Mathematics Methods for Computer Science

Instructor: Xubo Yang

SJTU-SE DALAB

Lecture

Nonlinear Systems I

Nonlinear Problems

Not all numerical problems can be solved with \ in Matlab.

Question

Have we already seen a nonlinear problem?

Question

Have we already seen a nonlinear problem?

$$\text{minimize } \|A\vec{x}\|_2$$
$$\text{such that } \|\vec{x}\|_2 = 1 \leftarrow \text{nonlinear!}$$

Root-Finding Problem

Given: $f : \mathbb{R}^n \to \mathbb{R}^m$

Find: $\vec{x}^*$ with $f(\vec{x}^*) = \vec{0}$

Root-Finding Applications

- Collision detection (graphics, astronomy)
- Graphics rendering (ray intersection)
- Robotics (kinematics)
- Optimization (line search)

Issue: Regularizing Assumptions

$$f(x) = \begin{cases} -1 \text{ when } x \leq 0 \\ 1 \text{ when } x > 0 \end{cases}$$

Issue: Regularizing Assumptions

$$f(x) = \begin{cases} -1 \text{ when } x \leq 0 \\ 1 \text{ when } x > 0 \end{cases}$$

$$g(x) = \begin{cases} -1 \text{ when } x \in \mathbb{Q} \\ 1 \text{ when } x \notin \mathbb{Q} \end{cases}$$

Typical Regularizing Assumptions

Continuous

$f(\vec{x}) \to f(\vec{y})$ as $\vec{x} \to \vec{y}$

Typical Regularizing Assumptions

Continuous

$f(\vec{x}) \to f(\vec{y})$ as $\vec{x} \to \vec{y}$

Lipschitz

$\|f(\vec{x}) - f(\vec{y})\|_2 \leq c\|\vec{x} - \vec{y}\|_2$ for all $\vec{x}, \vec{y}$ (same c)

Typical Regularizing Assumptions

Continuous

$f(\vec{x}) \to f(\vec{y})$ as $\vec{x} \to \vec{y}$

Lipschitz

$\|f(\vec{x}) - f(\vec{y})\|_2 \leq c\|\vec{x} - \vec{y}\|_2$ for all $\vec{x}, \vec{y}$ (same c)

Differentiable

$f'(\vec{x})$ exists for all $\vec{x}$

Typical Regularizing Assumptions

Continuous

$f(\vec{x}) \to f(\vec{y})$ as $\vec{x} \to \vec{y}$ 函数连续又被称为C^0

Lipschitz

$\|f(\vec{x}) - f(\vec{y})\|_2 \leq c\|\vec{x} - \vec{y}\|_2$ for all $\vec{x}, \vec{y}$ (same c)

Differentiable

$f'(\vec{x})$ exists for all $\vec{x}$

C^k

k derivatives exist and are continuous

C^∞: all derivatives of f exist and are continuous

Examples

$f(x) = cosx$ C^∞ and Lipschitz on R

$g(x) = x^2$ C^∞ but not Lipschitz on R

$|g(x) - g(0)| = x^2$ [0,1] "local Lipschitz"

$h(x) = |x|$ C^0, Lipschitz, not differentiable

Today

$$f : \mathbb{R} \to \mathbb{R}$$

Property of Continuous Functions

(中值定理)

Intermediate Value Theorem

Suppose that $f : [a, b] \to \mathbb{R}$ is continuous and that $f(a) < u < f(b)$ or $f(b) < u < f(a)$. Then, there exists $z \in (a, b)$ such that $f(z) = u$

Reasonable Input

- Continuous function $f(x)$
- $\ell, r \in \mathbb{R}$ with $f(\ell) \cdot f(r) < 0$ (why?)

由于连续，中间一定存在一个自变量的取值使得函数值为0

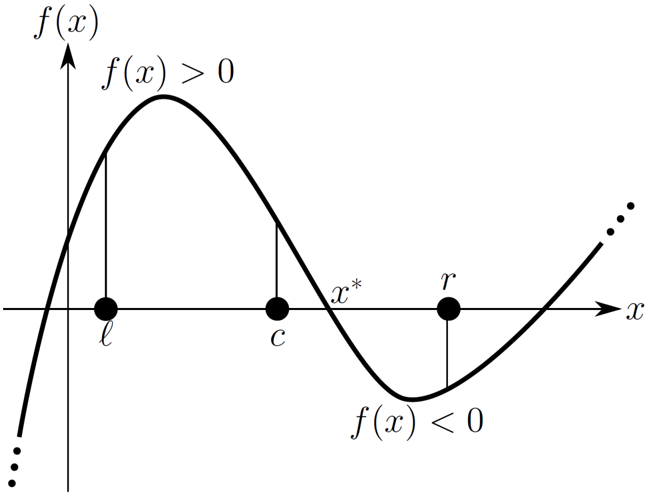
Bisection Algorithm

(二分法)

