Numerical Optimization

Unconstrained Optimization (II)

Shirish Shevade

Computer Science and Automation
Indian Institute of Science
Bangalore 560 012, India.

NPTEL Course on Numerical Optimization

Unconstrained Optimization

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

Unconstrained problem

$$\min_{x \in \mathbb{R}} \quad f(x)$$

- What are *necessary and sufficient conditions* for a local minimum?
  - Necessary conditions: Conditions satisfied by every local minimum
  - Sufficient conditions: Conditions which guarantee a local minimum
- Easy to characterize a local minimum if f is *sufficiently* smooth

Stationary Points

Let $f : \mathbb{R} \rightarrow \mathbb{R}, f \in \mathcal{C}^1$.
Consider the problem, $\min_{x \in \mathbb{R}} \; f(x)$.

Definition

x^* is called a *stationary point* if $f'(x^*) = 0$.

Necessity of an Algorithm

- Consider the problem

$$\min_{x \in \mathbb{R}} \ (x-2)^2$$

- We first find the stationary points (which satisfy $f'(x) = 0$).

$$f'(x) = 0 \Rightarrow 2(x-2) = 0 \Rightarrow x^* = 2.$$

- $f''(2) = 2 > 0 \Rightarrow\ x^*$ is a strict local minimum.
- Stationary points are found by solving a nonlinear equation,

$$g(x) \equiv f'(x) = 0.$$

- Finding the real roots of $g(x)$ may not be always easy.
  - Consider the problem to minimize $f(x) = x^2 + e^x$.
  - $g(x) = 2x + e^x$
  - Need an algorithm to find x which satisfies $g(x) = 0$.

One Dimensional Optimization

- Derivative-free methods (Search methods)
- Derivative-based methods (Approximation methods)
- Inexact methods

Unimodal Functions

- Let $\phi : \mathbb{R} \rightarrow \mathbb{R}$
- Consider the problem,

$$\min_{x \in \mathbb{R}} \quad \phi(x)$$

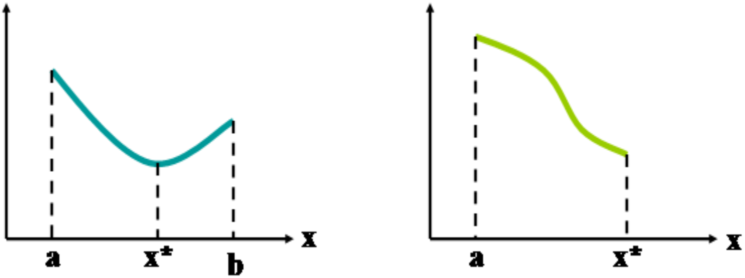
- Let x^* be the minimum point of $\phi(x)$ and $x^* \in [a, b]$

Definition

The function ϕ is said to be *unimodal* on $[a, b]$ if for $a \leq x_1 < x_2 \leq b$,

$$\begin{aligned} x_2 < x^* \quad &\Rightarrow \quad \phi(x_1) > \phi(x_2), \\ x_1 > x^* \quad &\Rightarrow \quad \phi(x_2) > \phi(x_1). \end{aligned}$$

