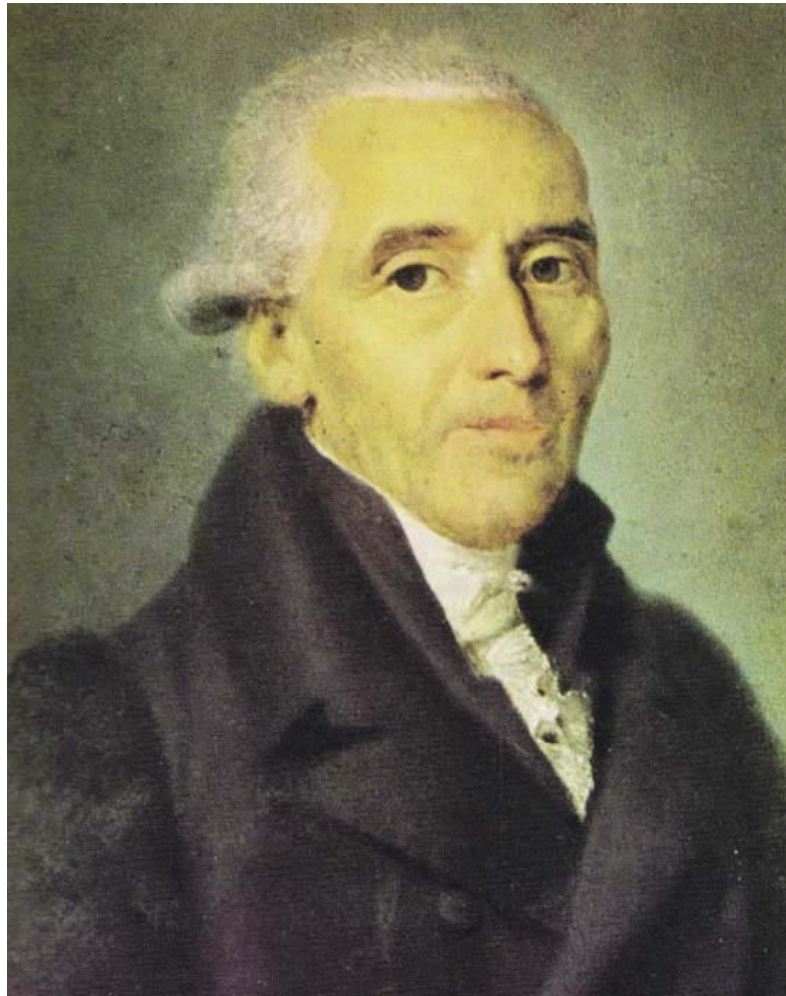


MULTIPLICADORES DE LAGRANGE



ADAY HARO HERNÁNDEZ
FERNANDO ACOSTA MARTÍNEZ
PATRICIA MARCET SANTANA

ÍNDICE

INTRODUCCIÓN TEÓRICA.....	2
DEFINICIÓN DEL PROBLEMA.....	29
SUBRRUTINA: MÉTODO DE OPTIMIZACION POR MULTIPLICADORES DE LAGRANGE.....	36
BIBLIOGRAFÍA.....	51

INTRODUCCIÓN TEÓRICA

Formas Cuadráticas

1) Forma cuadrática en el R^n es toda expresión de la forma $F = \sum_{i,j=1}^n a_{i,j} x_i x_j$ donde los coeficientes a_{ij} y las variables x_i, x_j toman valores reales y se verifica que $a_{ij} = a_{ji}$. Los coeficientes determinan una Matriz simétrica de orden n

$$M = \begin{pmatrix} a_{1,1} & a_{1,2} & a_{1,n} \\ a_{2,1} & a_{2,2} & a_{2,n} \\ a_{n,1} & a_{n,2} & a_{n,n} \end{pmatrix} \quad \text{Recíprocamente cada Matriz cuadrada, simétrica de orden } n \text{ está asociada a una forma cuadrática de orden } n$$

2) Una forma cuadrática F se llama definida positiva, si $F > 0$ para todo sistemas de valores no simultáneamente nulos de las x_i, x_j variables

