Numerical Optimization

Unit 1: One-dimensional Optimization

Che-Rung Lee

Scribe: 張雅芳

March 1, 2011

Basic optimization strategy (algorithm)

Basic Optimization Strategy

1. Given an initial guess x_0
2. For $k = 0, 1, 2, \ldots$ until converge
   1. Test x_k for convergence
   2. Calculate the search direction p_k
   3. Determine the step length α_k
   4. $x_{k+1} = x_k + \alpha_k p_k$

Questions:

- How to determine convergence?
- How to calculate the search direction p_k?
- How to determine the step length α_k?

How to determine convergence?

Assume the problem is to find the minimum of a function $f(x)$.

- What are the characteristic of the optimal solution(s)?

Definition (1.1)

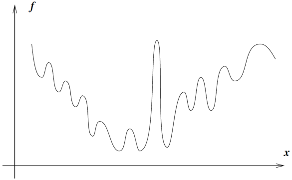
A point x^* is a global minimum of $f(x)$ if for all y in the feasible set of x,

$$f(x^*) \leq f(y)$$

Definition (1.2)

A point x^* is a local minimum of $f(x)$ in the neighborhood $N(x^*, r)$ if for all $y \in N(x^*, r)$,

$$f(x^*) \leq f(y)$$



$f(x) = x^4 \cos(1/x) + 2x^4, \qquad f(0) = 0,$

From computational viewpoint

1. Global minimum is not always available.
2. Local minimum is hardly to compute precisely.
3. Usually, approximate solutions are good enough.

What is a good approximation?

1. For the solution domain, a solution x is a good approximation to the minimizer x^*, if $|x - x^*| < \epsilon|x^*|$ for some tolerance parameter ϵ.
2. For the function domain, x is a good approximation if $|f(x) - f(x^*)| < \epsilon|f(x^*)|$.
3. Those two things are different: (example)

Other stopping criteria

1. Set the maximum number of iterations.
2. Stop if $|x_k - x_{k-1}| \leq \epsilon$ or $|f(x_k) - f(x_{k-1}| \leq \epsilon$.

How to calculate p_k and α_k?

How to calculate p_k?

- For one-dimensional problems, the search direction p_k can only be $+1$ or -1.
- We usually make $\|p_k\| = 1$.

How to calculate α_k?

- The Cauchy property:
  If a sequence $x_0, x_1, x_2, \ldots$, converges to x^*, $|x_{\kappa+1} - x_\kappa|$ converges to 0.
  $\Rightarrow$ the step size α_k should converge to 0.

Example: Find the minimization of unimodal functions

Definition (1.3)

A function $f(x)$, defined in $[a, b]$, is called *unimodal* if for $x \in [a, x^*]$, $f(x)$ is monotonically decreasing, and for $x \in [x^*, b]$, $f(x)$ is monotonically increasing.

- How to find x^*?
- Recall the basic strategy
    - How to determine the convergence?
    - How to decide the search direction?
    - How to decide the step size?

The binary search algorithm

The binary search algorithm

1. Let $x_1 = (a+b)/2, \alpha_0 = (b-a)/2$.
2. For $k = 1, 2, 3, \ldots$ until $\alpha_{k-1} < \epsilon$
   1. Evaluate $f(x_k)$ and $f(x_k + \epsilon)$.
   2. If $f(x_k + \epsilon) > f(x_k)$, $p_k = -1$. Otherwise, $p_k = +1$.
   3. $\alpha_k = \alpha_{k-1}/2$
   4. Let $x_{k+1} = x_k + \alpha_k p_k$.

- Can the algorithm find the minimizer x^* of a unimodal function?
- How fast can the algorithm find x^* (or stop)?

Convergence rate

Definition (1.4)

Suppose a sequence $\{x_k\}$ converges to x^*. The rate of convergence is defined as

$$\mu = \lim_{k\to\infty} \frac{|x_{k+1} - x^*|}{|x_k - x^*|}$$

- $0 < \mu < 1$, the smaller μ, the faster convergence.
- $|x_k - x^*| \le 2\alpha_k$ and $\alpha_k = \left(\frac{1}{2}\right)^{k+1}(b-a)$. $\Rightarrow \mu = 1/2$.
- If we call (a)(b)(c)(d) an iteration, it takes $k \ge \log_2\left(\frac{b-a}{\epsilon}\right)$ iterations to stop.
- This kind of convergence is called *linear convergence*.

Differentiable functions

- Suppose $f(x)$ is twice differentiable in its domain, and

$$g(x) = f'(x), h(x) = g'(x) = f''(x).$$

- Can the differentiability of $f(x)$ help to answer the three questions?

Convergence test

- If $\hat{x}$ is a local minimum of $f(x)$, $g(\hat{x}) = 0$ and $h(\hat{x}) > 0$.
- If $\hat{x}$ is a local maximum of $f(x)$, $g(\hat{x}) = 0$ and $h(\hat{x}) < 0$.
- But $g(\hat{x}) = 0$ only doesn't imply optimality.