Unimodal Functions



Unimodal functions

Derivative-free Methods

Let $f : \mathbb{R} \rightarrow \mathbb{R}$

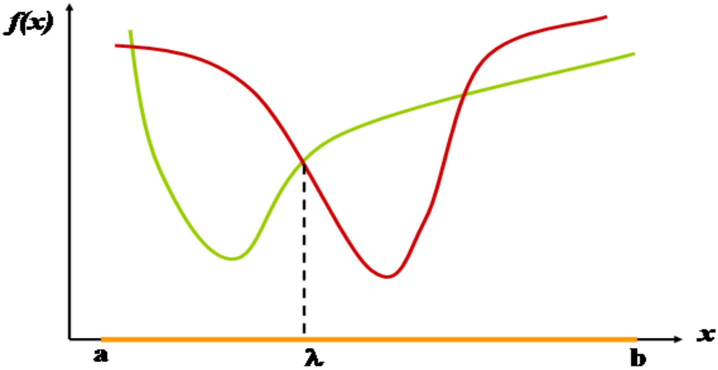
Unconstrained problem

$$\min_{x \in \mathbb{R}} \quad f(x)$$

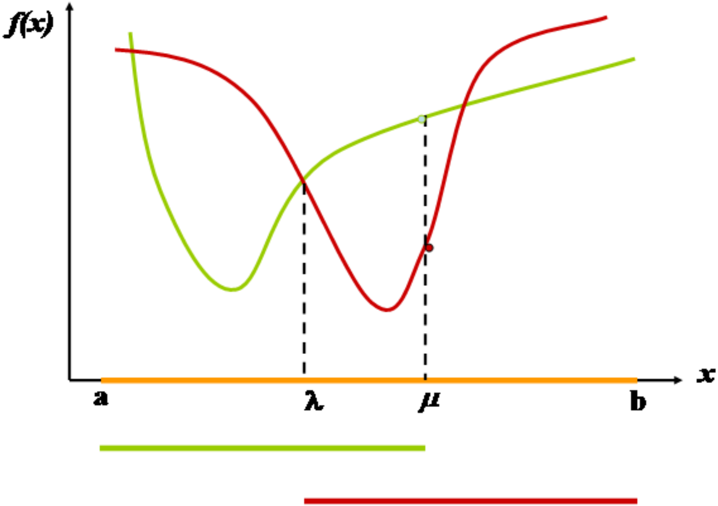
- Dichotomous search
- Fibonacci search
- Golden-section search

Require,

- Interval of uncertainty, $[a, b]$, which contains the minimum of f
- f to be *unimodal* in $[a, b]$.



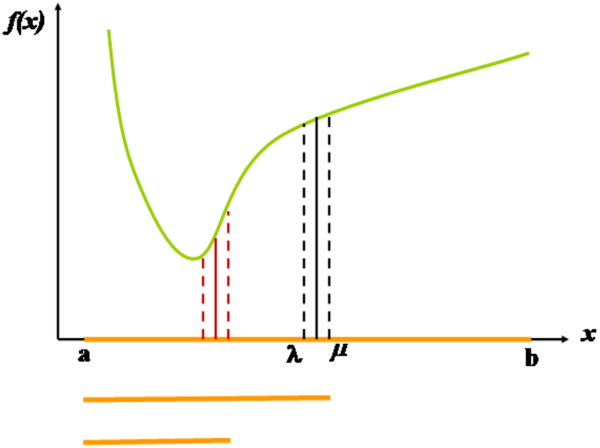
Function values at three points not enough to reduce the interval of uncertainty



Function values at four points enough to reduce the interval of uncertainty

Dichotomous Search

- Place λ and μ symmetrically, each at a distance ϵ from the mid-point of $[a, b]$



Dichotomous Search: Algorithm

1. **Input**: Initial interval of uncertainty, $[a, b]$,
2. **Initialization**:
   $k = 0, a^k = a, b^k = b, \epsilon \, (> 0), l$ (final length of uncertainty interval)
3. **while** $(b^k - a^k) > l$
4. $\quad \lambda^k = \frac{a^k + b^k}{2} - \epsilon, \mu^k = \frac{a^k + b^k}{2} + \epsilon$
5. $\quad$ if $f(\lambda^k) \geq f(\mu^k)$
6. $\qquad a^{k+1} = \lambda^k, b^{k+1} = b^k$
7. $\quad$ else
8. $\qquad b^{k+1} = \mu^k, a^{k+1} = a^k$
9. $\quad$ endif
10. $\quad k := k + 1$
11. **endwhile**
12. **Output**: $x^* = \frac{a^k + b^k}{2}$

