Introduction to Optimization and Nonlinear Equations

Zeyu Lu & Yuqiu Yang

1 Safe Univariate Methods:

2 Root finding

3 Stopping and Condition

Safe Univariate Methods:

Lattice Search

Golden Search

Bisection

Root finding

Newton's Method

The Secant Method

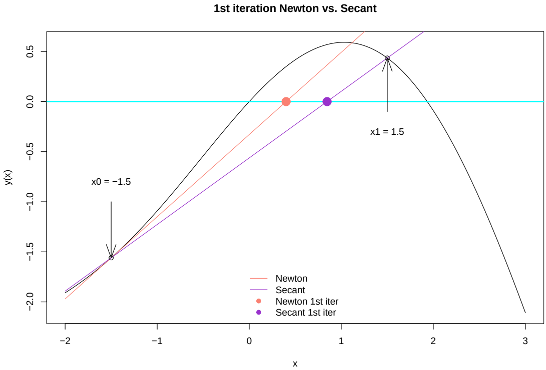
If g' is hard or even impossible to find, we can approximate $g'(x) \approx \frac{g(x+h) - g(x)}{h}$.

The iteration formula now becomes $x_{n+1} = x_n - g(x_n)\frac{x_n - x_{n-1}}{g(x_n) - g(x_{n-1})}$

Notice two initial approximations are required instead of one like the Newton's method.

The Secant Method: Geometrical Interpretation

Let $f(x) = sin(x) - (\frac{x}{2})^2$, $x_0 = -1.5$ and $x_1 = 1.5$

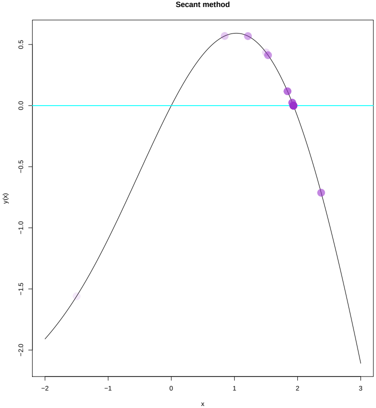


x_{n+1} is taken to be the abscissa of the point of intersection between the secant through $(x_{n-1}, f(x_{n-1}))$ and $(x_n, f(x_n))$ and the x-axis.

The Secant Method: An Example

Let $f(x) = sin(x) - (\frac{x}{2})^2$

$x_0 = -1.5$ and $x_1 = 1.5$



n	xn	fn
0	-1.5000000	-1.5599950
1	1.5000000	0.4349950
2	0.8458689	0.5696740
3	3.6127549	-3.7169217
4	1.2135787	0.5686801
5	1.5319385	0.4125362
6	2.3730538	-0.7127606
7	1.8402932	0.1172352
8	1.9155445	0.0238328
9	1.9347459	-0.0013123
10	1.9337439	0.0000131
11	1.9337538	0.0000000

The Secant Method: Several Definitions

Given $n+1$ distinct pairs $\{(x_0, g(x_0)), (x_1, g(x_1)), \ldots, (x_n, g(x_n))\}$, we will define:

$int(x_0, x_1, \ldots, x_n)$: the smallest interval that contains $x_0, \ldots, x_n$

The divided differences

$$g[x_0, x_1, \ldots, x_j, x] = \frac{g[x_0, x_1, \ldots, x_{j-1}, x] - g[x_0, x_1, \ldots, x_j]}{x - x_j}$$

, and

$$g[x_0, x] = \frac{g(x) - g(x_0)}{x - x_0}$$

The Secant Method: Newton's Interpolation Formula

Given $n+1$ distinct pairs $\{(x_0, g(x_0)), (x_1, g(x_1)), \ldots, (x_n, g(x_n))\}$, we can interpolate these points using a polynomial $q(x)$ of degree n. Specifically,

$$q(x) = g(x_0) + \sum_{j=1}^{n} g[x_0, x_1, \ldots, x_j] \prod_{i=0}^{j-1} (x - x_i)$$

