Numerical Optimization

Instructor : Sung Chan Jun

Week #3 : September 11, 2023 (Monday Class)

Course Syllabus (Tentative)

Calendar	Description	Remarks
1st week	*Introduction of optimization*	
2nd week	*Univariate Optimization*	
3rd week	Univariate Optimization	
4th week	Unconstrained Optimization	
5th week	Unconstrained Optimization	
6th week	Constrained Optimization, No Class	Oct. 2 (Temporary National Holiday)
7th week	Constrained Optimization, No Class	Oct. 9 (National Holiday)
8th week	Constrained Optimization, Midterm	Oct. 18 (Midterm)

Announcements

- Teaching Assistant (TA)
  - Dr. Cheolki Im (AI Graduate School)
    - Post-doc at Biocomputing Lab
    - E-mail: chim@gm.gist.ac.kr
    - Phone: 2266 (internal)
    - Office: DASAN Bldg. Room 505

Recall – Last Week

- Optimality Condition : Unconstrained Univariate
  - (Generalization of optimal conditions) Assume objective univariate function f(x) is at least n times continuously differentiable.

    Let $f'(x^*) = f''(x^*) = \dots = f^{(n-1)}(x^*) = 0$ & $f^{(n)}(x^*) \neq 0$. Then
    - If $f^{(n)}(x^*) > 0$ and n even, x^* is a local minimum.
- Multivariate Calculus
  - Differentiation of function $z = f(x, y) : R^2 \rightarrow R^1$
    - Partial differentiation (in general)

      $$\partial f(x, y)/\partial x = \lim_{h_x \rightarrow 0} [f(x + h_x, y) - f(x, y)]/h_x$$

      $$\partial f(x, y)/\partial y = \lim_{h_y \rightarrow 0} [f(x, y + h_y) - f(x, y)]/h_y$$

Recall – Last Week

- Multivariate Calculus
  - Differentiation of vector valued function

$$F : R^2 \rightarrow R^2 \text{ defined by } F(x, y) = (f_1(x, y), f_2(x, y))$$

    - Partial differentiation : $D_x F(x, y) = (\partial f_1(x, y)/\partial x \quad \partial f_2(x, y)/\partial x)^T$
    - Derivative matrix DF

$$\lim_{\mathbf{h}=(h_x\ h_y)\to\mathbf{0}} \left\| F(x+h_x, y+h_y) - F(x,y) - DF(x,y)(h_x\ \ h_y)^T \right\| / \left\| (h_x\ \ h_y) \right\| = 0$$

      - $DF(x, y) = \begin{pmatrix} \partial f_1/\partial x & \partial f_1/\partial y \\ \partial f_2/\partial x & \partial f_2/\partial y \end{pmatrix}$

Recall – Last Week

- Multivariate Calculus
  - Gradient (grad f, ∇f) : Let $f(\mathbf{x})$ be a scalar valued function $R^n \rightarrow R$.
    - $\nabla f(\mathbf{x}) = \nabla f(x_1, x_2, \ldots, x_n) = (\partial f/\partial x_1 \;\; \partial f/\partial x_2 \;\; \ldots \;\; \partial f/\partial x_n)^T$
    - Physical meaning : steepest increasing direction
  - Divergence (div F, $\nabla \cdot F$): Let $F = (f_1, f_2, \ldots, f_n) : R^n \rightarrow R^n$
    - $\nabla \cdot F = \partial f_1/\partial x_1 + \partial f_2/\partial x_2 + \ldots + \partial f_n/\partial x_n$
    - Physical meaning : rate of volume change per unit volume
  - 2nd derivative for multivariate $f(\mathbf{x}) = f(x_1, x_2)$: $R^2 \rightarrow R$
    - notation : $H(\mathbf{x}) = H(x_1, x_2)$ called 'Hessian'
    - definition : matrix $H(x_1, x_2)$ defined by

$$\begin{pmatrix} \partial^2 f/\partial x_1^2 & \partial^2 f/\partial x_1 \partial x_2 \\ \partial^2 f/\partial x_2 \partial x_1 & \partial^2 f/\partial x_2^2 \end{pmatrix}$$

Recall – Last Week

- Optimality Conditions : Unconstrained Multivariate
  - Assume objective function f(**x**) is at least twice-continuously differentiable.

> • (**NC**) Necessary condition for a local minimum
>
> 1. grad(f(**x**)) = 0
> 2. H(**x**) ("hessian") ≥ 0 i.e H(**x**) is positive semi-definite.
>
> • (**SC**) Sufficient condition for a local minimum
>
> 1. grad(f(**x**)) = 0
> 2. H(**x**) > 0 i.e H(**x**) is positive definite.

- A symmetric real matrix **A** ($n \times n$)
  - is said to be positive definite if $\mathbf{z}^T \mathbf{A} \mathbf{z} > 0$ (strictly positive) for every non-zero vector **z** of real numbers.