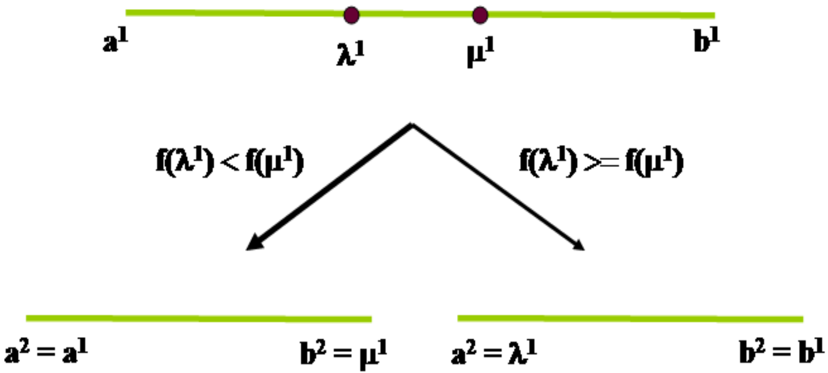
Dichotomous Search: An Example

Consider, $\min_x \; (1/4)x^4 - (5/3)x^3 - 6x^2 + 19x - 7$

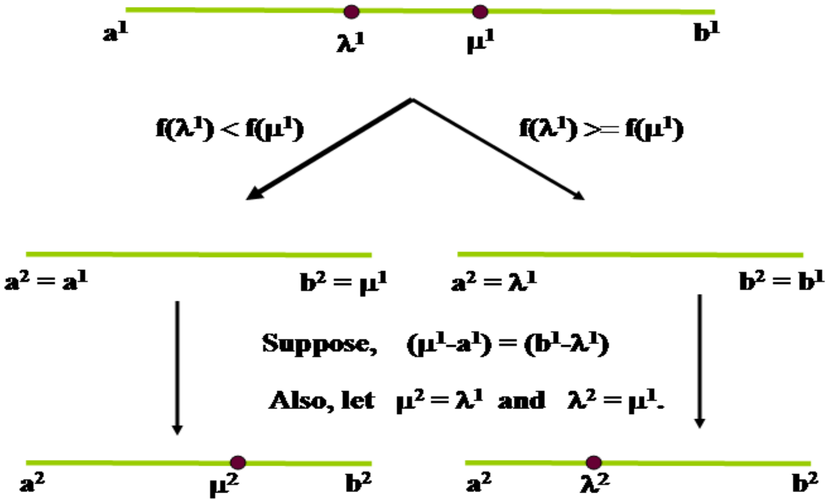
k	a^k	b^k	$b^k - a^k$
0	-4	0	4
1	-4	-1.98	2.02
2	-3.0001	-1.98	1.0201
3	-3.0001	-2.4849	.5152
$\vdots$	$\vdots$	$\vdots$	$\vdots$
10	-2.5669	-2.5626	.0043
$\vdots$	$\vdots$	$\vdots$	$\vdots$
20	-2.5652	-2.5652	4.65e-6
$\vdots$	$\vdots$	$\vdots$	$\vdots$
23	-2.5652	-2.5652	5.99e-7