, with the remainder

$$g(x) - q(x) = \frac{g^{n+1}(\xi) \prod_{i=0}^{n} (x - x_i)}{(n+1)!}$$

, where $\xi \in int(x_0, x_1, \ldots, x_n, x)$

The Secant Method: Order of convergence

According to Newton's interpolation formula, we have

$$g(x) = g(x_n) + (x - x_n)g[x_{n-1}, x_n] + \frac{1}{2}(x - x_n)(x - x_{n-1})g''(\xi)$$

where $g[x_{n-1}, x_n] = \dfrac{g(x_n) - g(x_{n-1})}{x_n - x_{n-1}}$, and $\xi \in int(x, x_n, x_{n-1})$

By the Secant Method, we have

$$x_{n+1} = x_n - g(x_n)\frac{x_n - x_{n-1}}{g(x_n) - g(x_{n-1})} \Rightarrow$$

$$0 = g(x_n) + (x_{n+1} - x_n)g[x_{n-1}, x_n]$$

The Secant Method: Order of convergence

Let the root the Secant Method approaches be c, then

$$0 = g(c) - g(x_n) - (x_{n+1} - x_n)g[x_{n-1}, x_n] =$$

$$g[x_{n-1}, x_n](c - x_{n+1}) + \frac{1}{2}(c - x_n)(c - x_{n-1})g''(\xi)$$

By the mean value theorem, we have

$$g[x_{n-1}, x_n] = g'(\eta), \eta \in (x_{n-1}, x_n)$$

Let $\epsilon_n = c - x_n$, we get $0 = g'(\eta)\epsilon_{n+1} + \frac{1}{2}\epsilon_n\epsilon_{n-1}g''(\xi) \Rightarrow$
$\epsilon_{n+1} = \frac{g''(\xi)}{2g'(\eta)}\epsilon_n\epsilon_{n-1}$

The Secant Method: Order of convergence

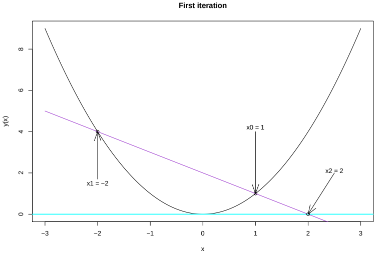
Now suppose the Secant Method converges, then when $n \to \infty$, $\xi \approx c$ and $\eta \approx c$. Let $C = \dfrac{g''(c)}{2g'(c)}$, then $|\epsilon_{n+1}| = C|\epsilon_n||\epsilon_{n-1}|$

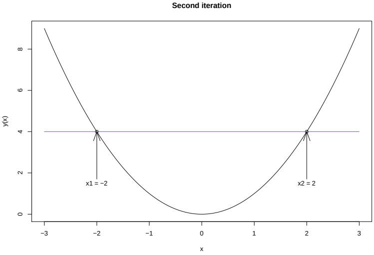
To find the order of convergence, we find p such that
$|\epsilon_{n+1}| \approx M|\epsilon_n|^p \Rightarrow$
$M|\epsilon_n|^p = MM|\epsilon_{n-1}|^p|\epsilon_{n-1}| \Rightarrow |\epsilon_n| = M|\epsilon_{n-1}|^{(1+p)/p}$
This implies $p = (1+p)/p \Rightarrow p = 1 + \phi \approx 1.618$

Since the exponent 1.618 lies between 1 (linear convergence) and 2 (quadratic convergence), the convergence rate of the Secant Method is called *superlinear*.

