.5 - \cos(0.5)}{1 + \sin(0.5)} \\ &= 0.5 - \frac{-0.3775}{1.479} = 0.7552 \end{aligned}$$

  repeating, we get $x^{(2)} = 0.7391$, $x^{(3)} = 0.7390$, and $x^{(4)} \approx 0.7390$

- note that $|x^{(4)} - x^{(3)}| < \epsilon$, $f'(x^{(4)}) \approx 0$, and $f''(x^{(4)}) = 1.672 > 0$
- hence, $x^{(4)}$ is an approximate local minimizer (it is an approximate global minima)

Example

$$f(x) = e^{x_1+x_2-1} + e^{x_1-x_2-1} + e^{-x_1-1}$$

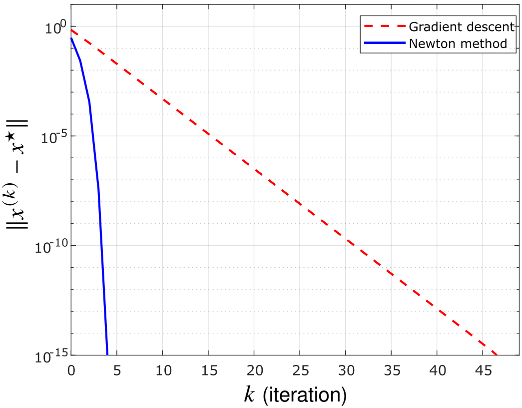
the gradient and Hessian are

$$\nabla f(x) = \begin{bmatrix} e^{x_1+x_2-1} + e^{x_1-x_2-1} - e^{-x_1-1} \\ e^{x_1+x_2-1} - e^{x_1-x_2-1} \end{bmatrix}$$

and

$$\nabla^2 f(x) = \begin{bmatrix} e^{x_1+x_2-1} + e^{x_1-x_2-1} + e^{-x_1-1} & e^{x_1+x_2-1} - e^{x_1-x_2-1} \\ e^{x_1+x_2-1} - e^{x_1-x_2-1} & e^{x_1+x_2-1} + e^{x_1-x_2-1} \end{bmatrix}$$

we apply gradient descent and Newton method with $x^{(0)} = (-1, 1)$ and $\alpha = 1$



- both algorithms converge to $x^\star = (-0.34657, 0)$
- Newton method is much faster since it uses second-order information

Matlab implementation

```
 g=@(x)[exp(x(1)+x(2)-1)+exp(x(1)-x(2)-1)-exp(-x(1)-1);...
exp(x(1)+x(2)-1)-exp(x(1)-x(2)-1)]; % gradient
hess=@(x)[exp(x(1)+x(2)-1)+exp(x(1)-x(2)-1)+exp(-x(1)-1) ...
exp(x(1)+x(2)-1)-exp(x(1)-x(2)-1);...
exp(x(1)+x(2)-1)-exp(x(1)-x(2)-1) ...
exp(x(1)+x(2)-1)+exp(x(1)-x(2)-1)] % hessain
%% Newton and GD iterations
x = [-1; 1];%GD initilization
xn = [-1; 1];%Newton initilization
alpha=1; %step-size
for k=1:50
%%%Gradient descent update%%%%
grad=g(x);
if (norm(grad) < 1e-16), break; end;
x = x - alpha*grad;
%%%Newton update%%%
vn=-hess(xn)\g(xn);
xn = xn + alpha*vn;
end
```

Alternative way to construct gradient and Hessian

$$f(x) = e^{x_1+x_2-1} + e^{x_1-x_2-1} + e^{-x_1-1}$$

we can write f as $f(x) = g(Ax + b)$, where $g(y) = e^{y_1} + e^{y_2} + e^{y_3}$, and

$$A = \begin{bmatrix} 1 & 1 \\ 1 & -1 \\ -1 & 0 \end{bmatrix}, \quad b = \begin{bmatrix} -1 \\ -1 \\ -1 \end{bmatrix}$$

the gradient and Hessian of g are

$$\nabla g(y) = \begin{bmatrix} e^{y_1} \\ e^{y_2} \\ e^{y_3} \end{bmatrix}, \quad \nabla^2 g(y) = \begin{bmatrix} e^{y_1} & 0 & 0 \\ 0 & e^{y_2} & 0 \\ 0 & 0 & e^{y_3} \end{bmatrix}$$

it follows that

$$\nabla f(x) = A^T \nabla g(Ax + b)$$
$$\nabla^2 f(x) = A^T \nabla^2 g(Ax + b) A$$

Matlab implementation

```
 A=[1 1;1 -1;-1 0];
b=[1;1;1];

for k=1:50
%%% Gradient descent update %%%
y=exp(A*x-b);
grad=A'*y;
if (norm(grad) < 1e-16), break; end;
x = x - alpha*grad;
%%% Newton's update %%%
yn=exp(A*xn-b);
gradn=A'*yn;
D = diag(yn);
H=A'*D*A;
vn=-H\gradn;
xn = xn + alpha*vn;
end;
```