$x^* = -2.5652, f(x^*) = -56.2626$

Fibonacci Search

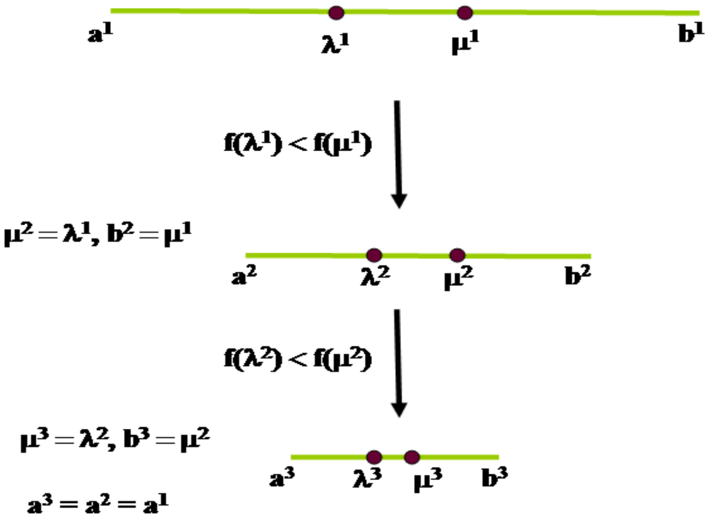


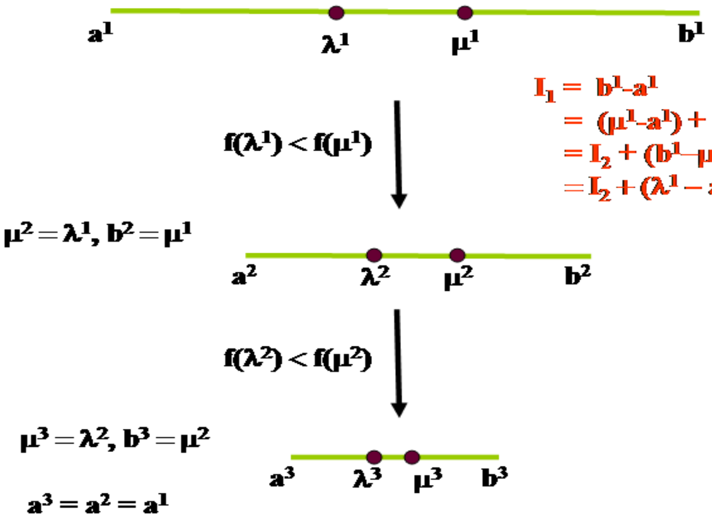
Fibonacci Search



Need one function evaluation at every iteration $k = 2, 3, \ldots$

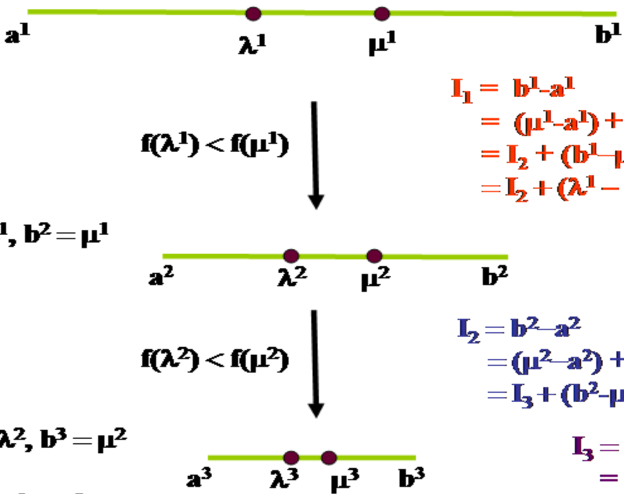
Fibonacci Search





$$
\begin{aligned}
I_1 &= b^1 - a^1 \\
&= (\mu^1 - a^1) + (b^1 - \mu^1) \\
&= I_2 + (b^1 - \mu^1) \\
&= I_2 + (\lambda^1 - a^1)
\end{aligned}
$$

Fibonacci Search



$$\begin{aligned} I_1 &= b^1 - a^1 \\ &= (\mu^1 - a^1) + (b^1 - \mu^1) \\ &= I_2 + (b^1 - \mu^1) \\ &= I_2 + (\lambda^1 - a^1) \end{aligned}$$

$\mu^2 = \lambda^1,\ b^2 = \mu^1$

$$\begin{aligned} I_2 &= b^2 - a^2 \\ &= (\mu^2 - a^2) + (b^2 - \mu^2) \\ &= I_3 + (b^2 - \mu^2) \end{aligned}$$

$\mu^3 = \lambda^2,\ b^3 = \mu^2$

$a^3 = a^2 = a^1$

$$\begin{aligned} I_3 &= b^3 - a^3 \\ &= \mu^2 - a^2 \\ &= \lambda^1 - a^1 \end{aligned}$$

Note: I_1 = I_2 + I_3

Fibonacci Search

We have $I_1 = I_2 + I_3$.
Generalizing further, we get

$$\begin{aligned}
I_1 &= I_2 + I_3 \\
I_2 &= I_3 + I_4 \\
&\vdots \\
I_n &= I_{n+1} + I_{n+2}
\end{aligned}$$

Assumption: The interval for iteration $n+2$ vanishes ($I_{n+2} = 0$).