Calculation of the search direction p_k

- If $g(x) > 0$, $f(x)$ is increasing.$\Rightarrow$ $p_k = -1$.
- If $g(x) < 0$, $f(x)$ is decreasing.$\Rightarrow$ $p_k = +1$.

Root finding algorithms

- Since $g(x) = 0$ is the necessary condition of the optimality, we can use the root finding algorithm to find x^* such that $g(x^*) = 0$.
- Two algorithms will be illustrated.
  1. Newton's method
  2. Secant method
- One more algorithm, the polynomial interpolation method, will be introduced later.

Note

Although both algorithms cannot guarantee to fine the optimal solution, they will give us many important ideas.

Newton's method for root finding

- **Problem**: Given a function $g(x)$, find x^* s.t. $g(x^*) = 0$.
- **Idea**: At each iteration, it approximates $g(x)$ by a straight line $\ell_k(x)$, which passes the point $(x_k, g(x_k))$ and has slope $g'(x_k)$.
$$\ell_k(x) = g'(x_k)(x - x_k) + g(x_k)$$
Then it uses the solution of $\ell_k(x) = 0$, $\hat{x}$, to approximate the solution of $g(x) = 0$.
$$\hat{x} = x_k - \frac{g(x_k)}{g'(x_k)}$$
- Newton's method uses $\hat{x}$ as the new approximate solution.

Newton's Method for root finding

1. Given an initial guess x_0
2. For $k = 1, 2, \ldots$ until $|g(x_k)| < \epsilon$,
$$x_{k+1} = x_k - \frac{g(x_k)}{g'(x_k)}$$

Convergence of Newton's method

Theorem (Convergence of the Newton method)

If x_0 *is sufficiently close to* x^* *and* $g(x), g'(x), g''(x)$ *are continuous near* x^* *and* $g'(x) \neq 0$*, then*

$$\lim_{k\to\infty} \frac{|x_{k+1} - x^*|}{|x_k - x*|^2} < M$$

for some $M > 0$*.*

Definition (1.5)

If a sequence $\{x_k\}$ satisfies

$$\lim_{k\to\infty} \frac{|x_{k+1} - x^*|}{|x_k - x^*|^2} = M \text{ and } \{x_k\} \longrightarrow x^*$$

We say $\{x_k\}$ converges to x^* *quadratically*.

Quadratic convergence

Does Newton's method converge faster than the binary search algorithm?

Example (Comparison of linear and quadratic convergence)

quadratic convergence	linear convergence
$M = 1, \quad \lvert x_0 - x^*\rvert = 0.1$	$M = 0.1, \quad \lvert x_0 - x^*\rvert = 0.1$
$\lvert x_1 - x^*\rvert = 0.01$	0.01
$\lvert x_2 - x^*\rvert = 10^{-4}$	10^{-3}
$\lvert x_3 - x^*\rvert = 10^{-8}$	10^{-4}

Yes, if Newton's method converges, it is very fast.

Fail to converge

Newton's method does not guarantee convergence.

Example ($g(x) = x^3 - 3x^2 + x + 3$ and $x_0 = 1$)

$$\begin{aligned}
g'(x) &= 3x^2 - 6x + 1 \\
x_1 &= x_0 - \frac{g(x_0)}{g'(x_0)} = 1 - \frac{2}{-2} = 2 \\
x_2 &= x_1 - \frac{g(x_1)}{g'(x_1)} = 2 - \frac{1}{1} = 1 \\
x_3 &= 2, x_4 = 1, \ldots
\end{aligned}$$

Definition (1.6)

- The convergence of Newton's method is called *local*, because it is sensitive to the initial guess.
- The convergence of the binary search is called *global*, because it guarantee to converge no matter which initial guess is given.

Secant method

In Newton's method, $g(x)$ is approximate by a line, $\ell_k(x)$, the tangent of $g(x)$ at x_k. The secant method replaces (approximates) the tangent by the secant line

$$g'(x_k) \approx \frac{g(x_{k-1}) - g(x_k)}{x_{k-1} - x_k} = h(x_k)$$

$$\hat{\ell}_k(x) = h(x_k)(x - x_k) + g(x_k)$$

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

Secant method for root finding

1. Given an initial guess x_0, x_1
2. For $k = 1, 2, \ldots$ until $|g(x_k)| < \epsilon$

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

Convergence of the secant method

Does it work? Yes, and pretty well.

Theorem (Convergence of the secant method)

If x_0 *is sufficiently close to* x^* *and* $g(x), g'(x), g''(x)$ *are continuous near* x^* *and* $g'(x) \neq 0$*, then for an* $\alpha \in [1, 2]$*,*

$$\lim_{k\to\infty} \frac{|x_{k+1} - x^*|}{|x_k - x^*|^\alpha} < M.$$

Definition (1.7)

If a sequence $\{x_k\}$ satisfies

$$\lim_{k\to\infty} \frac{|x_{k+1} - x^*|}{|x_k - x^*|^\alpha} = M \text{ and } \{x_k\} \longrightarrow x^*$$

for some $\alpha > 1$, we say $\{x_k\}$ converges to x^* *superlinearly*.