System of Nonlinear equations

N nonlinear equations in n unknowns

$$f_1(x_1, x_2, x_3, x_4, ,....x_n)$$
$$f_2(x_1, x_2, x_3, x_4, ,....x_n)$$
$$....$$
$$f_n(x_1, x_2, x_3, x_4, ,....x_n)$$

Sometimes you can use a fixed point iteration or an iteration similar to Gauss Seidel Iteration for linear equations.

Newton Raphson method for n variables:

Expand the functions of n variables into Taylor Series:

Newton Raphson Method for one variable:

$$f_{1,i+1} = f_{1,i} + \frac{\partial f_{1,i}}{\partial x_1}(x_{1,i+1} - x_{1,i}) + \frac{\partial f_{1,i}}{\partial x_2}(x_{2,i+1} - x_{2,i}) + ... + \frac{\partial f_{1,i}}{\partial x_1}(x_{n,i+1} - x_{n,i}) + HOTerms = 0$$

$$f_{2,i+1} = f_{2,i} + \frac{\partial f_{2,i}}{\partial x_1}(x_{1,i+1} - x_{1,i}) + \frac{\partial f_{2,i}}{\partial x_2}(x_{2,i+1} - x_{2,i}) + ... + \frac{\partial f_{2,i}}{\partial x_1}(x_{n,i+1} - x_{n,i}) + HOTerms = 0$$

$$....$$

$$f_{n,i+1} = f_{n,i} + \frac{\partial f_{n,i}}{\partial x_1}(x_{1,i+1} - x_{1,i}) + \frac{\partial f_{n,i}}{\partial x_2}(x_{2,i+1} - x_{2,i}) + ... + \frac{\partial f_{n,i}}{\partial x_1}(x_{n,i+1} - x_{n,i}) + HOTerms = 0$$

$$\frac{\partial f_{1,i}}{\partial x_1} x_{1,i+1} + \frac{\partial f_{1,i}}{\partial x_2} x_{2,i+1} + ... + \frac{\partial f_{1,i}}{\partial x_n} x_{n,i+1} = -f_{1,i} + \frac{\partial f_{1,i}}{\partial x_1} x_{1,i} + \frac{\partial f_{1,i}}{\partial x_2} x_{2,i} .... + \frac{\partial f_{1,i}}{\partial x_n} x_{n,i}$$

$$\frac{\partial f_{2,i}}{\partial x_1} x_{1,i+1} + \frac{\partial f_{2,i}}{\partial x_2} x_{2,i+1} + ... + \frac{\partial f_{2,i}}{\partial x_n} x_{n,i+1} = -f_{2,i} + \frac{\partial f_{2,i}}{\partial x_1} x_{1,i} + \frac{\partial f_{2,i}}{\partial x_2} x_{2,i} .... + \frac{\partial f_{2,i}}{\partial x_n} x_{n,i}$$

$$....$$

$$\frac{\partial f_{n,i}}{\partial x_1} x_{1,i+1} + \frac{\partial f_{n,i}}{\partial x_2} x_{2,i+1} + ... + \frac{\partial f_{n,i}}{\partial x_n} x_{n,i+1} = -f_{n,i} + \frac{\partial f_{n,i}}{\partial x_1} x_{1,i} + \frac{\partial f_{n,i}}{\partial x_2} x_{2,i} .... + \frac{\partial f_{n,i}}{\partial x_n} x_{n,i}$$

$$J_i x_{i+1} = B_i$$

$$J_i = \begin{bmatrix} \frac{\partial f_1}{dx_1}_i & \frac{\partial f_1}{dx_2}_i & \cdots & \frac{\partial f_1}{dx_n}_i \\ \frac{\partial f_2}{dx_1}_i & \frac{\partial f_2}{dx_2}_i & .. & \frac{\partial f_2}{dx_n}_i \\ \cdots & \cdots\cdot & \cdots & \cdots \\ \frac{\partial f_n}{dx_1}_i & \frac{\partial f_n}{dx_2}_i & .. & \frac{\partial f_n}{dx_n}_i \end{bmatrix}$$