Fibonacci Search

$$
\begin{array}{lclcl}
I_{n+1} & = & I_n - I_{n+2} & = & 1 I_n \\
I_n & = & I_{n+1} + I_{n+2} & = & 1 I_n \\
I_{n-1} & = & I_n + I_{n+1} & = & 2 I_n \\
I_{n-2} & = & I_{n-1} + I_n & = & 3 I_n \\
I_{n-3} & = & I_{n-2} + I_{n-1} & = & 5 I_n \\
I_{n-4} & = & I_{n-3} + I_{n-2} & = & 8 I_n \\
 & \vdots & & \vdots & \\
I_1 & = & I_2 + I_3 & = & ? I_n
\end{array}
$$

Fibonacci Sequence : $\{1, 1, 2, 3, 5, 8, 13, 21, 34, 55, \ldots\}$

$$F_k = F_{k-1} + F_{k-2}, \; k = 2, 3, \ldots$$

$$F_0 = 1, F_1 = 1.$$

Fibonacci Search

$$\begin{array}{lclclcl}
I_{n+1} & = & I_n - I_{n+2} & = & 1 I_n & \equiv & F_0 I_n \\
I_n & = & I_{n+1} + I_{n+2} & = & 1 I_n & \equiv & F_1 I_n \\
I_{n-1} & = & I_n + I_{n+1} & = & 2 I_n & \equiv & F_2 I_n \\
I_{n-2} & = & I_{n-1} + I_n & = & 3 I_n & \equiv & F_3 I_n \\
I_{n-3} & = & I_{n-2} + I_{n-1} & = & 5 I_n & \equiv & F_4 I_n \\
I_{n-4} & = & I_{n-3} + I_{n-2} & = & 8 I_n & \equiv & F_5 I_n \\
 & \vdots & & \vdots & & & \\
I_k & = & I_{k+1} + I_{k+2} & = & F_{n-k+1} I_n & & \\
 & \vdots & & \vdots & & & \\
I_1 & = & I_2 + I_3 & = & F_n I_n & &
\end{array}$$

Note: After n iterations,

$$I_n = \frac{I_1}{F_n}$$

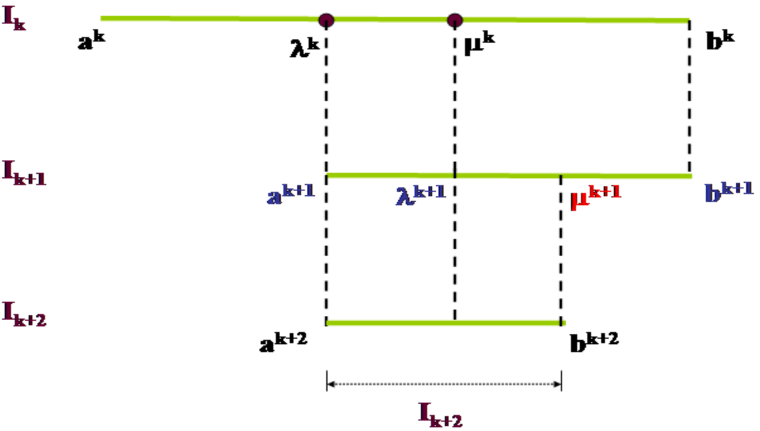
Fibonacci Search

- After n iterations,

$$I_n = \frac{I_1}{F_n}$$

  For example, after 10 iterations, $I_n = \frac{I_1}{89}$
- n should be known beforehand

Fibonacci Search



How to determine μ^{k+1} knowing a^{k+1}, b^{k+1}, λ^{k+1} and I_{k+1} ?

Note: Easy to find μ^{k+1} if I_{k+2} is known.