Recall – Last Week

- Univariate Optimization

  Minimize $f(x)$ on $x \in R$

  - Conventional strategy

    Due to optimality conditions,

    - first seek points x with $f'(x) = 0$ (stationary points).
    - then check the sign of $f''(x)$ at those points.

  - How to find zero of $f'(x)$? $\Rightarrow$ root finding

    - Conventional techniques for root finding
      - Method of bisection, Newton's method
      - Secant method, Regula falsi method

Optimality conditions for univariate problem

- Necessary condition for a local minimum

  $f'(x^*) = 0$ & $f''(x^*) \geq 0$

- Sufficient condition for a local minimum

  $f'(x^*) = 0$ & $f''(x^*) > 0$

Recall – Last Week

- Univariate Optimization: Root Finding - Method of Bisection
  - Interval [a, b] is given such that f(a)f(b) < 0.
  - Step 1. compute f(x) at the midpoint x = (a + b)/2.
  - Step 2. if f(x) = 0 or (b – a) < TOL, then terminate.

    if f(x)f(a) < 0, then b := x,

    else a: = x.
  - Step 3. Go to Step 1.

  Midpoint is one idea. Other strategies may be applicable in a similar manner.
  - Randomly chosen interior point
  - Any interior point

  - Guaranteed to converge to zero; too slow (convergence rate ½)

- The sequence $\{x_k\}$ converges with order r to x^*.
  - $\exists$ a constant c > 0 and integer N such that $\|x_{k+1} - x^*\| \le c\|x_k - x^*\|^r$
    - r = 1, linear convergence; r = 2, quadratic convergence; r > 1, superlinear convergence

Recall – Last Week

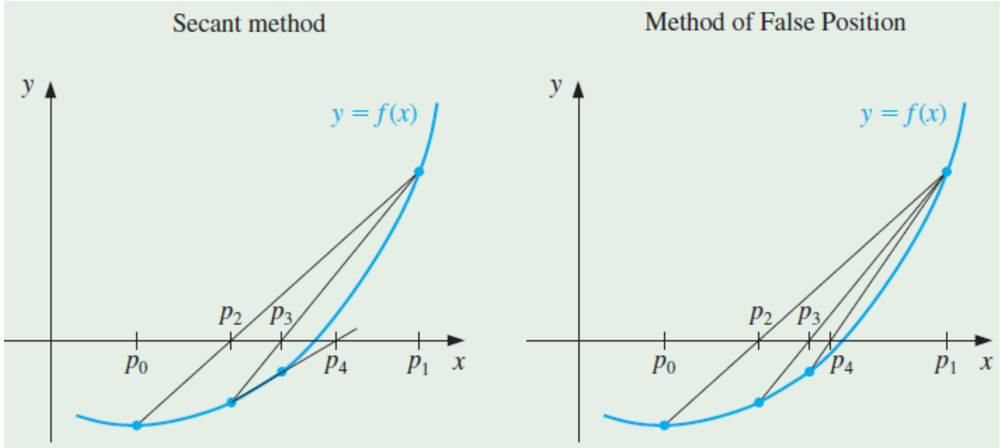
- Univariate Optimization: Root-finding Methods
  - Newton's method
    - Approximate f(x) by tangent line at the given point.
    - $x_{k+1} = x_k - f(x_k)/f'(x_k)$
    - Very fast converging (r = 2); convergence depending on initial guess; not working when $f'(x_k)$ is small; derivative is required
  - Secant method (method of linear interpolation)
    - Computing f'(x) is very expensive and impossible to compute in some cases.
    - Approximating tangent line by straight line passing two recent iterates
    - $x_{k+1} = x_k - [(x_k - x_{k-1})/(f(x_k) - f(x_{k-1})]f(x_k)$
    - rapid convergent (roughly rate r = 1.6180); divergent if straight line approximation is extrapolation

Recall – Last Week

- Univariate Optimization : Root-finding Methods
  - More Consideration on Secant Methods
    - How to approximate tangent line
      - There are many ways to do it
      - Two point approximation
        - $f'(x_n) \approx (f(x_n) - f(x_{n-1}))/(x_n - x_{n-1})$; $f'(x_n) \approx (f(x_n) - f(x_{n-2}))/(x_n - x_{n-2})$
      - Three point approximation
        - $f'(x_n) \approx \alpha(f(x_n) - f(x_{n-1}))/(x_n - x_{n-1}) + (1-\alpha)(f(x_n) - f(x_{n-2}))/(x_n - x_{n-2})$
      - Richardson's extrapolation: all points on axis are even spaced, that is, h is fixed
        - 3-point approximation, 4-point approximation
        - 6-point approximation

Recall – Last Week

- Univariate Optimization: Root-finding Methods
  - Regular falsi method (method of false position)
    - Consider the given interval I_k = [a, b] such that f(a)f(b) < 0.
    - Apply a secant method with two initial points a & b. Find a point x_{k+1} intersecting with x-axis and a secant.
    - Choose updated interval as follows:
      - $I_{k+1} = [a, x_{k+1}]$, if f(a) and $f(x_{k+1})$ have different signs, or $I_{k+1} = [x_{k+1}, b]$, otherwise
    - Keep doing in the same manner until termination criterion is satisfied.



