Biostatistics 615 - Statistical Computing

Topic 5
Single-dimensional Optimization Methods

Hyun Min Kang

Topic Overview

- One-dimensional Root Finding Algorithms
  1. Bisection Method
  2. Fixed Point Iteration
  3. Newton-Raphson Method
  4. The Secant Method
- One-dimensional Minimization Algorithms
  1. Finding a bracketing interval
  2. The Golden Search Algorithm
  3. Parabolic Interpolation
  4. Brent's Method

Topic Overview

- One-dimensional Root Finding Algorithms
  1. Bisection Method
  2. Fixed Point Iteration
  3. Newton-Raphson Method
  4. The Secant Method
- One-dimensional Minimization Algorithms
  1. Finding a bracketing interval
  2. The Golden Search Algorithm
  3. Parabolic Interpolation
  4. Brent's Method

One-dimensional root finding and minimization

Root finding

Given A function $f : \mathbb{R} \rightarrow \mathbb{R}$

Root finding solve $f(x) = 0$.

A root is a point x^* where $f(x^*) = 0$.

Minimization

Given $F : \mathbb{R} \rightarrow \mathbb{R}$

Minimization : minimize $F(x)$.

A strict local minimizer is a point x^* where $F(x^*) < F(x)$ for all x near x^*, $x \neq x^*$.

Root finding vs. Minimization

- Root finding and optimization are closely related problems.
- Each type of problem can be converted to the other.
  - **Root finding** $\rightarrow$ **Optimization**: minimize $F(x) = \{f(x)\}^2$.
  - **Optimization** $\rightarrow$ **Root finding**: solve $f(x) = F'(x) = 0$.
- In both cases, you will find all solutions of the original problem, and possibly some spurious solutions that do not solve the original problem.

Iterative Methods

How it works

- Start with an initial guess x_0.
- Produce a sequence of values $x_1, x_2, \ldots$.
- The value x_{i+1} depends on $f(x_0), \ldots, f(x_i)$.

Key assumptions

- Basic additions, multiplications, and function evaluations.
- Basic arithmetic is negligible; bottleneck is function evaluation.
- An efficient method requires a small number of function evaluations.

Errors in iterative methods

- The error at step n is $e_n = |x_n - x^*|$.
- An iterative method is of order p if $e_{n+1} \leq Ce_n^p$

Topic Overview

- One-dimensional Root Finding Algorithms
  1. Bisection Method
  2. Fixed Point Iteration
  3. Newton-Raphson Method
  4. The Secant Method
- One-dimensional Minimization Algorithms
  1. Finding a bracketing interval
  2. The Golden Search Algorithm
  3. Parabolic Interpolation
  4. Brent's Method

Bisection Method

Intuition

- Assume a continuous function $f(x)$.
- The intermediate value theorem then tells us that:
  - if $f(x)$ is positive at $x = a$ and negative at $x = b$...
  - there has to be a root in between a and b.

Bisection Algorithm

1. Start with initial points a_0, b_0 so that $f(a_0) > 0, f(b_0) < 0$
2. At the midpoint: $x_0 = (a_0 + b_0)/2$, calculate $f(x_0)$.
3. If $f(x_0) = 0$, then it is done; otherwise, one of two things happens:
   - $f(a_0)f(x_0) < 0$, in which case we set $a_1 = a_0$, $b_1 = x_0$; or
   - $f(a_0)f(x_0) > 0$, in which case we set $a_1 = x_0$, $b_1 = b_0$.
4. Set $x_1 = (a_1 + b_1)/2$, calculate $f(x_1)$.
5. Repeat steps 3–4 until $b_n - a_n$ is small enough.

Bisection Method (cont'd)

Key Properties of Bisection Methods

- It is impossible to estimate e_n exactly for this method, and the error may actually grow in some steps
- .. but if we interpret e_n as the maximum possible error $e_n = (b_n - a_n)/2$, then $e_{n+1} = (1/2)e_n$.
- The method is linearly convergent, with convergence factor $1/2$.

Implementation of Bisection Methods

- Recursive divide-and conquer implementation
- Loop-based implementation

Hands-on Session

Visit `https://bit.ly/615top05r`

- Bisection method for root finding

Topic Overview

- One-dimensional Root Finding Algorithms
  1. Bisection Method
  2. Fixed Point Iteration
  3. Newton-Raphson Method
  4. The Secant Method
- One-dimensional Minimization Algorithms
  1. Finding a bracketing interval
  2. The Golden Search Algorithm
  3. Parabolic Interpolation
  4. Brent's Method

Fixed Point Iteration

What is Fixed Point?