Fibonacci Search

Recall,

$$I_k = F_{n-k+1} I_n.$$

Therefore,

$$I_{k+2} = F_{n-k-1} I_n$$

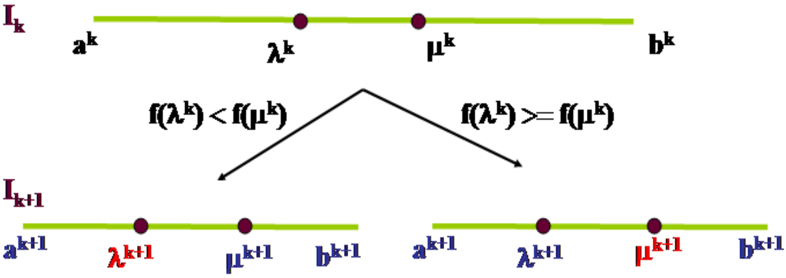
and

$$I_{k+1} = F_{n-k} I_n.$$

This gives,

$$I_{k+2} = \frac{F_{n-k-1}}{F_{n-k}} I_{k+1}.$$

Fibonacci Search



$\lambda^{k+1} = b^{k+1} - I_{k+2}$ $\qquad$ $\mu^{k+1} = a^{k+1} + I_{k+2}$

Only one function evaluation per iteration (after the first iteration)

The Fibonacci Search

- Consider, $\min_x \ (1/4)x^4 - (5/3)x^3 - 6x^2 + 19x - 7$
- Initial interval of uncertainty : $[-4, 0]$
- Required length of interval of uncertainty: 0.2
- Set n such that $F_n > \frac{4}{0.2} = 20$, $n = 7$

k	a^k	b^k	$b^k - a^k$
0	-4	0	4
1	-4	-1.52	2.48
2	-3.05	-1.52	1.53
3	-3.05	-2.11	0.94
4	-2.70	-2.11	0.59
5	-2.70	-2.34	0.36
6	-2.70	-2.47	0.23
7	-2.61	-2.47	0.14

Golden Section Search

- Fibonacci Search requires the number of iterations as input
- Golden section search : Ratio of two adjacent intervals is constant
$$\frac{I_k}{I_{k+1}} = \frac{I_{k+1}}{I_{k+2}} = \frac{I_{k+2}}{I_{k+3}} = \ldots = r$$
Therefore,
$$\frac{I_k}{I_{k+2}} = r^2$$
and
$$\frac{I_k}{I_{k+3}} = r^3.$$

Golden Section Search

Suppose, $I_k = I_{k+1} + I_{k+2}$. That is,

$$\frac{I_k}{I_{k+2}} = \frac{I_{k+1}}{I_{k+2}} + 1$$

This gives

$$r^2 = r + 1 \Rightarrow \; r = \frac{1+\sqrt{5}}{2} = 1.618034 \quad \text{(negative } r \text{ is irrelevant)}$$

Golden Ratio = 1.618034

Golden Section Search

- Every iteration is independent of n

$$\frac{I_k}{I_{k+1}} = \frac{I_{k+1}}{I_{k+2}} = \frac{I_{k+2}}{I_{k+3}} = \ldots = r$$

- Lengths of generated intervals,

$$\{I_1, I_1/r, I_1/r^2, \ldots\}$$

- After n function evaluations, $I_n^{GS} = I_1/r^{n-1}$

- For Golden Section search, $I_n^{GS} = I_1/r^{n-1}$
- For Fibonacci search, $I_n^F = I_1/F_n$
- When n is large,

$$F_n \equiv \frac{r^{n+1}}{\sqrt{5}}$$

  Therefore, $I_n^F \equiv \frac{\sqrt{5}}{r^{n+1}} I_1$

- $$\frac{I_n^G}{I_n^F} = \frac{r^2}{\sqrt{5}} \approx 1.17$$

- If the number of iterations is same, I_n^{GS} is larger than I_n^F by about 17%