3) Una forma cuadrática F se llama definida negativa, si $F < 0$ para todo sistemas de valores no simultáneamente nulos de las x_i, x_j variables

4) Una forma cuadrática F se llama semidefinida positiva, si $F \geq 0$ es decir se conserva positiva, anulándose solo para algún sistema de valores de las variables x_i, x_j

5) Una forma cuadrática F se llama semidefinida negativa, si $F \leq 0$ es decir se conserva negativa, anulándose solo para algún sistema de valores de las variables x_i, x_j

6) Una forma cuadrática F se llama indefinida, si F toma valores positivos, nulos o negativos, para distintos sistemas de valores de las variables x_i, x_j .

Llamaremos H al determinante de la Matriz M de los coeficientes de la forma cuadrática	$H = \begin{vmatrix} a_{1,1} & a_{1,2} & a_{1,n} \\ a_{2,1} & a_{2,2} & a_{2,n} \\ a_{n,1} & a_{n,2} & a_{n,n} \end{vmatrix}$	donde $a_{ij} = a_{ji}$
Al menor complementario de orden $k < n$ extraído de H lo llamaremos H_k	$H_k = \begin{vmatrix} a_{1,1} & a_{1,2} & a_{1,k} \\ a_{2,1} & a_{2,2} & a_{2,k} \\ a_{k,1} & a_{k,2} & a_{k,k} \end{vmatrix}$	

TEOREMA 1

Sea $F = \sum_{i,j=1}^n a_{i,j} x_i x_j$ donde los coeficientes a_{ij} y las variables x_i, x_j toman valores reales y se verifica que $a_{ij} = a_{ji}$

La condición necesaria y suficiente para que F sea definida positiva es que $H_k > 0$ para $k = 1..n$

La condición necesaria y suficiente para que F sea definida negativa es que $(-1)^k H_k > 0$ para $k = 1..n$. En esta última expresión, para $k = 1$ (impar), H_k debe ser negativo, de lo contrario no se puede asegurar nada.

DEMOSTRACIÓN

Nos limitaremos a efectuar la demostración en el espacio R^2

$$F = \sum_{i,j=1}^2 a_{i,j} x_i x_j = a_{1,1} x_1 x_1 + a_{1,2} x_1 x_2 + a_{2,1} x_2 x_1 + a_{2,2} x_2 x_2 = a_{1,1} x_1^2 + 2 a_{1,2} x_1 x_2 + a_{2,2} x_2^2$$

ya que $a_{1,2} = a_{2,1}$ multiplicamos y dividimos la expresión por $a_{1,1}$ y en los dos primeros términos completamos el binomio cuadrado haciendo el siguiente artificio

$$F = \frac{a_{1,1}^2}{a_{1,1}} x_1^2 + 2 \cdot \frac{a_{1,1}}{a_{1,1}} a_{1,2} x_1 x_2 + \frac{a_{1,2}^2}{a_{1,1}} x_2^2 - \frac{a_{1,2}^2}{a_{1,1}} x_2^2 + a_{2,2} x_2^2$$

$$F = \frac{1}{a_{1,1}} (a_{1,1} x_1 + a_{1,2} x_2)^2 + \frac{1}{a_{1,1}} (a_{1,1} a_{2,2} - a_{1,2}^2) x_2^2$$

En el espacio R^2 , $H_2 = \begin{vmatrix} a_{1,1} & a_{1,2} \\ a_{2,1} & a_{2,2} \end{vmatrix} = (a_{1,1} a_{2,2} - a_{1,2}^2)$ y $H_1 = a_{1,1}$ y reemplazamos en F

$$F = \frac{1}{H_1} (a_{1,1} x_1 + a_{1,2} x_2)^2 + \frac{1}{H_1} (H_2) x_2^2 = F(x_1, x_2)$$

Entonces si

$$\begin{cases} H_1 > 0 \\ H_2 > 0 \end{cases} \quad \underline{F \text{ es definida positiva}} \qquad \begin{cases} H_1 < 0 \\ (-1)^2 H_2 > 0 \end{cases} \quad \underline{F \text{ es definida negativa}}$$

EXTREMOS DE FUNCIONES DE VARIAS VARIABLES

En éste tema se estudiará la teoría de máximos y mínimos relativos, (o locales), de funciones de varias variables independientes o bien relacionadas entre sí mediante ciertas condiciones adicionales, que al igual que para funciones de una variable independiente constituye una importante aplicación del cálculo diferencial, y en particular de la fórmula de Taylor. Veamos, entonces, los extremos libres en primer término.