1. Compute $c = (\ell + r)/2$.
2. If $f(c) = 0$, return $x^* = c$.
3. If $f(\ell) \cdot f(c) < 0$, take $r \leftarrow c$.
   Otherwise take $\ell \leftarrow c$.
4. Return to step 1 until $|r - \ell| < \varepsilon$; then return c.

ε 可以理解为可以接受的根的误差

Bisection: Illustration



Two Important Questions

1. Does it converge?
(收敛)

Two Important Questions

1. Does it converge?
   Yes! Unconditionally.

Two Important Questions

1. Does it converge?
   Yes! Unconditionally.

2. How quickly?

Convergence Analysis

Examine E_k with

$$|x_k - x^*| < E_k$$

Bisection: Linear Convergence

$$E_{k+1} \leq \frac{1}{2} E_k$$

for $E_k \equiv |r_k - \ell_k|$

Fixed Points

$$g(x^*) = x^*$$

Fixed Points

$$g(x^*) = x^*$$

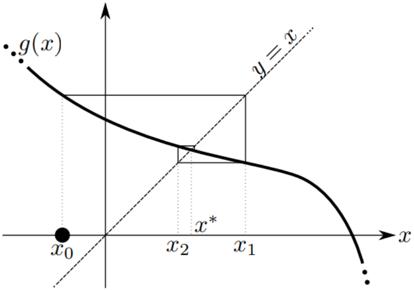
Question:

Same as root-finding?

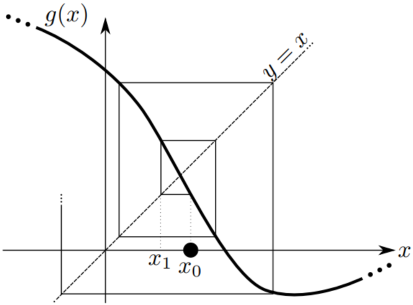
Simple Strategy

$$x_{k+1} = g(x_k)$$

Convergence of fixed point interation



(a) Convergence



(b) Divergence

由此图可以看出，即使初始点已经离最终结果很近，但是仍然可能发散而得不到最终正确解。

Convergence Criterion

$$\begin{aligned} E_k &\equiv |x_k - x^*| \\ &= |g(x_{k-1}) - g(x^*)| \end{aligned}$$

Convergence Criterion

$$
\begin{aligned}
E_k &\equiv |x_k - x^*| \\
&= |g(x_{k-1}) - g(x^*)| \\
&\leq c|x_{k-1} - x^*| \text{ if } g \text{ is Lipschitz} \\
&= cE_{k-1} \\
\Rightarrow E_k &\leq c^k E_0 \\
&\to 0 \text{ as } k \to \infty \; (c < 1)
\end{aligned}
$$

Alternative Criterion

Lipschitz near x^* with good starting point.

Alternative Criterion

Lipschitz near x^* with good starting point.

e.g. C^1 with $|g'(x^*)| < 1$

Convergence Rate of Fixed Point

When it converges... Always linear

Convergence Rate of Fixed Point

When it converges... Always linear

Often quadratic!

Convergence Rate of Fixed Point

Suppose g is differentiable with $g'(x^*) = 0$

$$g(x_k) = g(x^*) + \tfrac{1}{2}g''(x^*)(x_k - x^*)^2 + O\left((x_k - x^*)^3\right)$$

$$\begin{aligned}
E_k &= |x_k - x^*| \\
&= |g(x_{k-1}) - g(x^*)| \text{ as before} \\
&= \frac{1}{2}|g''(x^*)|(x_{k-1} - x^*)^2 + O\left((x_{k-1} - x^*)^3\right) \text{ from the Taylor argument} \\
&\le \frac{1}{2}(|g''(x^*)| + \varepsilon)(x_{k-1} - x^*)^2 \text{ for some } \varepsilon \text{ so long as } x_{k-1} \text{ is close to } x^* \\
&= \frac{1}{2}(|g''(x^*)| + \varepsilon)E_{k-1}^2.
\end{aligned}$$

Example (Fixed point interation)

Example 8.2 (Fixed point iteration). We can apply fixed point iteration to solving $x = \cos x$ by iterating $x_{k+1} = \cos x_k$. A numerical example starting from $x_0 = 0$ proceeds as follows:

k	0	1	2	3	4	5	6	7	8	9
x_k	0	1.000	0.540	0.858	0.654	0.793	0.701	0.764	0.722	0.750

In this case, fixed point iteration converges linearly to the root $x^* \approx 0.739085$.