Derivative-based Methods

Let $f : \mathbb{R} \rightarrow \mathbb{R}$

Unconstrained problem

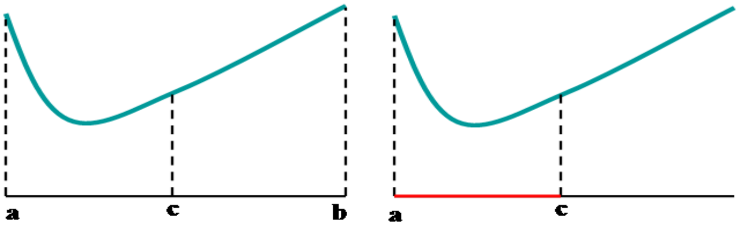
$$\min_{x \in \mathbb{R}} \quad f(x)$$

- Bisection method
  - Assumes $f \in \mathcal{C}^1$.
  - Interval of uncertainty, $[a, b]$, which contains the minimum of f needs to be provided.
  - Assumes that f is *unimodal* in $[a, b]$.
- Newton method
  - Assumes $f \in \mathcal{C}^2$.
  - Based on using the quadratic approximation of f at every iteration

Bisection Method

Assumption:

- $f \in \mathcal{C}^1$
- f is unimodal in the initial interval of uncertainty, $[a, b]$.



Idea: Compute $f'(c)$ where c is the midpoint of $[a, b]$

1. If $f'(c) = 0$, then c is a minimum point.
2. $f'(c) > 0 \Rightarrow [a, c]$ is the new interval of uncertainty
3. $f'(c) < 0 \Rightarrow [c, b]$ is the new interval of uncertainty

Bisection Method: Algorithm

1. **Initialization**: Initial interval of uncertainty $[a^1, b^1]$, $k = 1$. Let l be the allowable final level of uncertainty, choose smallest possible $n > 0$ such that $(\frac{1}{2})^n \leq \frac{l}{b^1 - a^1}$.
2. **while** $k \leq n$
3. $\quad c^k = \frac{a^k + b^k}{2}$
4. $\quad$ If $f'(c^k) = 0$, stop with c^k as an optimal solution.
5. $\quad$ If $f'(c^k) > 0$

   $\quad\quad a^{k+1} = a^k$ and $b^{k+1} = c^k$

   $\quad$ else

   $\quad\quad a^{k+1} = c^k$ and $b^{k+1} = b^k$

   $\quad$ endif
6. $\quad$ k:=k+1
7. **endwhile**
8. If $k = n + 1$, the solution is at the midpoint of $[a^n, b^n]$.

Bisection Method

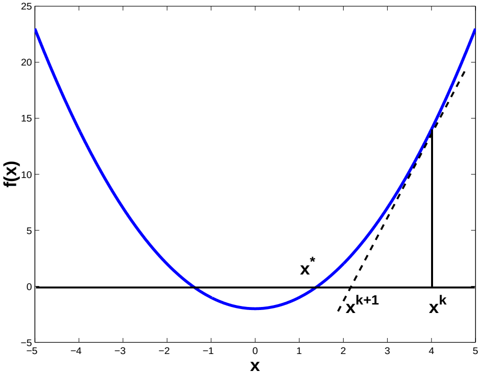
Bisection method

- requires initial interval of uncertainty
- converges to a minimum point within any degree of desired accuracy

- An iterative technique to find a root of a function
- *Problem*: Find an approximate root of the function,

$$f(x) = x^2 - 2.$$

- An iteration of Newton's method on $f(x) = x^2 - 2$.



Newton Method

- Consider the problem to minimize $f(x), \; x \in \mathbb{R}$
- *Assumption*: $f \in \mathcal{C}^2$.
- Idea:
  - At any iteration k, construct a quadratic model $q(x)$ which agrees with f at x^k up to second derivative,

$$q(x) = f(x^k) + f'(x^k)(x - x^k) + \frac{1}{2}f''(x^k)(x - x^k)^2$$

  - Estimate x^{k+1} by minimizing $q(x)$.
  - $q'(x^{k+1}) = 0 \Rightarrow f'(x^k) + f''(x^k)(x^{k+1} - x^k) = 0$
  - $x^{k+1} = x^k - \frac{f'(x^k)}{f''(x^k)}$
  - Repeat this process at x^{k+1}.