Example

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

- $a_i \in \mathbb{R}^n$ and $b_i \in \mathbb{R}$ are the problem data
- m and n can be very large
- suppose that we want to solve this problem using Newton's method with
  - initialization $x^{(0)} = \mathbf{1}$
  - stopping criteria $\|\nabla f(x^{(k)})\| < 10^{-5}$
  - line search parameters: $\alpha_0 = 1$, $\beta = 1/2$, and $\gamma = 0.01$
- for implementation, we first need to find the gradient and Hessian of the function f

the function f can be written as

$$f(x) = g(Ax - b) \quad \text{where} \quad g(y) = \sum_{i=1}^{m} \log(e^{y_i} + e^{-y_i})$$

and

$$A = \begin{bmatrix} a_1^T \\ \vdots \\ a_m^T \end{bmatrix}, \quad b = \begin{bmatrix} b_1 \\ \vdots \\ b_m \end{bmatrix}$$

the gradient and Hessian of h are:

$$\nabla g(y) = \begin{bmatrix} (e^{y_1} - e^{-y_1})/(e^{y_1} + e^{-y_1}) \\ \vdots \\ (e^{y_m} - e^{-y_m})/(e^{y_m} + e^{-y_m}) \end{bmatrix}$$

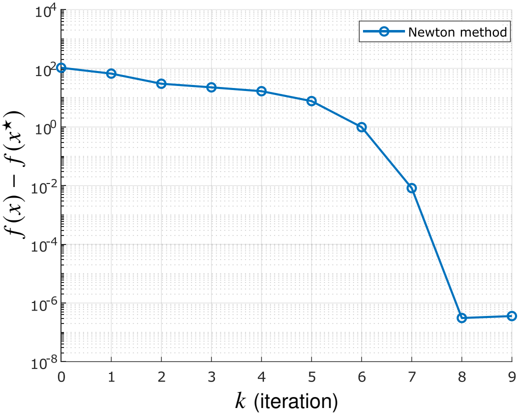
$$\nabla^2 g(y) = \operatorname{diag}\bigl(4/(e^{y_1} + e^{-y_1})^2, \ldots, 4/(e^{y_m} + e^{-y_m})^2\bigr)$$

using the composition with affine function property, we have

$$\nabla f(x) = A^T \nabla g(Ax - b), \quad \nabla^2 f(x) = A^T \nabla^2 g(Ax - b) A$$

MATLAB code

```
alpha_0=1;
beta=0.5;
gamma=0.01;
x = ones(n,1); %initialization
k=1;
y = A*x-b;
grad = A'*((exp(y)-exp(-y))./(exp(y)+exp(-y)));
while (norm(grad) >= 1e-5)
k=k+1; %iteration counter
hess = 4*A'*diag(1./(exp(y)+exp(-y)).^2)*A;
d = -hess\grad;
alpha = alpha_0;
f = sum(log(exp(y)+exp(-y)));
while (sum(log(exp(A*(x+alpha*d)-b)+exp(-A*(x+alpha*d)+b))) ...
> f + gamma*alpha*grad'*d)
alpha = beta*alpha;
end
x = x+alpha*d;
y = A*x-b;
f = sum(log(exp(y)+exp(-y)));
grad = A'*((exp(y)-exp(-y))./(exp(y)+exp(-y)));
end
```



10^4
10^2
10^0
10^{-2}
10^{-4}
10^{-6}
10^{-8}
$f(x) - f(x^\star)$
0
1
2
3
4
5
6
7
8
9
k (iteration)
Newton method

Convergence

quadratic convergence near the optimal solution

$$\|x^{(k+1)} - x^\star\| \le c\|x^{(k)} - x^\star\|^2 \quad \text{for some positive } c > 0$$

- if $\nabla^2 f(x) \succ 0$ (convex) then $v_{\rm n} = -\nabla^2 f(x^{(k)})^{-1}\nabla f(x^{(k)})$ is a descent direction; converges quadratically to a global minimizer under certain conditions
- may not work well when $\nabla^2 f(x)$ is not positive definite
  - in this case, Newton step is not always a descent direction
- can use hybrid gradient-Newton method by setting

$$v^{(k)} = \begin{cases} -\nabla^2 f(x^{(k)})^{-1}\nabla f(x^{(k)}) & \text{if} \quad \nabla^2 f(x^{(k)}) \succ 0 \\ -\nabla f(x^{(k)}) & \text{otherwise} \end{cases}$$

or $v^{(k)} = -(\nabla^2 f(x_k) + \gamma_k I)^{-1}\nabla f(x^{(k)})$

References and further readings

- E. K.P. Chong, Wu-S. Lu, and S. H. Zak. *An Introduction to Optimization: With Applications to Machine Learning*. John Wiley & Sons, 2023. (ch 8 and 9)
- A. Beck. *Introduction to Nonlinear Optimization: Theory, Algorithms, and Applications with Python and MATLAB.* SIAM, 2023. (ch 4 and 5)
- L. Vandenberghe, *EE133A Lecture Notes*, UCLA.
- U. M. Ascher. *A First Course on Numerical Methods*. Society for Industrial and Applied Mathematics, 2011. (ch 9)