Consider an equivalent equation of $f(x) = 0$ represented as follows:

$$g(x) = x$$

We refer to its solution as **a fixed point** of g.

What is Fixed Point Algorithm?

For $n = 0, 1, \ldots, N$, iteratively evaluate

$$x_{n+1} = g(x_n)$$

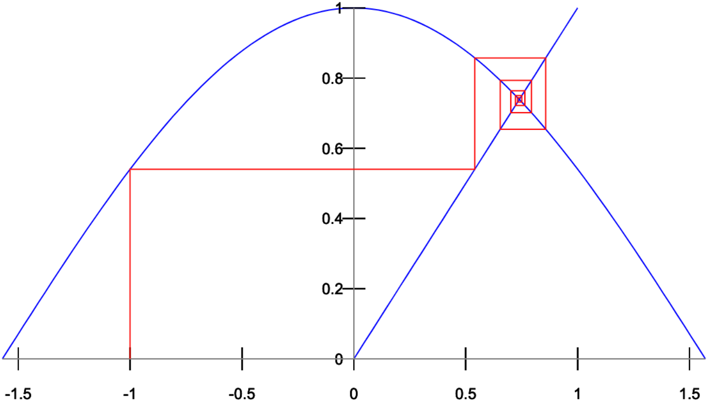
Do you think x_n will converge to the root of $f(x) = 0$?

Fixed Point Iteration - An Example

Solving for a specific equation

- Target function $f(x) = \cos x - x = 0$.
- Define
$$g(x) = f(x) + x = \cos x$$
.
- Then $g(x) = x$ when $f(x) = 0$.
- The fixed point algorithm evaluates
$$x_{n+1} = g(x_n) = \cos x$$
for $n = 0, 1, \ldots, N$.

Fixed Point Iteration - Illustration



source: Wikipedia Commons (Tinctorius)

Fixed Point Iteration - Another Example

Calculating a square root

- Target function $f(x) = \frac{1}{2}\left(\frac{a}{x} - x\right) = 0$.
  - Because $f(\sqrt{a}) = 0$, a root is $x = \sqrt{a}$.
- Define

$$g(x) = f(x) + x = \frac{1}{2}\left(\frac{a}{x} + x\right)$$

  .

- Then $g(x) = x$ when $f(x) = 0$.
- The fixed point algorithm evaluates

$$x_{n+1} = g(x_n) = \frac{1}{2}\left(\frac{a}{x} + x\right)$$

  for $n = 0, 1, \ldots, N$.

Would x_n converge to $\sqrt{a}$?

Fixed Point Iteration - Convergence

- Let x^* be a fixed point of g.
  - If g is continuously differentiable, and $|g'(x^*)| < 1$
  - ... then fixed point iteration will converge linearly to x^*
  - ... if x_0 is close enough to x^*.
  - If $|g'(x^*)| > 1$, fixed point iteration will not converge to x^*.
- When $g'(x^*) \neq 0$, $e_{n+1} < \max_{\xi \in N(x^*)} |g'(\xi)| e_n$, $N(x^*)$ is a neighborhood of x^*.
- When $g'(x^*) = 0$ but $g''(x^*) \neq 0$, $e_{n+1} < \max_{\xi \in N(x^*)} |g''(\xi)| e_n^2$.

Visit https://bit.ly/615top05r

- Fixed point method for root finding

Topic Overview

- One-dimensional Root Finding Algorithms
  1. Bisection Method
  2. Fixed Point Iteration
  3. Newton-Raphson Method
  4. The Secant Method
- One-dimensional Minimization Algorithms
  1. Finding a bracketing interval
  2. The Golden Search Algorithm
  3. Parabolic Interpolation
  4. Brent's Method