The Secant Method: Pros and Cons

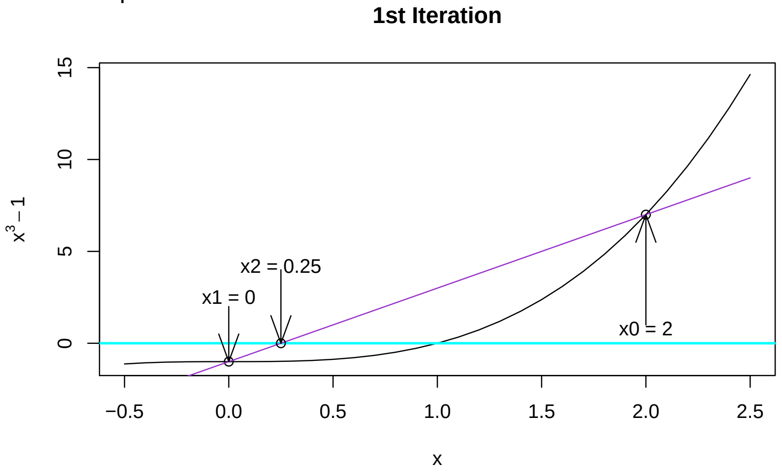
1. Pros:
   - Superlinear convergence
   - No need to evaluate derivatives
2. Cons:
   - Convergence is not guaranteed
   - Not well behaved when g is relatively flat



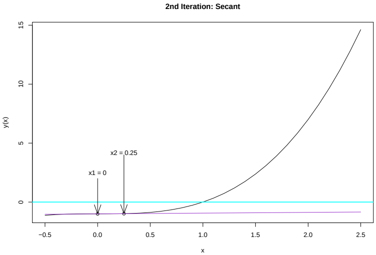


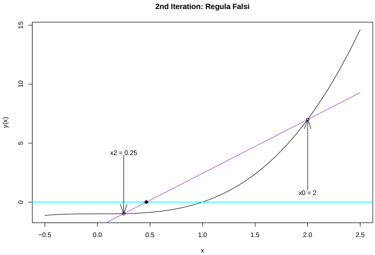
Regula Falsi: A Motivative Example

Borrowing the idea of the Bisection Method, what if we start with two points that straddle the root?



Regula Falsi: A Motivative Example



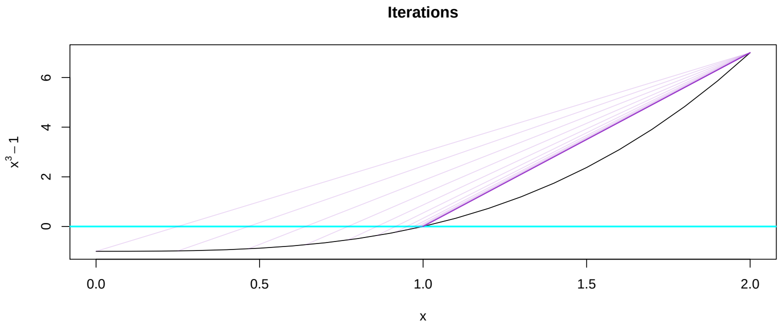


In this case, since the slope of the secant used in the Secant Method is so close to 0, the root is out of our scope.

However, by straddling the root, the Regula Falsi makes sure that the new root is always between the previous two values.

Regula Falsi

A variant of the Secant Method where instead of choosing the secant through $(x_n, g(x_n))$ and $(x_{n-1}, g(x_{n-1}))$, one finds the secant through $(x_n, g(x_n))$ and $(x_{n'}, g(x_{x'}))$ where $n' < n$ is the largest index for which $g(x_n)g(x_{n'}) < 0$.



Regula Falsi: Order of Convergence

Like the Bisection Method, the Regula Falsi is “safe”. However, from the previous example, we see that this method is in general a first-order method.