Newton Method: Algorithm

Consider the problem to minimize $f(x), f \in \mathcal{C}^2$
Need to find the roots of $g(x)(= f'(x))$.

1. **Initialization:** Choose initial point x^0, ϵ and set $k := 0$
2. **while** $|g(x^k)| > \epsilon$
3. $\quad x^{k+1} = x^k - \frac{g(x^k)}{g'(x^k)}$
4. $\quad k := k + 1$
5. **endwhile**
6. **Output**: x^k

Remarks

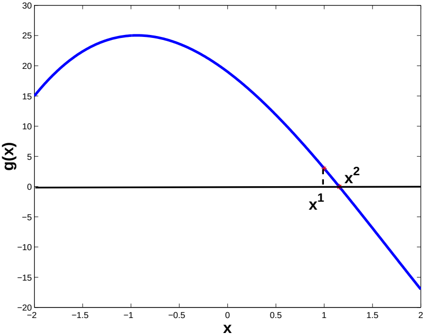
- Starting with an arbitrary initial point, Newton method does not converge to a stationary point
- If the starting point is *sufficiently close* to a stationary point, then Newton method converges.
- Useful when $g'(x^k) > 0$

Newton Method Iterations

Consider the problem,

$$\min_{x \in \mathbb{R}} \quad (1/4)x^4 - (5/3)x^3 - 6x^2 + 19x - 7$$

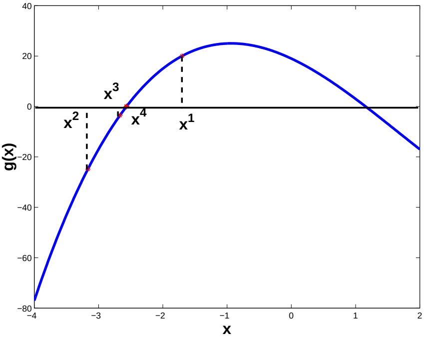
- $f(x) = (1/4)x^4 - (5/3)x^3 - 6x^2 + 19x - 7$
- $g(x) = x^3 - 5x^2 - 12x + 19$



Consider the problem,

$$\min_{x \in \mathbb{R}} \quad (1/4)x^4 - (5/3)x^3 - 6x^2 + 19x - 7$$

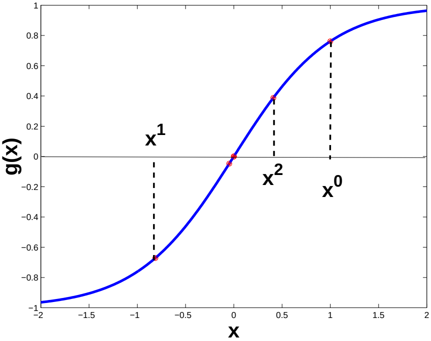
- $f(x) = (1/4)x^4 - (5/3)x^3 - 6x^2 + 19x - 7$
- $g(x) = x^3 - 5x^2 - 12x + 19$



Consider the problem,

$$\min_{x \in \mathbb{R}} \quad \log(e^x + e^{-x})$$

- $f(x) = \log(e^x + e^{-x})$
- $g(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$

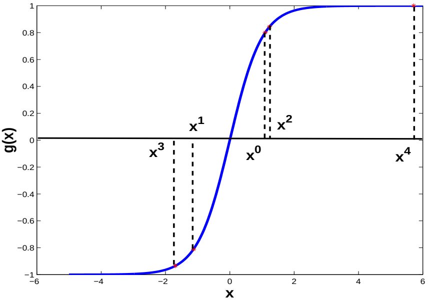


Newton Method Iterations

Consider the problem,

$$\min_{x \in \mathbb{R}} \quad \log(e^x + e^{-x})$$

- $f(x) = \log(e^x + e^{-x})$
- $g(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$



Newton method does not converge with this initialization of x^1