$$B_i = \begin{bmatrix} -f_{1,i} + \frac{\partial f_1}{\partial x_1}_i x_{1,i} + \frac{\partial f_1}{\partial x_2}_i x_{2,i} \cdots + \frac{\partial f_1}{\partial x_n}_i x_{n,i} \\ -f_{2,i} + \frac{\partial f_2}{\partial x_1}_i x_{1,i} + \frac{\partial f_2}{\partial x_2}_i x_{2,i} \cdots + \frac{\partial f_2}{\partial x_n}_i x_{n,i} \\ \cdots\cdots\cdot \\ -f_{2,i} + \frac{\partial f_n}{\partial x_1}_i x_{1,i} + \frac{\partial f_n}{\partial x_2}_i x_{2,i} \cdots + \frac{\partial f_n}{\partial x_n}_i x_{n,i} \end{bmatrix}$$

(J is known as the Jacobian Matrix)

Solve for x_{i+1} using any one of the methods discussed and repeat till there is convergence or

$$J_i \delta x_{i+1} = C_i$$

$$J_i = \begin{bmatrix} \frac{\partial f_1}{dx_{1\,i}} & \frac{\partial f_1}{dx_{2\,i}} & \cdots & \frac{\partial f_1}{dx_{n\,i}} \\ \frac{\partial f_2}{dx_{1\,i}} & \frac{\partial f_2}{dx_{2\,i}} & .. & \frac{\partial f_2}{dx_{n\,i}} \\ \cdots & .... & \cdots & \cdots \\ \frac{\partial f_n}{dx_{1\,i}} & \frac{\partial f_n}{dx_{2\,i}} & .. & \frac{\partial f_n}{dx_{n\,i}} \end{bmatrix}$$

$$C_i = \begin{bmatrix} -f_{1,i} \\ -f_{2,i} \\ .... \\ -f_{2,i} \end{bmatrix}$$

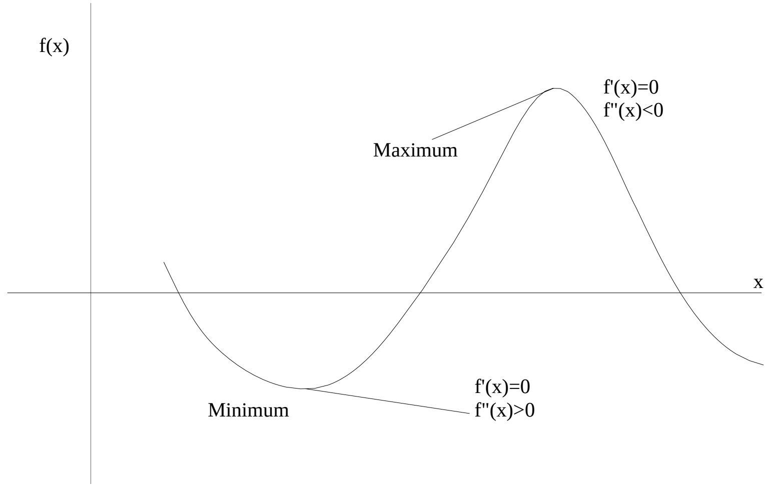
Where

$$x_{i+1} = x_i + \delta x_{i+1}$$

Repeat Until:

$$\varepsilon_a = abs\left|\frac{\delta x_{i+1}}{x_{i+1}}\right| * 100 < \varepsilon_s$$

Determination of the maximum (or minimum) of a function



$$x = [x_1, x_2, x_3....., x_n]$$

In General:

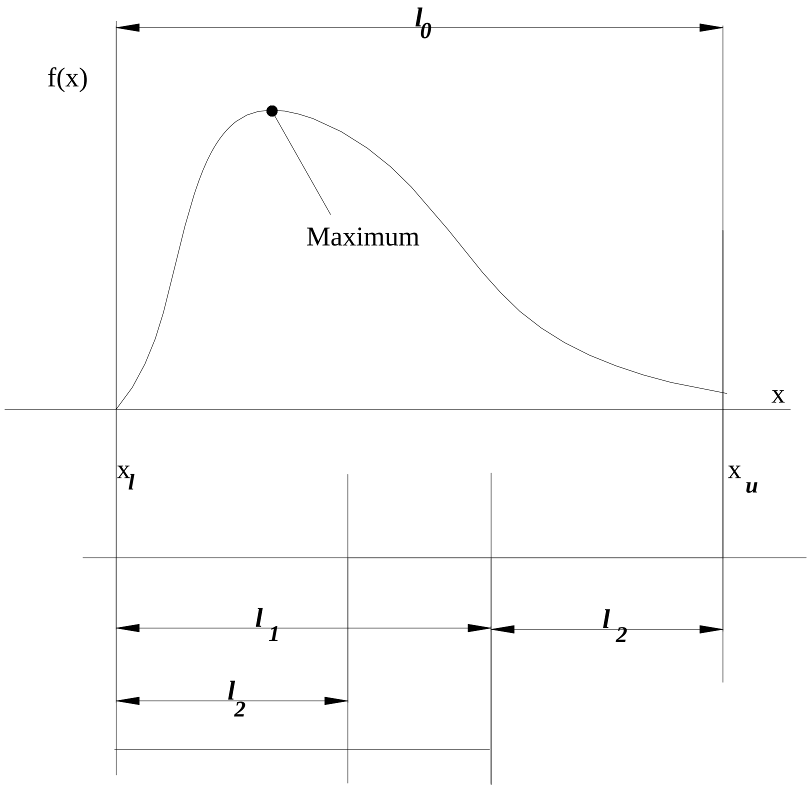
Find ***x*** which minimizes f(***x***)

Subject to:

$$d_i(x) \leq 0$$ (i=1,2,...m) Inequality Constraints

$$e_i(x) = b_i$$ (i=1,2,...p) Equality Constraints

Golden Section Search



$$\ell_0 = \ell_1 + \ell_2$$

$$\frac{\ell_1}{\ell_0} = \frac{\ell_2}{\ell_1}$$

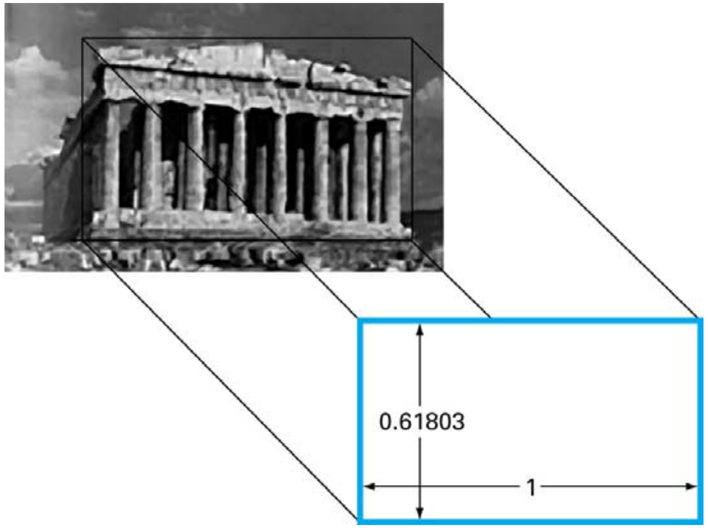
$$\frac{\ell_1}{\ell_1 + \ell_2} = \frac{\ell_2}{\ell_1}$$