Recall – Last Week

- Univariate Optimization : Root-finding Methods
  - Principles of root-finding methods
    - Method of bisection : bracketing, that is, interval is used.
    - Newton's method : straight line is used.
    - Secant method : straight line is used.
    - Regula falsi method : straight line and bracketing are used.
  - Why are straight lines mainly used in root finding techniques?
    - Straight line (1^{st} order polynomial) is the simplest shape in approximation.
    - Finding root (intersecting point with x-axis) of straight line is very easy.
  - How about other approaches in place of a straight line?
    - More complex shape may be applicable in the same context.
    - Curve (2^{nd} order or higher order polynomial) may be possible.

Recall – Last Week

- Univariate Optimization : Root-finding Methods
  - More advanced root-finding approaches
    - Higher order polynomial approximation
      - Higher order polynomials (quadratic, cubic…) are used for approximation of original function f(x).
        - That would be much rapidly convergent.
        - Seeking the zero point of it is more difficult than a straight line.
    - Rational function approximation (rational interpolation)
      - Approximate f(x) by rational function of the form $f_{rat}(x) = \dfrac{x - c}{d_0 + d_1 x + d_2 x^2}$
        - d_0, d_1, d_2, c are chosen so that the function value and derivatives of $f_{rat}(x)$ agree with those of f(x) at two points.
      - This approximation is easy to find zero point, which is just 'c'.

Univariate Optimization

Minimize f(x) on $x \in R$

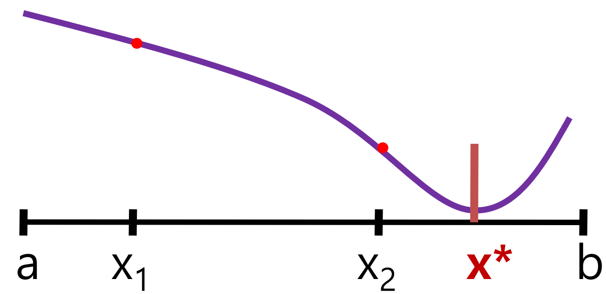
- When f(x) is differentiable
  - Univariate optimization comes to

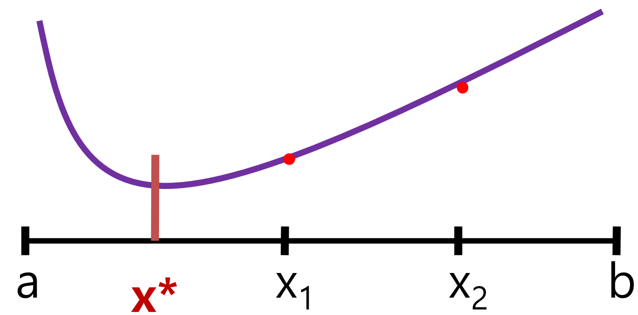
    finding root problem : $f'(x) = 0$.

- When f(x) is not differentiable
  - How can we solve the optimization problem?
    - Consider methods using function evaluations only

Univariate Optimization:
Unimodality

- Unimodality
  - f(x) is unimodal in [a, b] if there exists a unique $x^* \in [a, b]$ such that for any $x_1, x_2 \in [a, b]$ and $x_1 < x_2$,
    - If $x_2 < x^*$ then $f(x_1) > f(x_2)$
    - If $x_1 > x^*$ then $f(x_1) < f(x_2)$





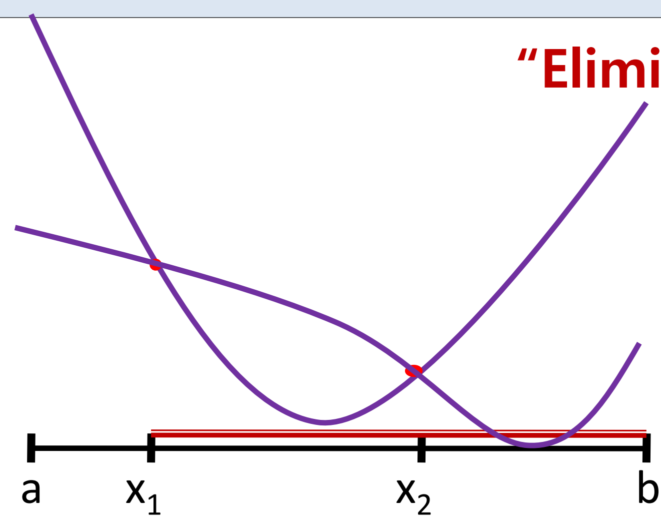
  - If f is unimodal in the given interval, it exists a strong local minimum in it.