EXTREMOS LIBRES DE FUNCIONES DE VARIAS VARIABLES

Aquí denotaremos por $y = f(x_1, x_2, x_3, \dots, x_n) = f(\vec{x})$ una función de “n” variables independientes definida en un cierto subconjunto S de \mathbb{R}^n , $\vec{a} = (a_1, a_2, \dots, a_n)$ un punto de S y $N = N(\vec{a}, r)$ un entorno del punto \vec{a} , de radio $r > 0$.

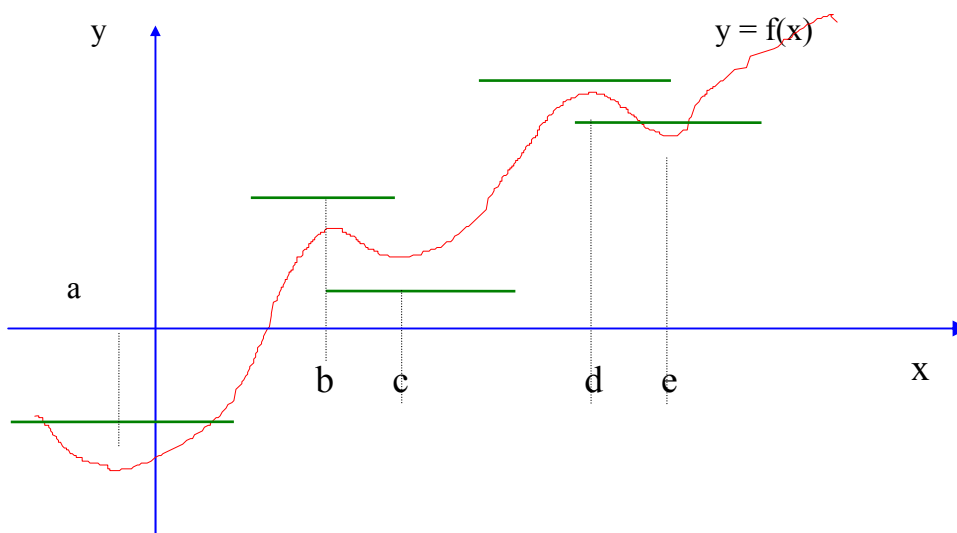
Definición 1: Se dice que $y = f(\vec{x})$ tiene un máximo relativo o local en $\vec{a} \in S$, si:
 $\exists r > 0 / \forall \vec{x} \in S \cap N(\vec{a}, r) : f(\vec{x}) \leq f(\vec{a})$

Definición 2: Se dice que $y = f(\vec{x})$ tiene un mínimo relativo o local en $\vec{a} \in S$, si :
 $\exists r > 0 / \forall \vec{x} \in S \cap N(\vec{a}, r) : f(\vec{a}) \leq f(\vec{x})$

Definición 3: Se dice que $y = f(\vec{x})$ tiene un extremo absoluto o global en $\vec{a} \in S$, si:
(I) $\forall \vec{x} \in S : f(\vec{x}) \leq f(\vec{a})$, o bien :
(II) $\forall \vec{x} \in S : f(\vec{a}) \leq f(\vec{x})$

En el caso (I), $f(\vec{a})$ constituye el máximo absoluto de f , y en el caso (II) $f(\vec{a})$ constituye el mínimo absoluto de f .

Nota: Los máximos y mínimos locales se denominan extremos relativos o locales. La palabra “relativo” (“local”) indica que se compara el valor de la función en el punto $\vec{x} = \vec{a}$ con los valores que ella toma en una vecindad de dicho punto solamente. Así, una función con máximos y mínimos locales puede tomar valores mayores que sus máximos locales y menores que sus mínimos locales. Además, notemos también que una función $y = f(\vec{x})$ puede tener varios máximos y mínimos locales, iguales o no entre sí. Estas últimas observaciones, para el caso de funciones de una variable, se justifican al considerar la siguiente figura.