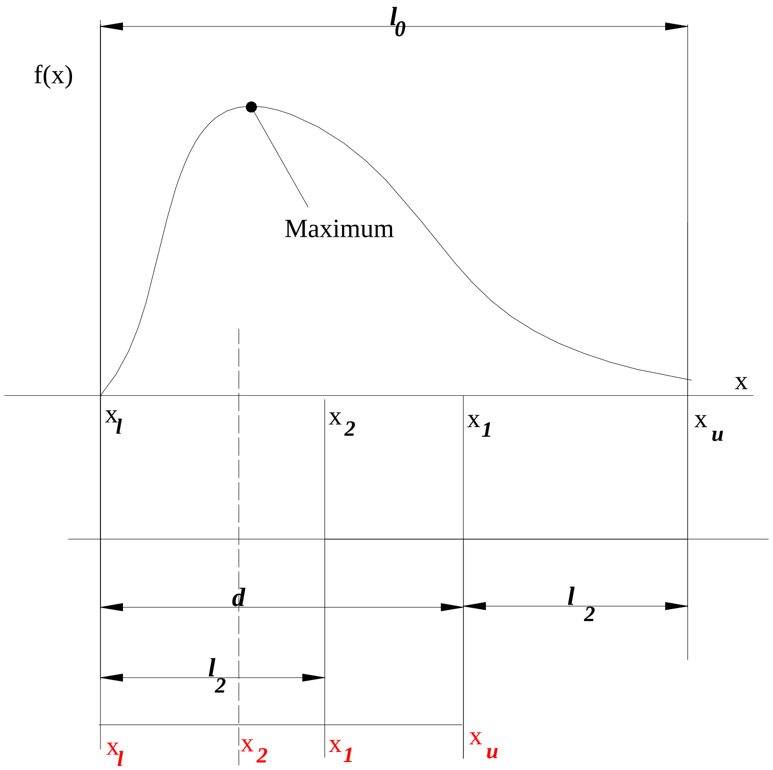
$$R = \frac{\ell_2}{\ell_1}$$

$$1 + R = \frac{1}{R}$$

$$R^2 + R - 1 = 0$$

$$R = \frac{-1 + \sqrt{1 - 4(-1)}}{2} = \frac{\sqrt{5} - 1}{2} = 0.61803$$





$$d = \frac{\sqrt{5}-1}{2}(x_u - x_L) = R(x_u - x_L)$$
$$x_1 = x_L + d$$
$$x_2 = x_u - d$$