Univariate Optimization:
Unimodality

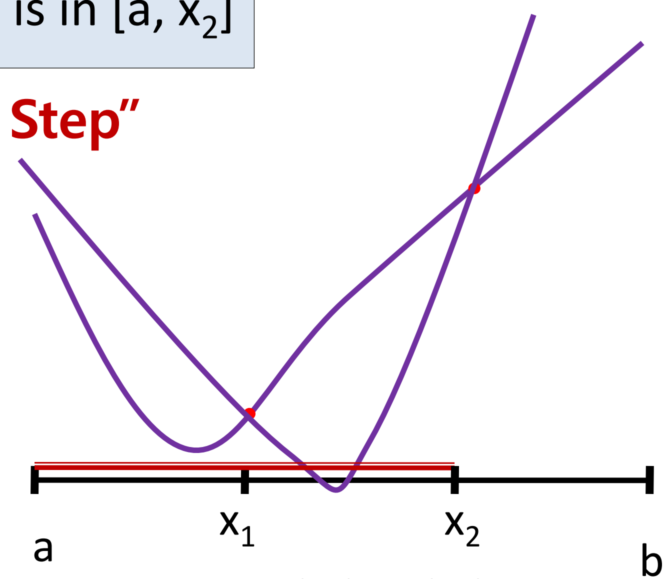
- When unimodal f(x) is evaluated at two interior points x_1 and x_2 ($x_1 < x_2$) for given interval [a, b], then

> - if $f(x_1) > f(x_2)$, then a minimum is in $[x_1, b]$
> - Otherwise (if $f(x_1) \leq f(x_2)$), a minimum is in $[a, x_2]$

"Elimination Step"



When $f(x_1) > f(x_2)$



When $f(x_1) \leq f(x_2)$

Univariate Optimization:
Unimodality

- Let f be unimodal and $x^* \in [a, b]$ be minimum.
  - By elimination step, (letting $[a_0, b_0] = [a, b]$)
    - 1^{st} step : choose interior points α_1, β_1 in $[a_0, b_0]$ such that $\alpha_1 < \beta_1$

      $x^* \in [a_0, \beta_1] \subset [a_0, b_0]$ when $f(\alpha_1) < f(\beta_1)$

      $x^* \in [\alpha_1, b_0] \subset [a_0, b_0]$ when $f(\alpha_1) > f(\beta_1)$

      $x^* \in$ smaller interval $[a_1, b_1]$
    - 2^{nd} step : choose interior points α_2, β_2 in $[a_1, b_1]$ such that $\alpha_2 < \beta_2$

      $x^* \in [a_1, \beta_2] \subset [a_1, b_1] \subset [a_0, b_0]$ when $f(\alpha_2) < f(\beta_2)$

      $x^* \in [\alpha_2, b_1] \subset [a_1, b_1] \subset [a_0, b_0]$ when $f(\alpha_2) > f(\beta_2)$

      $x^* \in$ smaller interval $[a_2, b_2]$
    - 3^{rd} step …

      $x^* \in$ smaller interval $[a_3, b_3]$

      ⋮

      ⋮

    - n^{th} step

      $x^* \in$ smaller interval $[a_n, b_n]$

Univariate Optimization: Unimodality

- Let f be unimodal and $x^* \in [a, b]$ be minimum.
  - By elimination step, (letting $[a_0, b_0] = [a, b]$)
    - So finally, we got the following bracket method:

$$[a_0, b_0] \supset [a_1, b_1] \supset [a_2, b_2] \supset [a_3, b_3] \supset \ldots \supset [a_n, b_n] \supset \ldots$$

sufficiently reduced

sufficiently reduced

sufficiently reduced

    - Whether or not this bracket method successfully works (that is, eventually it approaches the solution) depends on how to choose interior points.

Univariate Optimization:
Unimodality

Assume f(x) is unimodal.

- To efficiently reduce the interval of uncertainty by elimination step, we should choose two interior points every iteration.
- How to find two interior points?
  - Definitely, there are many ways to choose them
  - Two efficient ways to consider
    - Fibonacci search
    - Golden section search

Sequence of Numbers

- Look at the following number sequence

  1, 1, 2, 3, 5, 8, 13, 21, 34, 55

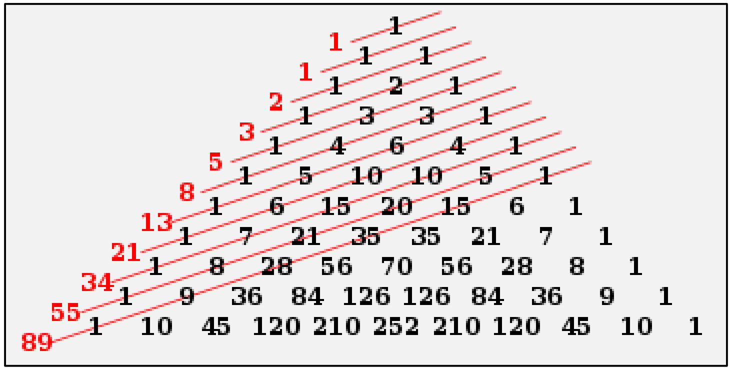
- What should be the next number?