Condiciones necesarias para la existencia de extremos locales de funciones derivables

Teorema: Sea $y = f(x_1, x_2, \dots, x_n) = f(\vec{x})$ una función definida en un recinto S (conjunto abierto y conexo de \mathbb{R}^n) y derivable en un punto $\vec{a} = (a_1, a_2, \dots, a_n) \in S$. Condición necesaria pero no suficiente para que f tome un valor extremo local en $\vec{x} = \vec{a}$ es que:

$$\begin{cases} \frac{\partial f(\vec{a})}{\partial x_1} = 0 \\ \frac{\partial f(\vec{a})}{\partial x_2} = 0 \\ \vdots \\ \frac{\partial f(\vec{a})}{\partial x_n} = 0, \end{cases} \quad (1)$$

es decir que en ese punto $\vec{x} = \vec{a}$ se anulan todas las derivadas parciales de primer orden de f .

Demostración: Supongamos que $y = f(\vec{x})$ toma un valor extremo local en el punto $\vec{a} \in S$. Ahora consideremos las variables x_2, x_3, \dots, x_n fijadas en los valores a_2, a_3, \dots, a_n respectivamente. En estas condiciones, resulta $y = f(x_1, a_2, a_3, \dots, a_n) = F(x_1)$ una función de la única variable " x_1 ". Por otra parte, esta función $F(x_1)$ debe tener un extremo local en $x_1 = a_1$, como es obvio, y por ello, al ser $F(x_1) = f(x_1, a_2, a_3, \dots, a_n)$ derivable en $x_1 = a_1$, debe ser

$$\frac{dF\vec{x}}{dx_1} = 0 \quad \text{en } x_1 = a_1 \quad \text{pero} \quad \frac{dF\vec{x}}{dx_1} = 0 \quad \text{en } x_1 = a_1 \quad \text{es} \quad \frac{df(x_1, a_1, \dots, a_n)}{dx_1} = \frac{\partial f(a_1, \dots, a_n)}{\partial x_1}$$

Luego, resulta $\frac{\partial f(\vec{a})}{\partial x_1} = 0$. En forma análoga se prueban las demás relaciones (1).

Ahora veamos que si se verifican las relaciones (1), pueden presentarse distintas situaciones en una vecindad del punto $\vec{x} = \vec{a}$.

Es decir, veamos que (1) no es una condición suficiente para garantizar la existencia de extremo de $f(x)$ en $\vec{x} = \vec{a}$ (como lo prueba el ejemplo 3).

Ejemplo 1:

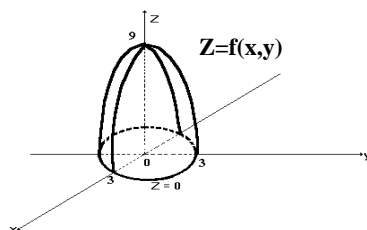
Sea $f(x,y) = 9 - x^2 - y^2$, $\forall (x,y) \in \mathbb{R}^2$. Las derivadas parciales de primer orden de f son

$$\begin{cases} f_x(x,y) = -2x, \\ f_y(x,y) = -2y. \end{cases}$$

Así vemos que $(0,0)$ es el único punto en el cual se anulan simultáneamente ambas derivadas parciales de primer orden de f (punto crítico de f). Por otra parte, es

$$\forall (x,y) \in \mathbb{R}^2: f(x,y) = 9 - x^2 - y^2 = 9 - (x^2 + y^2) \leq 9 = f(0,0).$$

Luego, f tiene en $(0,0)$ un máximo local. También aquí se ve que $f(0,0) = 9$ es el máximo absoluto de f en \mathbb{R}^2 .



Ejemplo 2:

Sea $f(x,y) = 9 + x^2 + y^2 \quad \forall (x,y) \in \mathbb{R}^2$.