Newton-Raphson method

- A special way to construct the fixed point iteration

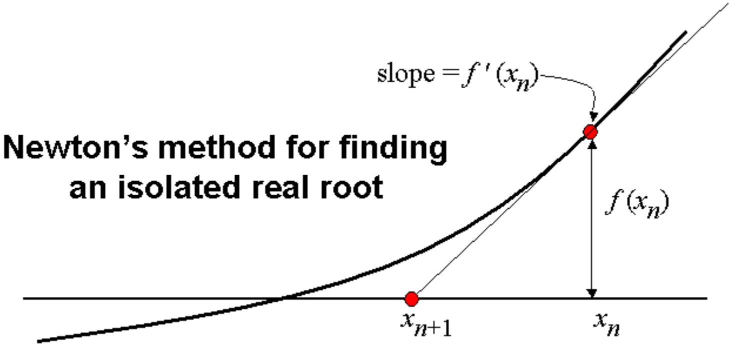
$$g(x) = x - \frac{f(x)}{f'(x)}$$

- **Algorithm:** given an initial guess x_0; for $n = 0, 1, \ldots, N$, calculate the tangent to f at x_n and intersect it with the x-axis. This is the next point:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

- This method converges quadratically (order of 2) as long as f has two continuous derivatives. Why?

Illustration of Newton-Raphson method



$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

source: Wikipedia Commons (Mohsalsae)

Hands-on Session

Visit https://bit.ly/615top05r

- Newton-Raphson method for root finding

Newton-Raphson Algorithm : Potential problems

- **Huge jump**: from x_n to x_{n+1}, if $f'(x_n) \approx 0$.
- **Cycles**: in the x_n. If x_{n+2} happens to be equal to x_n ...
- **Divergence**: the x_n get bigger and bigger ...
- **Require Derivative**: closed form of $f'(x)$ may be tricky to obtain ...
- **Solutions**: ?

Topic Overview

- One-dimensional Root Finding Algorithms
  1. Bisection Method
  2. Fixed Point Iteration
  3. Newton-Raphson Method
  4. The Secant Method
- One-dimensional Minimization Algorithms
  1. Finding a bracketing interval
  2. The Golden Search Algorithm
  3. Parabolic Interpolation
  4. Brent's Method

The Secant Method

- A discrete version of Newton-Raphson method that does not require derivatives.
- Given two initial points x_0, x_1: for $n = 0, 1 \ldots, n$, draw a straight line through $(x_n, f(x_n))$ and $(x_1, f(x_1))$. The next point x_{n+2} is the place where this line intersects the x-axis. We can calculate that

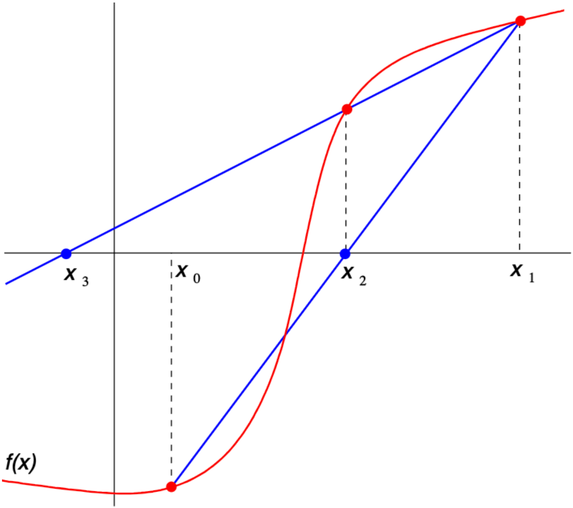
$$x_{n+2} = \frac{x_n f(x_{n+1}) - x_{n+1} f(x_n)}{f(x_{n+1}) - f(x_n)}$$

  Or equivalently

$$x_{n+2} = x_{n+1} - \frac{f(x_{n+1})}{\{f(x_{n+1}) - f(x_n\})/(x_{n+1} - x_n)}$$

- Suppose $f'(x^*) \neq 0$. If x_0 is sufficiently close to x^*, then the secant method has $e_{n+2} \approx C e_{n+1} e_n$, thus it converges of order $\frac{\sqrt{5}+1}{2} \approx 1.618$.