Fibonacci Numbers

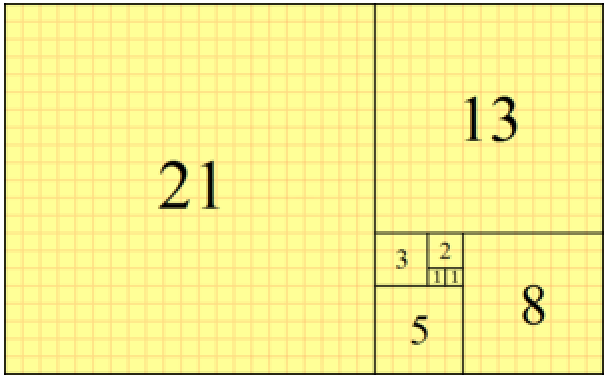
- Integer sequences generated by the following recurrence relation

$$\begin{cases} F_0 = F_1 = 1 \\ F_k = F_{k-1} + F_{k-2} \end{cases}$$

- Thus, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, ….



Pascal's triangle



https://en.wikipedia.org/

Univariate Optimization:
Fibonacci Search

- Fibonacci search on [a, b]

S1. Assume N function evaluations are possible.

S2. Generate Fibonacci numbers $\{F_0, F_1, F_2, \ldots, F_N\}$ such that $F_0 = F_1 = 1$, $F_k = F_{k-1} + F_{k-2}$.

S3. Choose two interior points x_1 and x_2 ($x_1 \le x_2$, let $L := b - a$) as follows:

$$x_1 = a + F_{N-2}/F_N * L = a\, F_{N-1}/F_N + b\, F_{N-2}/F_N$$

$$x_2 = b - F_{N-2}/F_N * L = a\, F_{N-2}/F_N + b\, F_{N-1}/F_N$$

Internally dividing points of [a, b]
x_1 = ratio $F_{N-2} : F_{N-1}$
x_2 = ratio $F_{N-1} : F_{N-2}$

S4. Compute $f(x_1)$ & $f(x_2)$. A new reduced interval $[a_{new}, b_{new}]$ is generated by elimination step.

S5. Set $N := N - 1$, $a := a_{new}$, $b := b_{new}$.

S6. Go to **S1** and repeat this until $N = 1$.

Univariate Optimization:
Fibonacci Search

- Example

Minimize |x - 0.3| on [0, 1] using Fibonacci search with N=5 function evaluations.

- N = 5, [a, b] = [0, 1], L = b - a = 1, $\{F_0, F_1, F_2, F_3, F_4, F_5\} = \{1, 1, 2, 3, 5, 8\}$

1st iteration

- $\boxed{x_1=a+F_{N-2}/F_N*L=F_3/F_5=3/8}$
- $x_2=b-F_{N-2}/F_N*L=1-F_3/F_5=5/8$
- $f(x_1)=f(3/8)=0.075$, $f(x_2)=f(5/8)=0.325$
- interval of uncertainty (reduced interval) : [0, 5/8], N=4

2nd iteration

- $\boxed{x_1=F_2/F_4*5/8=1/4}$
- $\boxed{x_2=5/8-F_2/F_4*5/8=3/8}$
- $f(x_1)=f(1/4)=0.05$, $f(x_2)=f(3/8)=0.075$
- interval of uncertainty : [0, 3/8], N=3

3rd iteration

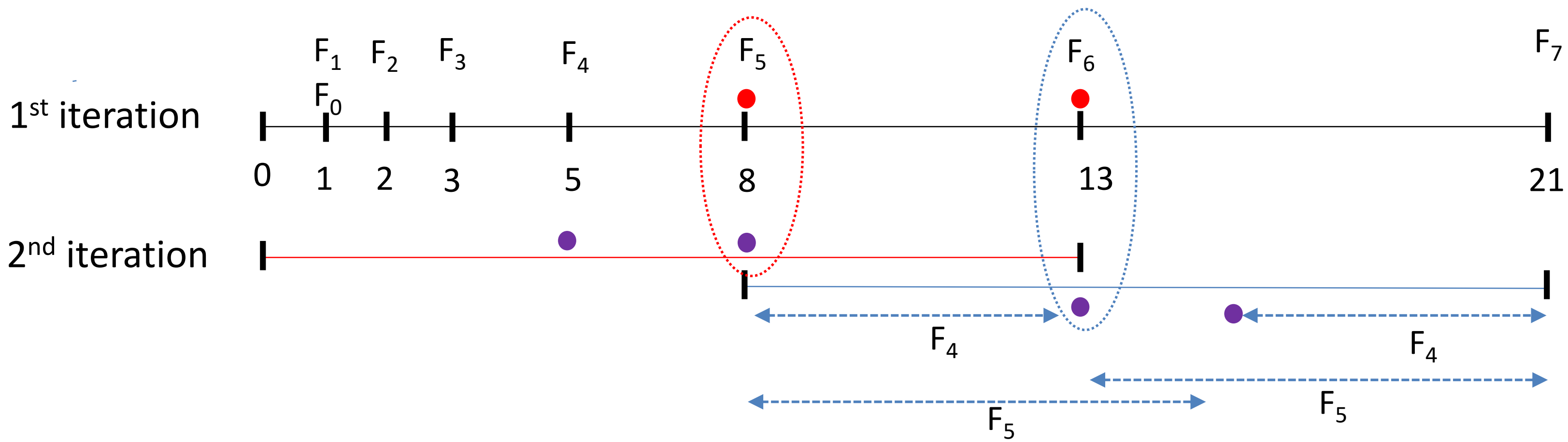
- $x_1=F_1/F_3*3/8=1/8$
- $\boxed{x_2=3/8-F_1/F_3*3/8=1/4}$
- $f(x_1)=f(1/8)=0.175$, $f(x_2)=f(1/4)=0.05$
- interval of uncertainty is [1/8, 3/8], N=2