Calculamos las derivadas parciales de primer orden de f .

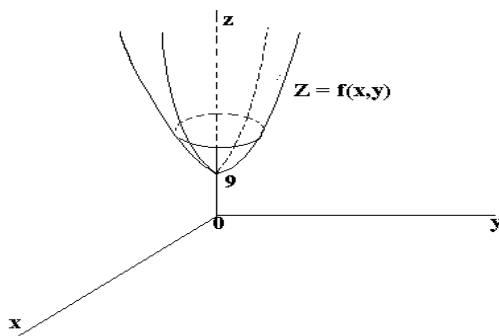
$$\begin{cases} f_x(x,y) = 2x \\ f_y(x,y) = 2y \end{cases}$$

Vemos que en $(0,0)$ se anulan simultáneamente estas derivadas parciales de f .

Además $\forall (x,y) \in \mathbb{R}^2 : f(x,y) = 9 + x^2 + y^2 \geq 9 = f(0,0)$.

Entonces, f tiene en $(0,0)$ un mínimo local.

Obviamente también $f(0,0) = 9$ resulta ser el mínimo absoluto de f en \mathbb{R}^2 .



Ejemplo 3:

Sea $f(x,y) = x^2 - y^2, \quad \forall (x,y) \in \mathbb{R}^2$

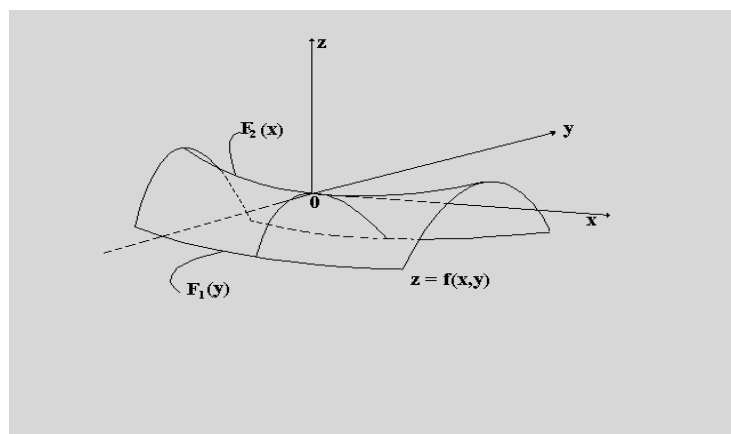
Luego, es $\begin{cases} f_x(x,y) = 2x \\ f_y(x,y) = -2y \end{cases}$

Como siempre, calculadas las derivadas parciales de primer orden de f , vemos que en $(0,0)$ se anulan estas simultáneamente. Luego $(0,0)$ es un punto crítico de f .

Notamos que $\forall y \neq 0 : f(0,y) = 0^2 - y^2 = -y^2 < 0 = f(0,0)$. Y también que $\forall x \neq 0 : f(x,0) = x^2 - 0^2 = x^2 > 0 = f(0,0)$.

De lo anterior concluimos que en todo entorno de $(0,0)$ existen puntos $(x,0)$ para los cuales $f(x,0) > f(0,0)$ y puntos $(0,y)$ para los cuales $f(0,y) < f(0,0)$.

Entonces resulta que en $(0,0)$ f no tiene extremo local, aunque en él ambas derivadas parciales de f se anulan simultáneamente.



Observaciones:

1) Si $y = f(\bar{x})$ es diferenciable, las condiciones (1) implican que la diferencial total de f en el punto \bar{a} en el cual f tiene un extremo relativo, es nula:

$$df(\bar{a}) = \frac{\partial f(\bar{a})}{\partial x_1} dx_1 + \dots + \frac{\partial f(\bar{a})}{\partial x_n} dx_n = 0 \cdot dx_1 + \dots + 0 \cdot dx_n = 0$$

Luego, si f es diferenciable, las condiciones (1) quedan resumidas en: $df(\bar{a}) = 0$ (2)
 En particular, si f es una función de dos variables independientes $z = f(x, y)$, la condición (2) implica que el plano tangente a la gráfica de f en el punto $(a_1, a_2, f(a_1, a_2))$ -cuya ecuación es:
 $z - f(a_1, a_2) = f_x(a_1, a_2) \cdot (x - a_1) + f_y(a_1, a_2) \cdot (y - a_2) = df(a_1, a_2)$ -es horizontal, puesto que resulta: $z - f(a_1, a_2) = df(a_1, a_2) = 0$, o sea $z = f(a_1, a_2)$ constante.