1. If $f(x_1) > f(x_2)$ then

   $x_L = x_2$,

   $x_2 = x_1$,

   $x_1 = x_L + R(x_u - x_L)$

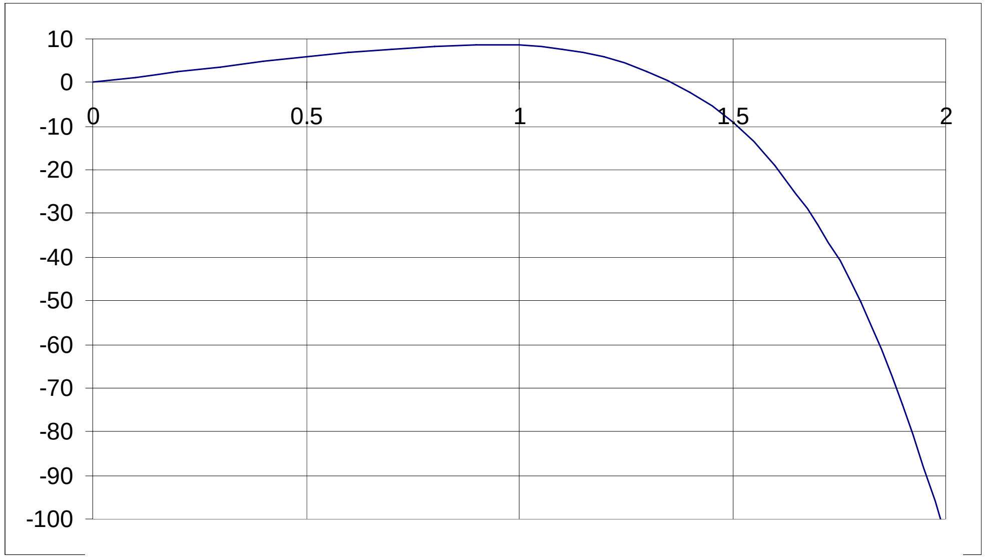
2. If $f(x_1) < f(x_2)$ then

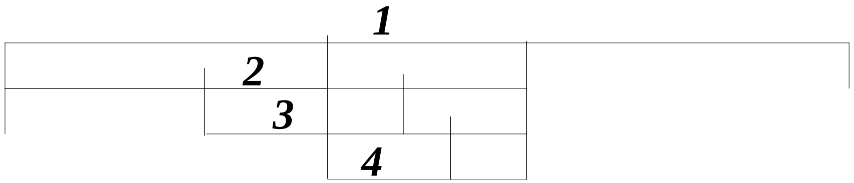
   $x_u = x_1$,

   $x_1 = x_2$,

   $x_2 = x_u - R(x_u - x_L)$

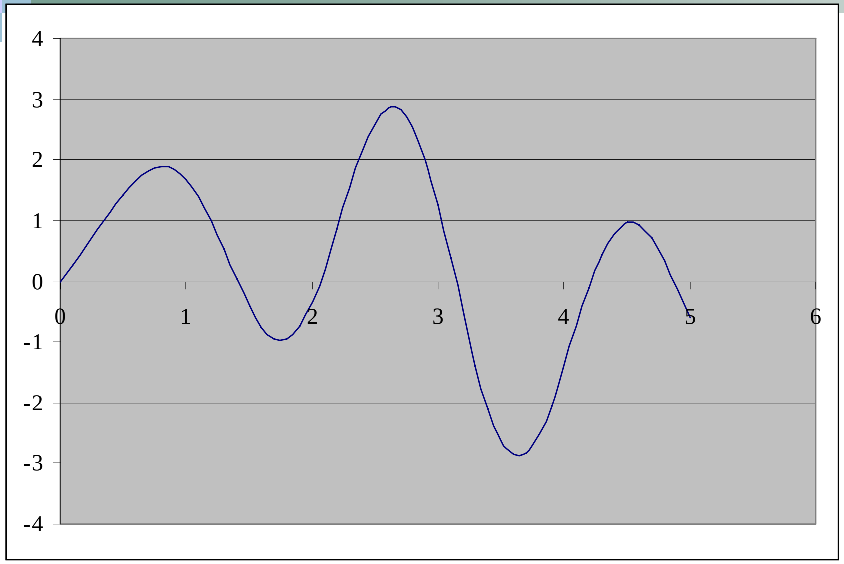
$$f(x) = -1.5x^6 - 2x^4 + 12x$$





x	**0.00000**	**0.76393**	**1.23607**	**2.00000**
f(x)	0.00000	8.18789	4.81418	-104.00000
x	**0.00000**	**0.47214**	**0.76393**	**1.23607**
f(x)	0.00000	5.54964	8.18789	4.81418
x	**0.47214**	**0.76393**	**0.94427**	**1.23607**
f(x)	5.54964	8.18789	8.67784	4.81418
x	**0.76393**	**0.94427**	**1.05573**	**1.23607**
f(x)	8.18789	8.67784	8.10740	4.81418
x	**0.76393**	**0.87539**	**0.94427**	**1.05573**
f(x)	8.18789	8.65523	8.67784	8.10740
x	**0.87539**	**0.94427**	**0.98684**	**1.05573**
f(x)	8.65523	8.67784	8.55989	8.10740
x	0.87539	0.91796	0.94427	0.98684
f(x)	8.65523	8.69790	8.67784	8.55989
x	0.87539	0.90170	0.91796	0.94427
f(x)	8.65523	8.69202	8.69790	8.67784
x	0.90170	0.91796	0.92801	0.94427
f(x)	8.69202	8.69790	8.69469	8.67784
x	0.90170	0.91175	0.91796	0.92801
f(x)	8.69202	8.69724	8.69790	8.69469

```
Function GoldenSection(xlow,xhigh,Error, MaxIterations)
d = R * (xu - xlw)
x1 = xlw + d: x2 = xu - d
   f1 = Fun(x1)
   f2 = Fun(x2)
If f1 > f2 Then
   xopt = x1  : fx = f1
Else
   xopt = x2:   fx = f2
End If
Do While Iter < MaxIterations
   d = R * d
   If f1 > f2 Then
      xlw = x2
      x2 = x1
      x1 = xlw + d
      f2 = f1
      f1 = Fun(x1)
   Else
      xu = x1
      x1 = x2
      x2 = xu - d
      f1 = f2
      f2 = Fun(x2)
   End If
Iter = Iter + 1
   If f1 > f2 Then
      xopt = x1
      fx = f1
   Else
      xopt = x2
      fx = f2
   End If
   If xopt <> 0 Then
      Ea = (1 - R) * Abs((xu -
xlw) / xopt) * 100
   End If
   If Ea < Error Then GoTo 10
Loop
10 GoldenSection = xopt
End
```



$$f(x) = \sin(x) + \frac{5}{3}\sin(3x) - \frac{4}{5}\sin(4x)$$

$$0 \le x \le 5$$

1. Start with $x_0=0$, k=0 and assume Δx, determine $x_{i+1}=x_i+\Delta x$.
2. Determine the value of the function at x_i, x_{i+1},x_{i+2} f_i,f_{i+1},f_{i+2}
3. If $f_i<f_{i+1}<f_{i+2}$ or $f_i>f_{i+1}>f_{i+2}$ then the function is increasing or decreasing within the interval $x_i<x<x_{i+1}$. Let