Especially, if $g(x)$ is convex on $[x_0, x_1]$, then

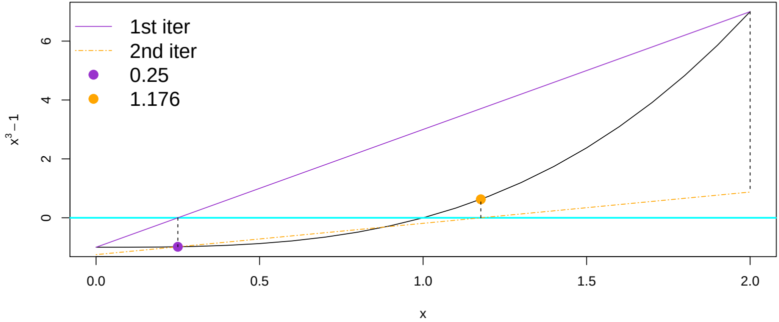
$$|\epsilon_{n+1}| \approx C|\epsilon_n||\epsilon_0| = C'|\epsilon_n|$$

where $C = \dfrac{g''(c)}{2g'(c)}$

The Regula Falsi Method tends to retain one end-point for several iterations. As a result, it can be a good “start” method or a part of a “hybrid” method, but it should not be used near a root.

Illinois Algorithm: Building on Regula Falsi

In the previous example, if we artificially create a shallower secant, then maybe the end-point will no longer be retained.

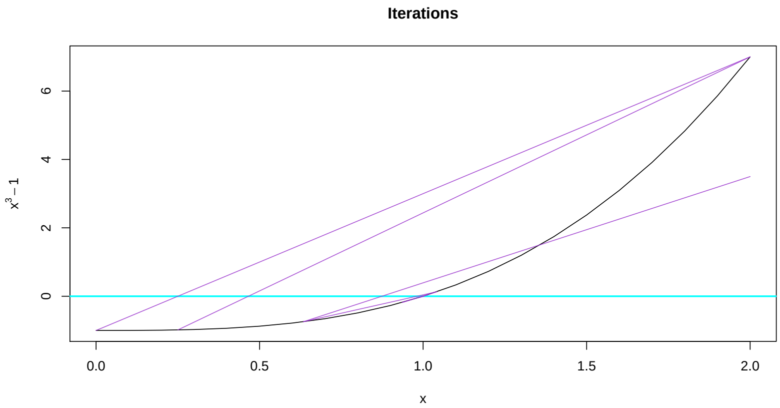


By dividing the function value at 2 by 8 and calculating the new secant, we find a new root on right of the root. In the next iternation, the new root 1.176 instead of 2 will be used.

Illinois Algorithm

1. During the Regula Falsi procedure, once we find one end-point has been retained more than once, we half the function value at that point, find the secant line and the new root.
2. If the point still retains, we repeat Step 1.
3. Once the point changes, we proceed with the Regula Falsi

Illinois Algorithm: An Example



Compared with the Regula Falsi Method, the Illinois Algorithm gets in a small neighborhood of the root in just 4 or 5 iterations.

Illinois Algorithm: Order of Convergence

Recall the errors of the Secant Method satisfy:
$\epsilon_{n+1} = \dfrac{g''(\xi)}{2g'(\eta)}\epsilon_n\epsilon_{n-1}$.
If g'' is continuous, then when the Illinois Algorithm gets into a sufficient small neighborhood of the root c, we can assume g' and g'' have constant sign.
This implies that $\dfrac{\epsilon_{n+1}}{\epsilon_n\epsilon_{n-1}}$ also has constant sign.
Since $g_{n-1}g_n < 0 \Rightarrow \epsilon_{n-1}\epsilon_n < 0$, we then necessarily have the sign of ϵ_n's follow one of the two schemes:

$$\cdots + - + + - + + - + \ldots$$

or

$$\cdots + - - + - - + - - \ldots$$

Illinois Algorithm: Order of Convergence

Previous analysis shows that asymptotically, an end-point will be retained twice in consecutive three iterations.
In other words, we will perform the Illinois step (halving the function value) once every third time.
Further asymptotic analysis shows that an Illinois step has $\epsilon_{n+1} \approx -\epsilon_n$
Putting the pieces together, we have

$$\epsilon = -\epsilon_{n-1} \Rightarrow \epsilon_{n+1} = -C\epsilon_{n-1}^2 \Rightarrow \epsilon_{n+2} = C^2\epsilon_{n-1}^3$$