Illustration of Secant method



source: Wikipedia Commons (Jitse Niesen)

Hands-on Session

Visit https://bit.ly/615top05r

- Secant method for root finding

Topic Overview

- One-dimensional Root Finding Algorithms
  1. Bisection Method
  2. Fixed Point Iteration
  3. Newton-Raphson Method
  4. The Secant Method
- One-dimensional Minimization Algorithms
  1. Finding a bracketing interval
  2. The Golden Search Algorithm
  3. Parabolic Interpolation
  4. Brent's Method

Minimization Problem

Finding global minimum

- The lowest possible value of the function
- Very hard problem to solve generally

Finding local minimum

- Smallest value within finite neighborhood
- Relatively easier problem

Maximization Problem

- Consider a complex function $F(x)$ (e.g. likelihood)
- Find x which $F(x)$ is maximum or minimum value
- Maximization and minimization are equivalent
  - Replace $F(x)$ with $-F(x)$

Derivative-Free Minimization

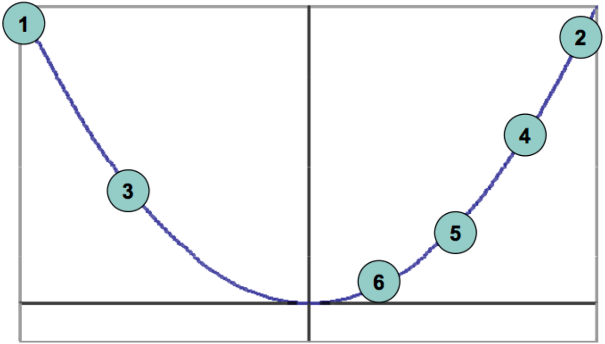
Problem setting

- A function $F : \mathbb{R} \to \mathbb{R}$ (may be costly to evaluate)
- The derivative of F is not easy to evaluate.
- Want to find the minimum of F without derivative.

A Derivative-Free Minimization Strategy

1. Find 3 points a, b, c such that
   - $a < b < c$
   - $F(b) < F(a)$ and $F(b) < F(c)$
2. Then search for minimum by
   - Selecting trial point in the interval $[a, c]$
   - Keep minimum and flanking points

Minimization after Bracketing



- Goal: starting from two points a_0 and b_0, find a, b, c such that
  - $a < b < c$
  - $F(b) < F(a)$ and $F(b) < F(c)$

Hands-on Session

Visit https://bit.ly/615top05r

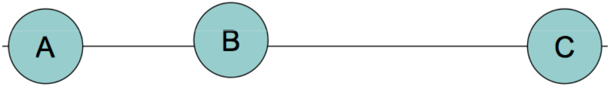
- Finding a bracketing interval for minimization

Topic Overview

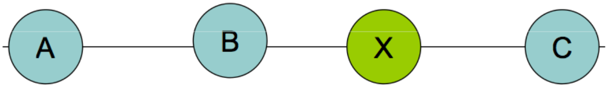
- One-dimensional Root Finding Algorithms
  1. Bisection Method
  2. Fixed Point Iteration
  3. Newton-Raphson Method
  4. The Secant Method
- One-dimensional Minimization Algorithms
  1. Finding a bracketing interval
  2. The Golden Search Algorithm
  3. Parabolic Interpolation
  4. Brent's Method

Step 2 : Finding Minimum After Bracketing

- Given 3 points such that
  - $a < b < c$
  - $F(b) < F(a)$ and $F(b) < F(c)$
- How do we select new trial point?



What we want



We want to minimize the size of next search interval, which will be either from A to X or from B to C

- If $F(X) < F(B)$, the next search interval will be (B, C)
- If $F(X) > F(B)$, the next search interval will be (A, X)

What would be an effective strategy to determine X*?*

- Define

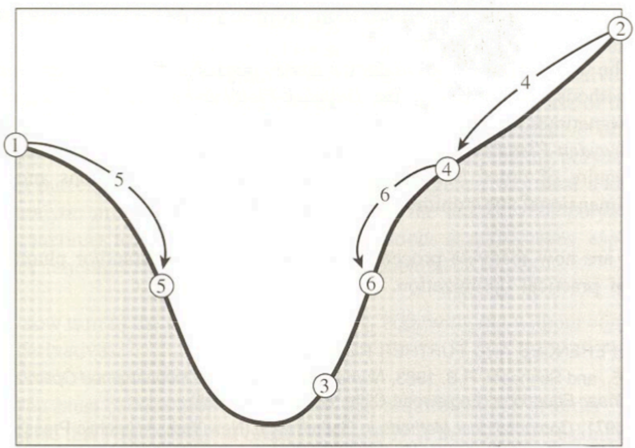
$$w = \frac{|AB|}{|AC|} = \frac{b-a}{c-a}, \quad z = \frac{|BX|}{|AC|} = \frac{x-b}{c-a}$$

Segments will have length either $|BC| = (1-w)|AC|$ or $|AX| = (w+z)|AC|$.

