5. Single-variable optimization

- single variable minimization
- nonlinear equations and iterative methods
- bisection method
- Newton's method

Scalar minimization

$$\text{minimize} \quad f(x) \tag{5.1}$$

- $x \in \mathbb{R}$ is the *variable*
- $f : \mathbb{R} \to \mathbb{R}$ is the *objective function*

First-order necessary condition (FONC): a minimizer x^o satisfies

$$f'(x^o) = 0$$

- this condition is necessary but not sufficient
- there may be other points, known as stationary points or critical points, that satisfy $f'(\hat{x}) = 0$ and are not minimizers (maximizers or saddle points)
- we need to verify whether the solutions of $f'(\hat{x}) = 0$ are minimizers

Intuition and proof

Intuition:

- $f'(x) > 0$ implies that f is increasing at x (positive slope), therefore, a point $\tilde{x}$ slightly less than x gives $f(\tilde{x}) < f(x)$
- $f'(x) < 0$ implies that f is decreasing at x (negative slope), then a point $\tilde{x}$ slightly more than x gives $f(\tilde{x}) < f(x)$, which means that x is not a minimizer of f

Proof: if x^o is a local minimizer, then $f(x^o) \leq f(x^o + \epsilon)$ for sufficiently small ϵ; from the definition of the derivatives, when $\epsilon > 0$, the limit from the right is

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

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

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

hence, $0 \leq f'(x^o) \leq 0$, *i.e.*, $f'(x^o) = 0$

Example 5.1

consider the function

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

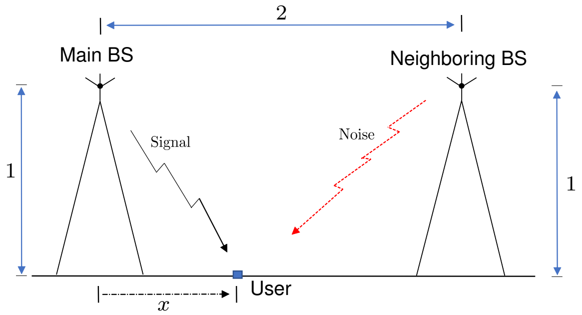
the first-order necessary conditions (FONC) is

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

the stationary points are $x = 1$ and $x = 3$

- the point $x = 1$ is not a local optimal point because the derivative $f'(x)$ does not change its sign around $x = 1$
- the point $x = 3$ is a local minimizer since the derivative changes its sign from negative to positive when passing through $x = 3$
- since $f(x) \to \infty$ as $|x| \to \infty$, the point $x = 3$ must be a global minimizer

Example 5.2



- power of the received signal measured by the user from each antenna is the reciprocal of the squared distance from the corresponding antenna
- find the position x of the user (relative to the main base station) that maximizes the 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 $x = 1 \pm \sqrt{2}$
- checking the objective values, we see that $x = 1 - \sqrt{2}$ gives larger objective, and the derivative changes its sign from positive to negative when passing through $x = 1 - \sqrt{2}$
- hence, $x^o = 1 - \sqrt{2}$ is a local maximizer
- it is a global maximizer since $f(x) \to 1 < f(x^o)$ as $|x| \to \infty$

Second-order condition

Necessary condition: if x^o is a local minimizer, then $f'(x^o) = 0$ and $f''(x^o) \geq 0$

Sufficient condition: if $f(x^o) = 0$ and $f''(x^o) > 0$, then x^o is a strict local minimizer

Necessary and sufficient condition: if $f''(x) \geq 0$ for all x ('f is convex'), then $x^\star$ is global minimizer if and only if $f'(x^\star) = 0$

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

Example 5.3

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

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

which holds for the points $\hat{x}_1 \approx 2.84$ 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

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

- since $f''(2.84) > 0$, $f''(-2.84) > 0$, and $f''(0) < 0$, the points $\hat{x}_1$ 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 $|x| \to \infty$