Via finding p such that $|\epsilon_n| = M|\epsilon_{n-1}|^p$, we get $\dfrac{3}{p^2} = p \Rightarrow p \approx 1.44$

Successive Parabolic Interpolation

Recall the Newton's Method for optimization can be written as

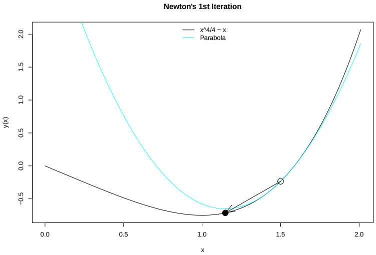
$$x_{n+1} = x_n - \frac{f'(x_n)}{f''(x_n)}$$

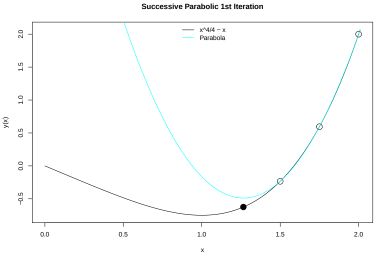
The essence of Newton's Method is locally approximating a function via a sequence of parabolas.

If f' or f'' is hard to find, the Successive Parabolic Interpolation Method can be used to find the extremum.

In each iteration, we fit a parabola to 3 unique points and replace the "oldest" one with the extremum of the fitted parabola.

Successive Parabolic Interpolation: vs. Newton's Method



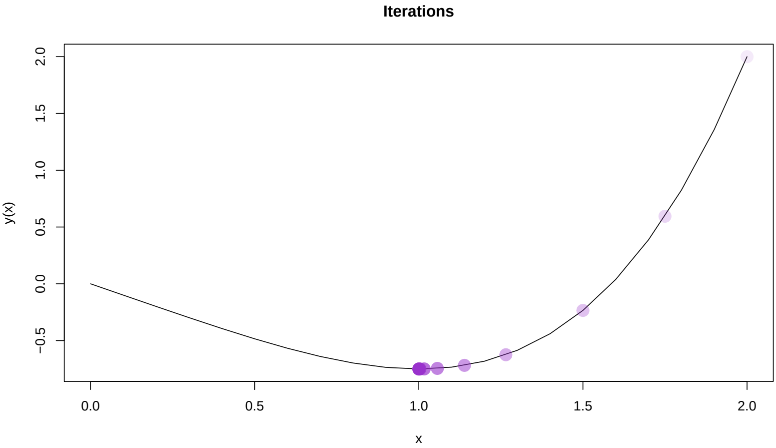


The parabola fitted in the Successive Parabolic Interpolation depends on the 3 points we chose.

In the next iteration, Newton's Method will fit a parabola based on the point 1.1481.

The Successive Parabolic Interpolation will fit a parabola based on 1.75, 1.5, and 1.2653.

Successive Parabolic Interpolation: An Example



The Order of Convergence of the Successive Parabolic Interpolation is approximately 1.3.

Summary: Convergence Rates

	Root Finding	Optimization
Linear	Bisection, Regula Falsi	Golden Section
Superlinear	Secant, Illinois	Parabolic Interpolation
Quadratic	Newton	Newton

Stopping and Condition

Three options for termination

1. Too many steps

- Usually indicate a serious error in problem specification

2. No change in x or No change in the function values

- Need to check if a root or an extremum is being approached
- Root finding: in some cases, no x will produce $g(x)$ "close" to 0
- Root finding: at some function value, there may appear to be multiple roots
- Both absolute change $|x_{n+1} - x_n| < \epsilon_x$ and relative change $||x_{n+1} - x_n| < |x_n|\epsilon_x|$ can be used for x.
- For g only absolute change for root finding. Relative change is appropriate with optimization.