- Optimal case:
  - The two possible next search intervals have the same length
  - The proportion of reducing the interval lenghts remains the same over iterations.

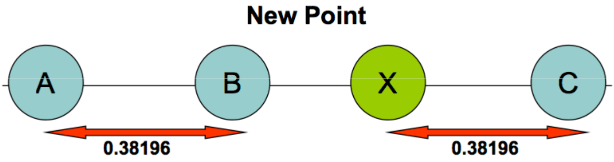
$$\begin{cases} \frac{|BC|}{|AC|} = \frac{|AX|}{|AC|} \\ \frac{|BX|}{|BC|} = \frac{|AB|}{|AC|} \end{cases} \rightarrow \begin{cases} 1-w = w+z \\ \frac{z}{1-w} = w \end{cases} \rightarrow w = \frac{3-\sqrt{5}}{2} = 0.38197$$

The Golden Search



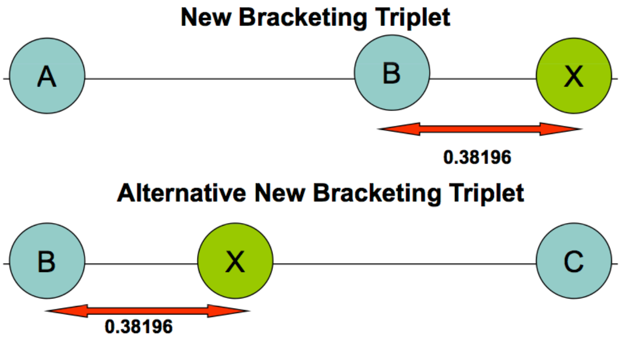
The Golden Ratio

The Golden Ratio



The number 0.38196 is related to the *golden mean* studied by Pythagoras

The Golden Ratio



Golden Search

- Reduces bracketing by 38.2% after function evaluation
- Performance is independent of the function that is being minimized
  - Guaranteed bound of number of function evaluations
- In many cases, better schemes are available

Hands-on Session

Visit https://bit.ly/615top05r

- The Golden Search Algorithm

Discussion

- What is the convergence rate of the Golden Search?
- How can the convergence rate be improved?

Topic Overview

- One-dimensional Root Finding Algorithms
  1. Bisection Method
  2. Fixed Point Iteration
  3. Newton-Raphson Method
  4. The Secant Method
- One-dimensional Minimization Algorithms
  1. Finding a bracketing interval
  2. The Golden Search Algorithm
  3. Parabolic Interpolation
  4. Brent's Method

Parabolic Approximation

Why Parabolic Approximation?

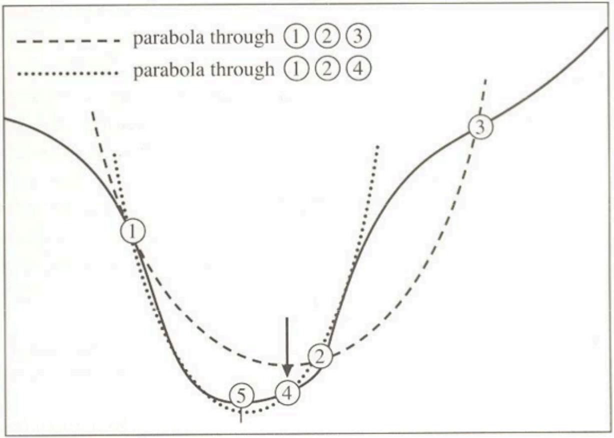
- For a root finding problem, linear interpolation likely works better than binary search.
- For a minimization problem, parabolic interpolation likely works better than golden search.

Key Idea

- Approximate the function as a quadratic equation (i.e. parabola)
- Estimate the minimal point based on parabola, and repeat the iteration.
- An extension of linear interpolation to the minimization problem.
- Typically more efficient than golden search, but convergence is not guaranteed.

Do you expect that parabolic approximation works always better than golden search?

Illustrating Parabolic Approximation



Brent's method (1973)

Key Idea

- A careful mixture of golden-section search and parabolic interpolation.
- Predict whether parabolic interpolation would behave poorly.
- If so, switch to golden section search.

Features

- Combines the reliability of golden-section search with the efficiency of parabolic interpolation.
- The most widely used method for single-dimensional minimization
- Does not require evaluation of derivatives.
- Detailed implementation is quite complicated.

Hands-on Session

Visit https://bit.ly/615top05r

- Using Brent's method