En la práctica, para determinar los extremos de una función derivable, se comienza por resolver el sistema de ecuaciones (1), obteniéndose un número finito o infinito de raíces (a_1, a_2, \dots, a_n) , (b_1, b_2, \dots, b_n) que se denominan puntos críticos, estacionarios o extremantes de la función f porque en ellos puede haber extremo relativo o absoluto de f .
 El examen directo del comportamiento de f en entornos de cada uno de ellos acaba de decidir la cuestión.

CONDICIONES SUFICIENTES PARA LA EXISTENCIA DE EXTREMOS LOCALES

En general, el problema de detectar extremos de funciones de varias variables puede llegar a ser muy complejo. Aquí sólo se estudiará, como aplicación de la fórmula de Taylor, un teorema que provee condiciones suficientes para la existencia de extremo local en un punto crítico dado, en el caso de una función de dos variables independientes que cumpla ciertas condiciones.

Para estudios más detallados y generales del tema en cuestión, remitimos al estudiante a la abundante bibliografía existente sobre el particular; por ejemplo: Análisis Matemático, Rey Pastor - Pi Calleja - Trejo, volumen 2, Editorial Kapelusz, 1957.

Antes de pasar al teorema mencionado, recordaremos un teorema de Bolzano - Weierstrass, y daremos una definición que nos será de utilidad en lo que sigue.

Teorema:(Bolzano - Weierstrass)

Si una función $y = f(x)$ está definida en un intervalo cerrado $[a,b]$ y es continua en él, entonces $\exists p \in [a,b], \exists q \in [a,b] / f(p) = \max. f(x) \wedge f(q) = \min. f(x) \text{ con } x \in [a,b]$

Definición:

Si $f(x,y)$ es una función definida en un recinto D , con derivadas de segundo orden continuas, llamaremos Hessiano de f en D al siguiente determinante funcional.

$$H(x, y) = \begin{vmatrix} f_{xx}(x, y) & f_{xy}(x, y) \\ f_{yx}(x, y) & f_{yy}(x, y) \end{vmatrix} = f_{xx}(x, y) \cdot f_{yy}(x, y) - f_{xy}^2(x, y) \quad \forall (x, y) \in D.$$

Teorema: Si una función $z = f(x,y)$ definida en un recinto $R \subset \mathbb{R}^2$ tiene derivadas parciales segundas continuas en un entorno del punto $P(a,b)$ y no simultáneamente nulas en el punto $P(a,b)$, siendo $f_x(a,b) = f_y(a,b) = 0$, entonces :

- 1) $H(a,b) = f_{xx}(a,b) \cdot f_{yy}(a,b) - [f_{xy}^2(a,b)] > 0$ y $f_{xx}(a,b) < 0 \Rightarrow f$ tiene un máximo local en $P(a,b)$.
- 2) $H(a,b) > 0$ y $f_{xx}(a,b) > 0 \Rightarrow f$ tiene un mínimo local en $P(a,b)$.
- 3) $H(a,b) < 0 \Rightarrow f$ tiene en $P(a,b)$ un punto de ensilladura, donde no hay extremo local.
- 4) $H(a,b) = 0 \Rightarrow$ no se podrá afirmar o negar la existencia de extremo local en $P(a,b)$.

Demostración:

Supongamos entonces que la función $z = f(x,y)$ definida en un recinto $R \subset \mathbb{R}^2$ tenga derivadas parciales segundas continuas en un entorno de un punto crítico $P(a,b)$ de f del recinto R y que estas derivadas no se anulen simultáneamente en $P(a,b)$.