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

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

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

c) for $f(x) = x^3$, we have $f'(x) = 3x^2 = 0$, which holds for $\hat{x} = 0$; note that $f''(x) = 6x$ and $f''(0) = 0$, but $\hat{x} = 0$ is not a local minimizer since $f(x) < f(0)$ for any $x < 0$

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

Outline

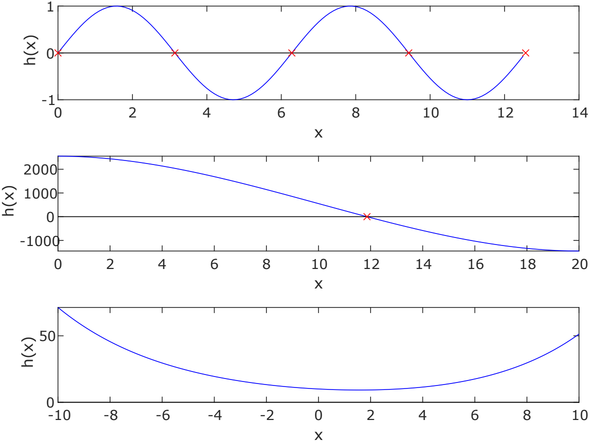
- single variable minimization
- **nonlinear equations and iterative methods**
- bisection method
- Newton's method

Nonlinear equation in one variable

$$h(x) = 0, \quad \text{where} \quad x \in [a, b]$$

- the *root* or *zero* is any solution of the above equation
- we assume h is a continuous function on the interval $[a, b]$
- observe that for minimizing $f(x)$, we can find stationary points by solving a nonlinear equation with $h(x) = f'(x)$

Example 5.4



(i) $h(x) = \sin(x)$ on $[0, 4\pi]$, (ii) $h(x) = x^3 - 30x^2 + 2552$ on $[0, 20]$, and (iii) $h(x) = 10\cosh\left(\frac{x}{4}\right) - x$ on $[-10, 10]$ where $\cosh(t) = \frac{e^t + e^{-t}}{2}$

Iterative methods

- for many nonlinear equations, obtaining a solution through an explicit formula or a deterministic, finite-step procedure is not feasible
- we often resorts to iterative techniques that start with an initial guess, denoted x_0, and yield a series of subsequent guesses $x^{(1)}, x^{(2)}, \ldots, x^{(k)}, \ldots$ which ideally converge to a root of the target continuous function

$$x^{(k)} \to x^\star \quad \text{as } k \to \infty$$

- one initial strategy to approximate root locations involves graphing the function to study its characteristics
- another complementary tactic is to compute the function's value at various points, aiming to discern intervals where its sign alters

Stopping iterative methods

- *Absolute error:* $|x^{(k)} - x^{(k-1)}| < \epsilon$
- *Relative error:* $|x^{(k)} - x^{(k-1)}|/|x^{(k)}| < \epsilon$
- *Function value:* $|h(x^{(k)})| < \epsilon$

here, ϵ is a tolerance level constant determined by the user

Convergence rate

assume the sequence $x^{(k)}$ converges to a limit $x^\star$

Linear convergence: if there exists a constant $c \in (0,1)$ such that

$$|x^{(k)} - x^\star| \leq c|x^{(k-1)} - x^\star|$$

for sufficiently large k

R-linear convergence if a positive constant M and a value $c \in (0,1)$ exist such that

$$|x^{(k)} - x^\star| \leq Mc^k \quad \text{for large values of } k$$

- $x^{(k)} = 1 + (1/2)^k$ linearly converges to $x^\star = 1$:

  $$|x^{(k+1)} - x^\star| = (1/2)^{k+1} = \frac{1}{2}|x^{(k)} - x^\star|$$

  meets the definition with $c = 1/2$
- every linearly convergent sequence is also R-linearly convergent, but the reverse is not necessarily true