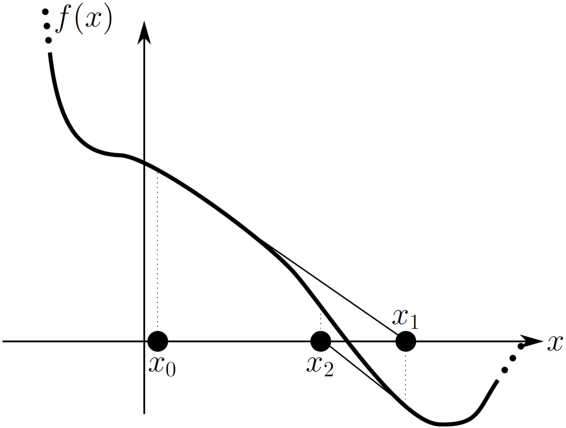
The root-finding problem $x = \sin x^2$ satisfies the condition for quadratic convergence near $x^* = 0$. For this reason, fixed point iteration $x_{k+1} = \sin x_k^2$ starting at $x_0 = 1$ converges more quickly to the root:

k	0	1	2	3	4	5	6	7	8	9
x_k	1	0.841	0.650	0.410	0.168	0.028	0.001	0.000	0.000	0.000

Finally, the roots of $x = e^x + e^{-x} - 5$ do *not* satisfy convergence criteria for fixed point iteration. Iterates of the failed fixed point scheme $x_{k+1} = e^{x_k} + e^{-x_k} - 5$ starting at $x_0 = 1$ are shown below:

k	0	1	2	3	4	5	6	7
x_k	1	-1.914	1.927	2.012	2.609	8.660	5760.375	$\cdots$

Approach for Differentiable $f(x)$



Newton's Method

$$f\left(x\right) \approx f\left(x_k\right) + f'\left(x_k\right)\left(x - x_k\right)$$

Newton's Method

$$f\left(x\right) \approx f\left(x_k\right) + f'\left(x_k\right)\left(x - x_k\right)$$

$$\Rightarrow \quad \mathsf{x}_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$$

Newton's Method

$$f(x) \approx f(x_k) + f'(x_k)(x - x_k)$$

$$\Rightarrow \quad \mathsf{x}_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$$

Fixed point iteration on

$$g(x) \equiv x - \frac{f(x)}{f'(x)}$$

Convergence of Newton's Method

Define

$$g(x) = x - \frac{f(x)}{f'(x)}$$

Differentiating

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

Convergence of Newton

Simple Root

A root x^* with $f'(x^*) \neq 0$.

Convergence of Newton

Simple Root

A root x^* with $f'(x^*) \neq 0$.

Quadratic convergence in this case!
(Otherwise, can be linear or worse)

Example (Newton's method)

Example 8.3 (Newton's method)**.** The last part of Example 8.2 can be expressed as a root-finding problem on $f(x) = e^x + e^{-x} - 5 - x$. The derivative of $f(x)$ in this case is $f'(x) = e^x - e^{-x} - 1$, so Newton's method can be written

$$x_{k+1} = x_k - \frac{e^{x_k} + e^{-x_k} - 5 - x_k}{e^{x_k} - e^{-x_k} - 1}.$$

This iteration quickly converges to a root starting from $x_0 = 2$:

k	0	1	2	3	4
x_k	2	1.9161473	1.9115868	1.9115740	1.9115740

Example 8.4 (Newton's method failure)**.** Suppose $f(x) = x^5 - 3x^4 + 25$. Newton's method applied to this function gives the iteration

$$x_{k+1} = x_k - \frac{x_k^5 - 3x_k^4 + 25}{5x_k^4 - 12x^3}.$$

These iterations converge when x_0 is sufficiently close to the root $x^* \approx -1.5325$. For instance, the iterates starting from $x_0 = -2$ are shown below:

k	0	1	2	3	4
x_k	-2	-1.687500	-1.555013	-1.533047	-1.532501

Farther away from this root, however, Newton's method can fail. For instance, starting from $x_0 = 0.25$ gives a divergent set of iterates:

k	0	1	2	3	4
x_k	0.25	149.023256	119.340569	95.594918	76.599025

Issue

Differentiation is hard!