4nd iteration

- $\boxed{x_1=1/8+F_0/F_2*1/4=1/4}$
- $x_2=3/8-F_0/F_2*1/4=1/4$ (modified $1/4+\delta$)
- $f(x_1)=f(1/4)=0.05$, $f(x_2)=f(1/4+\delta)=0.05-\delta$
- interval of uncertainty is [1/4, 3/8], N=1

Univariate Optimization:
Fibonacci Search

- Due to Fibonacci sequences, every step requires just one more function evaluation except for the first step.
  - For $[a, b] = [0, F_N]$ and $L = F_N$,



Univariate Optimization: Fibonacci Search

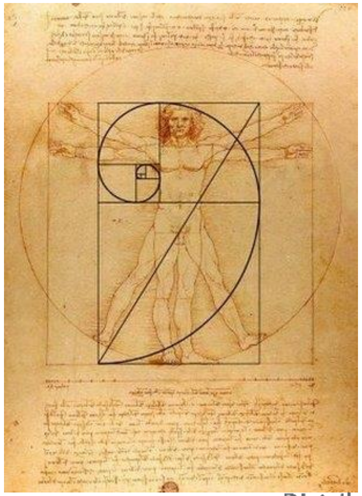
- Final interval of uncertainty (N evaluations) : $1/F_N*(b - a)$
  - If tolerance of error ε is given, we can estimate N:
    - When $1/F_N*(b - a) < \varepsilon$, find the smallest N such that $F_N > (b - a)/\varepsilon$.
- Cons
  - Require to store the Fibonacci numbers
  - Is not easy to apply for the case when termination criterion requires.

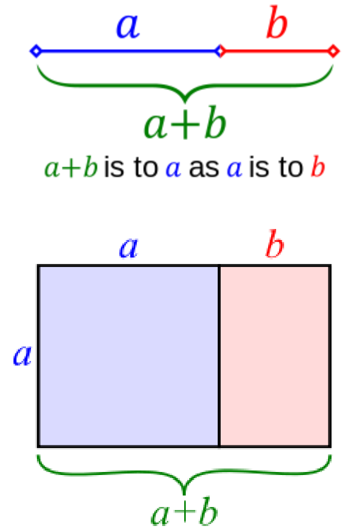
Univariate Optimization:
Fibonacci Search

- Problem : Consider Minimize $|x - 0.65|$ on $[0, 1]$.
  - Use Fibonacci search with $N = 8$ function evaluations and give the interval of uncertainty.
  - Infer how the length of interval of uncertainty is behaved.

Golden Section Ratio

- Two quantities are in the golden ratio if their ratio is the same as the ratio of their sum to the larger of the two quantities.
- Other names
  - golden mean ,extreme and mean ratio, medial section, divine proportion,
  - divine section, golden proportion, golden cut, golden number

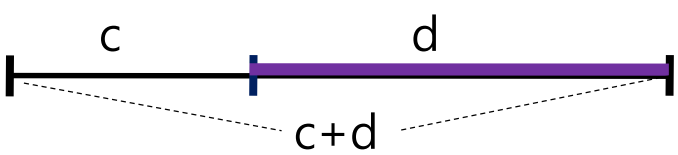




Univariate Optimization:
Golden Section Search

- Golden section search
  - Two interior points on [0, 1] are chosen as

    τ and $1-\tau$ such that $\tau > 1-\tau$.
  - By elimination step, we can get the reduced interval of length τ.
  - Keeping N times in this way, the final interval of uncertainty is length of τ^N.