   $X_i = x_{i+1}$, $x_{i+1}= x_{i+2}$, $x_{i+2}= x_{i+1}+\Delta x$

   $f_i = f_{i+1}$, $f_{i+1}= f_{i+2}$, $f_{i+2}=f(x_{i+1}+\Delta x)$

GO TO Step 2

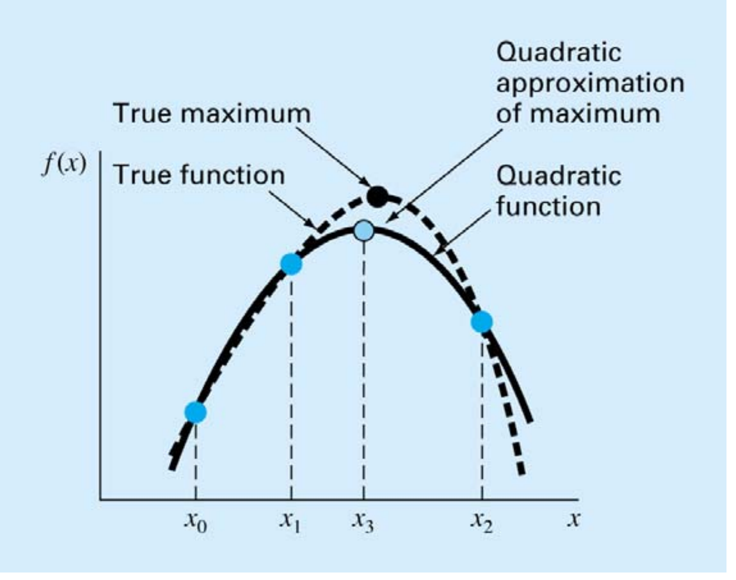
4. else, the function has a maximum within the interval $x_i<x<x_{i+1}$ when $f_{i+1}>f_i$ and $f_{i+1}>f_{i+2}$ (minimum within the interval $x_i<x<x_{i+1}$ when $f_{i+1}<f_i$ and $f_{i+1}<f_{i+2}$)

5. Let $x_{low}=x_i$, $x_{high}= x_{i+2}$ and call Golden Section method and determine a local maximum (or minimum)(x_{opt}(k))

6. Let $x_0=x_{high}$ and k=k+1. GO TO Step 2.

7. Repeat until $x_{i+2}>5$

8. Determine the maximum (or minimum) of the x_{opt}(k)



$$f(x_0) = ax_0^2 + bx_0 + c$$

$$f(x_1) = ax_1^2 + bx_1 + c$$

$$f(x_2) = ax_2^2 + bx_2 + c$$

$$f'(x_3) = 2ax_3 + b = 0$$

$$x_3 = \frac{-b}{2a}$$

$$x_3 = \frac{-b}{2a} = -\frac{\begin{bmatrix} x_0^2 & f(x_0) & 1 \\ x_1^2 & f(x_1) & 1 \\ x_2^2 & f(x_2) & 1 \end{bmatrix}}{2\begin{bmatrix} f(x_0) & x_0 & 1 \\ f(x_1) & x_1 & 1 \\ f(x_2) & x_2 & 1 \end{bmatrix}}$$

$$x_3 = \frac{f(x_0)(x_1^2 - x_2^2) + f(x_1)(x_2^2 - x_0^2) + f(x_2)(x_0^2 - x_1^2)}{2f(x_0)(x_1 - x_2) + 2f(x_1)(x_2 - x_0) + 2f(x_2)(x_0 - x_1)}$$

i.e: Newton Raphson Method for solving *f*(x)=0:

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}$$

For the optimum of f(x) (necessary but not sufficient condition is *f*'(x)=0). Hence determine the root of *f*'(x) = 0 by Newton Raphson method.

$$x_{i+1} = x_i - \frac{f'(x_i)}{f''(x_i)}$$

Direct Methods (random Search methods, Heuristic methods, Genetic Algorithms)

Gradient (steepest descent) methods