Example

Example (Rocket design)

Suppose we are designing a rocket and wish to know how much fuel to add to the engine.

For a given number of gallons x, we can write a function f(x) giving the maximum height of the rocket during flight; our engineers have specified that the rocket should reach a height h, so we need to solve f(x) = h.

Evaluating f(x) involves simulating a rocket as it takes off and monitoring its fuel consumption, which is an expensive proposition. Even if f is differentiable, we might not be able to evaluate f' in a practical amount of time.

Secant Method

用割线代替切线

$$f'(x_k) \approx \frac{f(x_k)-f(x_{k-1})}{x_k-x_{k-1}}$$

Secant Method

$$f'(x_k) \approx \frac{f(x_k)-f(x_{k-1})}{x_k-x_{k-1}}$$

$$x_{k+1} = x_k - \frac{f(x_k)(x_k - x_{k-1})}{f(x_k) - f(x_{k-1})}$$

Secant Method

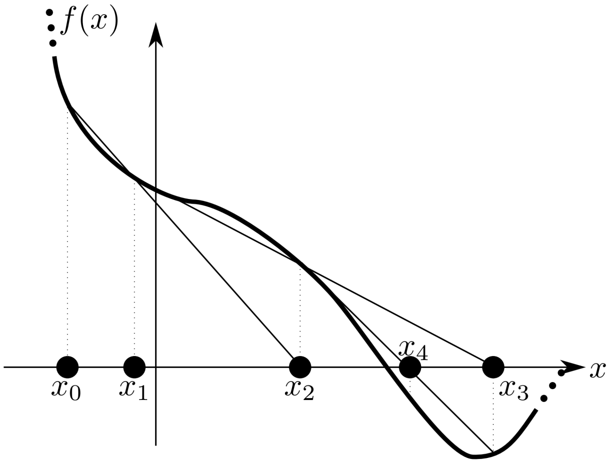
$$f'(x_k) \approx \frac{f(x_k)-f(x_{k-1})}{x_k - x_{k-1}}$$

$$x_{k+1} = x_k - \frac{f(x_k)(x_k - x_{k-1})}{f(x_k) - f(x_{k-1})}$$

Trivia:

Converges at rate $\frac{1+\sqrt{5}}{2} \approx 1.6180339887\ldots$
(”Golden Ratio”)

Secant Method



Example (Secant method)

Example 8.6 (Secant method)**.** Suppose $f(x) = x^4 - 2x^2 - 4$. Iterates of Newton's method for this function are given by

$$x_{k+1} = x_k - \frac{x_k^4 - 2x_k^2 - 4}{4x_k^3 - 4x_k}.$$

Contrastingly, iterates of the secant method for the same function are given by

$$x_{k+1} = x_k - \frac{(x_k^4 - 2x_k^2 - 4)(x_k - x_{k-1})}{(x_k^4 - 2x_k^2 - 4) - (x_{k-1}^4 - 2x_{k-1}^2 - 4)}.$$

By construction, a less expensive way to compute these iterates is to save and reuse $f(x_{k-1})$ from the previous iteration. We can compare the two methods starting from $x_0 = 3$; for the secant method we also choose $x_{-1} = 2$:

k	0	1	2	3	4	5	6
x_k (Newton)	3	2.385417	2.005592	1.835058	1.800257	1.798909	1.798907
x_k (secant)	3	1.927273	1.882421	1.809063	1.799771	1.798917	1.798907

The two methods exhibit similar convergence on this example.

Hybrid Methods

Want: Convergence rate of secant/Newton with convergence guarantees of bisection

Hybrid Methods

Want: Convergence rate of secant/Newton with convergence guarantees of bisection

e.g. **Dekker's Method:** Take secant step if it is in the bracket, bisection step otherwise

Single-Variable Conclusion

- Unlikely to solve exactly, so we settle for iterative methods
- Must check that method converges at all
- Convergence rates:
  - Linear: $E_{k+1} \leq CE_k$ for some $0 \leq C < 1$
  - Superlinear: $E_{k+1} \leq CE_k^r$ for some $r > 1$
  - Quadratic: $r = 2$
  - Cubic: $r = 3$
- Time per iteration also important