  - How to determine τ?
    - Golden section ratio (τ)
      - $\tau = d/(c+d) = c/d$

$$\tau^2 + \tau - 1 = 0$$

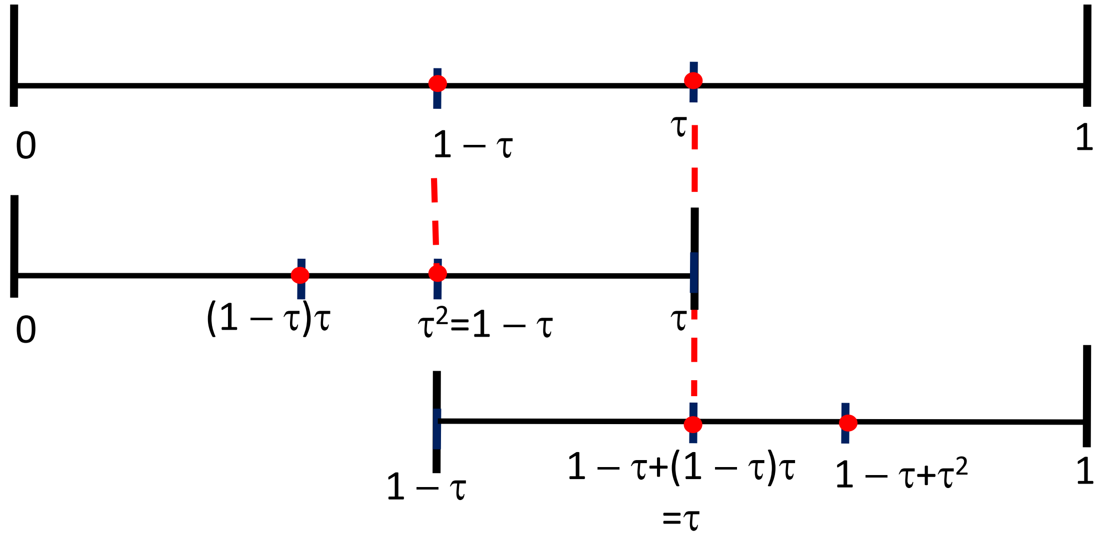
$$\tau = \frac{-1+\sqrt{5}}{2} \approx 0.6180$$

Univariate Optimization:
Golden Section Search

- Golden section search is a limiting case of Fibonacci search.

$$\lim_{k\to\infty}\frac{F_{k-1}}{F_k}=\tau$$

- It keeps good property of Fibonacci search
  - it requires just one additional function evaluation every step after 1st step.



$$\tau^2+\tau-1=0$$

Univariate Optimization:
Golden Section Search

- Final interval of uncertainty (length of interval)
  - $\tau^{N-1}*(b - a)$, for N function evaluations
- It is easy to answer how many function evaluation is needed to yield the given accuracy (tolerance of error ε).
  - If ε is given as acceptable error bound, then $\tau^{N-1}*(b - a) \le \varepsilon$ should be satisfied. Finally, at least N ($\ge 1 + \log[\varepsilon/(b - a)]/\log\tau$) steps are required.

Univariate Optimization:
Search Algorithms

Fibonacci Search	Golden Section Search
▪ Use Fibonacci Sequences. ▪ Pros Every step requires one function evaluation only. ▪ Cons ✓ Require to store the Fibonacci numbers. ✓ not easy to apply for the case when termination criterion requires. ▪ Final length of interval $1/F_N$*(b–a) (after N function evaluations)	▪ Use Golden Section Ratio. ▪ Pros ✓ Every step requires one function evaluation only. ✓ Easily estimate how many iterations are needed to get the given accuracy. ▪ Final length of interval τ^{N-1}*(b–a) (after N function evaluations) ▪ This is a limiting case of Fibonacci search. $\boxed{\lim_{k\to\infty} \frac{F_{k-1}}{F_k} = \tau}$

Univariate Optimization:
Seeking bound

- How to find initial interval [a, b] for a unimodal function f(x)?
  - One of possible ideas

S1. Set randomly initial point x_0, step size $d_0 > 0$

S2. Evaluate $f_- := f(x_0 - d_0)$, $f_0 := f(x_0)$, $f_+ := f(x_0 + d_0)$

S3. If $f_- \geq f_0 \geq f_+$, then set $d := d_0$, $x_{-1} := x_0 - d_0$, $x_1 := x_0 + d_0$

If $f_- \leq f_0 \leq f_+$, then set $d := -d_0$, $x_{-1} := x_0 + d_0$, $x_1 := x_0 - d_0$

If $f_- \geq f_0 \leq f_+$, then set $[a, b] := [x_0 - d_0, x_0 + d_0]$ and stop.

S4. For k=1,2,… $x_{k+1} = x_k + \boxed{2^k d}.$ Many ideas exist

- If $f(x_{k+1}) \geq f(x_k)$ & $d > 0$, then set $[a, b] := [x_{k-1}, x_{k+1}]$ and stop.
- If $f(x_{k+1}) \geq f(x_k)$ & $d < 0$, then set $[a, b] := [x_{k+1}, x_{k-1}]$ and stop.

Univariate Optimization

Minimize f(x) on $x \in R$

- When f(x) is not differentiable
  - Consider methods using function evaluations only
    - Fibonacci Search, Golden Section Search
  - What other methods?
- When f(x) is differentiable
  - Univariate optimization comes to finding root problem : $f'(x) = 0$.
    - Method of Bisection, Newton's, Secant, Regular falsi
  - What other methods?