Superlinear convergence: if a sequence $c_k > 0$ with $c_k \to 0$ exists and ensures that

$$|x^{(k)} - x^\star| \leq c_k |x^{(k-1)} - x^\star| \quad \text{for large } k$$

Quadratic convergence: if a constant $c > 0$ exists such that

$$|x^{(k)} - x^\star| \leq c |x^{(k-1)} - x^\star|^2 \quad \text{for large } k$$

- $x^{(k)} = 1 + (1/2)^{2^k}$ has quadratic convergence to $x^\star = 1$, as

$$|x^{(k+1)} - x^\star| = (1/2)^{2^{k+1}} = \left((1/2)^{2^k}\right)^2 = |x^{(k)} - x^\star|^2$$

  and this satisfies the definition with $c = 1$
- $x^{(k)} = 1 + (1/(k+1))^k$ has superlinear convergence:

$$|x^{(k)} - x^\star| = \frac{1}{(k+1)^k} = \frac{k^{k-1}}{(k+1)^k}\frac{1}{k^{k-1}} = \frac{k^{k-1}}{(k+1)^k}|x^{(k-1)} - x^\star|$$

  which satisfies the definition with $c_k = k^{k-1}/(k+1)^k$, a value that indeed approaches zero

Outline

- single variable minimization
- nonlinear equations and iterative methods
- **bisection method**
- Newton's method

The bisection method

given: a, b with $a < b$ and $h(a)h(b) < 0$, tolerance ϵ
repeat

1. $x = (a + b)/2$
2. compute $h(x)$; **if** $h(x) = 0$, **return** x
3. **if** $h(x)h(a) < 0$, $b = x$, **else**, $a = x$
4. **stop** if $a - b \leq \epsilon$

MATLAB implementation

```
function [p,k] = bisect(func,a,b,fa,fb,atol)
% assuming fa = func(a), fb = func(b), and fa*fb < 0,
% there is a value root in (a,b) such that func(root) = 0.
% this function returns in p a value such that
% | p - root | < atol
% and in k the number of iterations required.
if (a >= b) | (fa*fb >= 0) | (atol <= 0)
disp('something wrong with the input: quitting');
p = NaN; k=NaN;
return
end
k = ceil(log2 (b-a) - log2 (2*atol));
for i=1:k
p = (a+b)/2;
fp = feval(func,p);
if abs(fp) < eps, k = i; return, end
if fa * fp < 0
b = p;
fb = fp;
else
a = p;
```

Example 5.5

- for `func(x) =` x^3 `-` $30x^2$ `+ 2552`, starting from the interval [0,20] with a tolerance of 1×10^{-8}, the method converges to $x^\star \approx 11.86150151$ after 30 iterations

- for `func(x) = 2.5 sinh(x/4) - 1`, beginning with the interval [-10,10] and using a tolerance of 1×10^{-10}, the method converges to $x^\star \approx 1.5601412791$ after 37 iterations

  the associated MATLAB script for the second function is:

```
format long g
[x,k] = bisect('fex3',-10,10,fex3(-10),fex3(10),1.e-10)
function f = fex3(x)
f = 2.5 * sinh (x/4) - 1;
```

Convergence

let $[a^{(k)}, b^{(k)}]$ be the interval after iteration k, then

$$b^{(k)} - a^{(k)} = \frac{b^{(0)} - a^{(0)}}{2^k}$$

- after k iterations, the midpoint $x^{(k)} = \left(b^{(k)} + a^{(k)}\right)/2$ satisfies

$$\left|x^{(k)} - x^\star\right| \leq b^{(k)} - a^{(k)} \leq (1/2)^k \left(b^{(0)} - a^{(0)}\right)$$

  thus, it is R-linearly convergent with $c = 1/2$ and $M = b^{(0)} - a^{(0)}$
- the exit condition $b^{(k)} - a^{(k)} \leq \epsilon$ will be satisfied if

$$\log_2\left(\frac{b^{(0)} - a^{(0)}}{2^k}\right) = \log_2(b^{(0)} - a^{(0)}) - k \leq \log_2 \epsilon$$

  the algorithm therefore terminates after

$$\left\lceil \log_2\left(\frac{b^{(0)} - a^{(0)}}{\epsilon}\right)\right\rceil$$

  iterations ($\lceil \alpha \rceil$ is the smallest integer greater than or equal to α)