Univariate Optimization:
Interpolation methods

- Assume f(x) is unimodal and twice continuously differentiable on [a, b].
  - Newton's method
    - Let f be twice continuously differentiable.
    - f ≈ quadratic interpolation function f^
    - By Taylor's expansion, with $f(x_k)$, $f'(x_k)$ and $f''(x_k)$

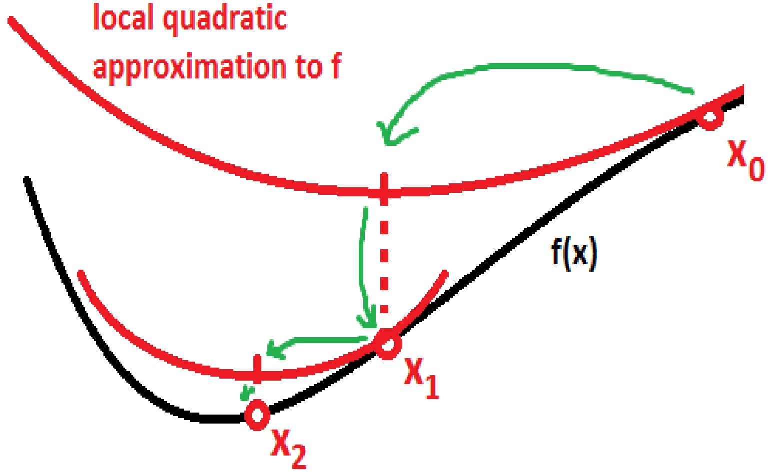
$$f^{\wedge}(x) = f(x_k) + f'(x_k)(x - x_k) + \frac{1}{2} f''(x_k)(x - x_k)^2$$

    - Find its minimum and call it x_{k+1} , then

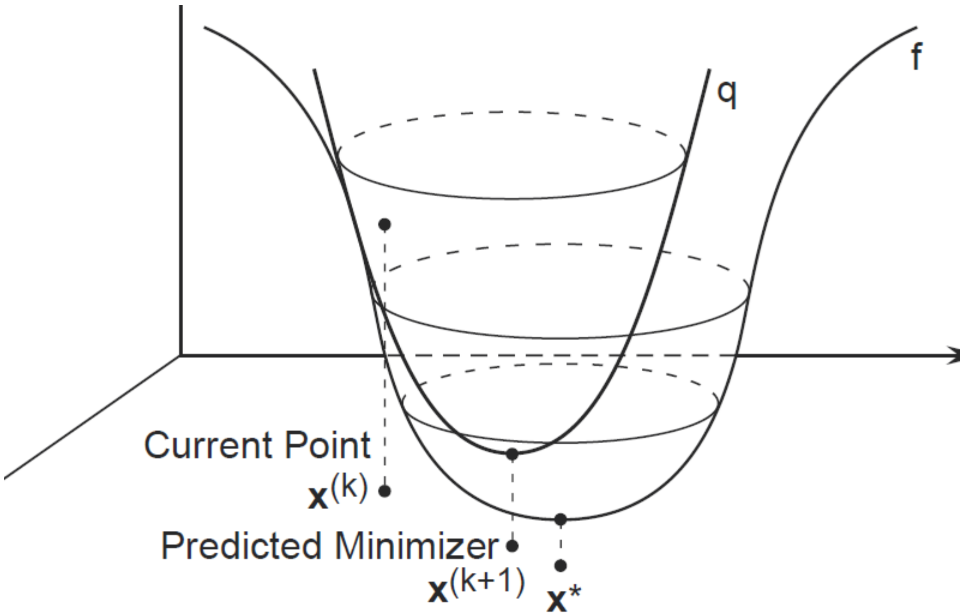
$$x_{k+1} = x_k - f'(x_k)/f''(x_k)$$

Univariate Optimization:
Interpolation methods

- Newton's Method in Optimization



1-dimensional
problem



2-dimensional
problem

Univariate Optimization:
Interpolation Methods

- Assume f(x) is unimodal and continuous on [a, b].
  - Quadratic Interpolation without derivatives
    - Set interval to [a, b] and midpoint c:=(a+b)/2.
    - Evaluate f at three points : (a, f(a)), (b, f(b)), (c, f(c)).
    - f ≈ quadratic function passing through three points, find its minimum x:

$$x = \frac{f(a)(b^2 - c^2) + f(b)(c^2 - a^2) + f(c)(a^2 - b^2)}{2[f(a)(b-c) + f(b)(c-a) + f(c)(a-b)]}$$

    - Update the interval and do the same way again.

Univariate Optimization:
Safeguarded methods

- Assume f(x) is unimodal on [a, b]
  - Mixed method (reliable + rapid)
    - Reliable and guaranteed method
      - Fibonacci search
      - Golden Section search
    - Rapidly convergent method
      - Quadratic interpolation, and etc.