Outline

- single variable minimization
- nonlinear equations and iterative methods
- bisection method
- **Newton's method**

Derivation of Newton's method

leveraging Taylor's expansion, h can be approximated around the current iterate $x^{(k)}$ as:

$$h(x) = h(x^{(k)}) + h'(x^{(k)})(x - x^{(k)}) + \frac{h''(x^{(k)})(x - x^{(k)})^2}{2},$$

- if h were linear (meaning $h'' \equiv 0$), determining the root would involve solving for $0 = h(x^{(k)}) + h'(x^{(k)})(x^* - x^{(k)})$, leading to $x^* = x^{(k)} - \frac{h(x^{(k)})}{h'(x^{(k)})}$
- for nonlinear functions, the subsequent iterate is defined similarly:

$$x^{(k+1)} = x^{(k)} - \frac{h(x^{(k)})}{h'(x^{(k)})}, \quad k = 0, 1, 2, \ldots$$

  this iteration update omits the term $\frac{h''(x^{(k)})(x^\star - x^{(k)})^2}{2}$, operating under the assumption that $x^{(k)}$ is nearing the root $x^\star$

Newton's method

given: initial x and tolerance ϵ
repeat

1. compute $h(x)$ and $h'(x)$
2. **if** $|h(x)| < \epsilon$, **return** x
3. $x = x - h(x)/h'(x)$

Example 5.6

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

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

- applying Newton's method, we have

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

  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 (in fact it is an approximate global minimizer)

Example 5.7

applying Newton's method on $h(x) = 2\cosh\left(\frac{x}{4}\right) - x$ gives

$$x^{(k+1)} = x^{(k)} - \frac{2\cosh(x^{(k)}/4) - x^{(k)}}{0.5\sinh(x^{(k)}/4) - 1}$$

with tolerance of 1×10^{-8}, we have

- starting from $x_0 = 2$, 4 iterations are needed to get $x_1^\star = 2.35755106$ within the specified tolerance
- from $x_0 = 8$, 5 iterations are enough to reach $x_2^\star = 8.50719958$ to the given accuracy

for $x_0 = 8$, the values of $h(x^{(k)})$ evolve as:

k	0	1	2	3	4	5
$h(x^{(k)})$	$-4.76e-1$	$8.43e-2$	$1.56e-3$	$5.65e-7$	$7.28e-14$	$1.78e-15$

Secant method

the secant method modifies Newton's approach by estimating the derivative $h'(x^{(k)})$:

$$h'(x^{(k)}) \approx \frac{h(x^{(k)}) - h(x^{(k-1)})}{x^{(k)} - x^{(k-1)}}$$

incorporating this into Newton's formula, we get the secant method equation:

$$x^{(k+1)} = x^{(k)} - \frac{h(x^{(k)})(x^{(k)} - x^{(k-1)})}{h(x^{(k)}) - h(x^{(k-1)})}, \quad k = 1, 2, \ldots$$

Example: for $h(x) = 2\cosh(x/4) - x$, we implement the secant method using the provided tolerance from before with two initial guesses, x_0 and x_1: the $h(x^{(k)})$ values, initiating from $x_0 = 10$ and $x_1 = 8$, are as shown:

k	0	1	2	3	4	5	6
$h(x^{(k)})$	2.26	$-4.76\text{e}-1$	$-1.64\text{e}-1$	$2.45\text{e}-2$	$-9.93\text{e}-4$	$-5.62\text{e}-6$	$1.30\text{e}-9$

References

- E. KP. Chong and S. H. Zak. *An Introduction to Optimization,* John Wiley & Sons, 2013.
- Uri M. Ascher. *A First Course on Numerical Methods*. Society for Industrial and Applied Mathematics, 2011.
- L. Vandenberghe. *EE133A lecture notes,* Univ. of California, Los Angeles. (`http://www.seas.ucla.edu/~vandenbe/ee133a.html`)