Desarrollando la función $f(x,y)$ por la fórmula de Taylor para $n=1$ en un entorno del punto crítico $P(a,b)$, tendremos:

$$f(a+h, b+k) = f(a,b) + \left(\frac{\partial}{\partial x} \cdot h + \frac{\partial}{\partial y} \cdot k\right) f(a,b) + \frac{1}{2!} \left(\left(\frac{\partial}{\partial x} \cdot h + \frac{\partial}{\partial y} \cdot k\right)^2 f(a+\theta h, b+\theta k)\right) \text{ con } 0 < \theta < 1$$

Ahora, teniendo en cuenta que es $f_x(a,b) = f_y(a,b) = 0$ y la continuidad de las derivadas segundas en $P(a,b)$, resulta:

$$\Delta f(a,b) = f(a+h, b+k) - f(a,b) = \frac{1}{2} [h^2(f_{xx} + \varepsilon_1) + 2hk(f_{xy} + \varepsilon_2) + k^2(f_{yy} + \varepsilon_3)],$$

con $\varepsilon_i = \varepsilon_i(h,k) \rightarrow 0$ para $\rho = \sqrt{h^2 + k^2} \rightarrow 0$ ($i = 1,2,3$), es decir

$$\Delta f(a, b) = \frac{1}{2} (f_{xx} h^2 + 2 f_{xy} hk + f_{yy} k^2) + \frac{1}{2} o(\rho^2) \quad (a)$$

donde f_{xx} , f_{xy} , f_{yy} indican derivadas tomadas precisamente en el punto $P(a,b)$, y $o(\rho^2) = \varepsilon_1 h^2 + 2\varepsilon_2 hk + \varepsilon_3 k^2$ es un infinitésimo para $\rho \rightarrow 0$ de orden superior a ρ^2 .

Recordemos que si en un entorno de (a,b) es

$\Delta F > 0$ en (a,b) existe un **MINIMO RELATIVO Estricto**

$\Delta F < 0$ en (a,b) existe un **MAXIMO RELATIVO Estricto**

$\Delta F \geq 0$ en (a,b) existe un **MINIMO RELATIVO AMPLIO**

$\Delta F \leq 0$ en (a,b) existe un **MAXIMO RELATIVO AMPLIO**

Si h y k son suficientemente pequeños el signo de ΔF es el signo del primer paréntesis en (a)

Pero esa expresión es una forma cuadrática en h y k cuyo determinante vale $H = \begin{bmatrix} f_{xx} & f_{xy} \\ f_{yx} & f_{yy} \end{bmatrix}$

$= f_{xx} \cdot f_{yy} - f_{xy}^2 = \frac{\partial(f_{(x)}, f_{(y)})}{\partial(x, y)}$ este determinante recibe el nombre de HESSIANO de la función $f(x,y)$.

- Si la forma cuadrática es definida positiva $\Delta F > 0$ en (a,b) existe un **MINIMO RELATIVO Estricto**
- Si la forma cuadrática es definida negativa $\Delta F < 0$ en (a,b) existe un **MAXIMO RELATIVO Estricto**
- Si la forma cuadrática es indefinida ΔF no tiene siempre el mismo signo tomando valores positivos y negativos.
- No obstante que el plano tangente es horizontal a la superficie en el punto (a,b) no existe extremo y este es un **PUNTO DE ENSILLADURA**.
- Si la forma cuadrática es semidefinida, ΔF tomara signo constante, anulándose para algún conjunto de valores de h y k (no simultáneamente nulos).
- Este se trata de un caso **DUDOSO** porque no podemos afirmar o negar que en (a,b) exista extremo relativo.

El Plano tangente a la superficie no lo es solo en un punto, sino a lo largo de una recta llamada singular. Se trata entonces de un CASI-MINIMO o un CASI-MAXIMO según que la forma cuadrática sea semidefinida positiva o negativa respectivamente.

Entonces dada $z=f(x,y)$ para determinar los puntos extremos hacemos lo siguiente:

- 1) Hallamos las derivadas primeras de $f(x,y)$ respecto de x e y respectivamente.
- 2) Planteamos el sistemas de ecuaciones $\begin{cases} f_x(x,y)=0 \\ f_y(x,y)=0 \end{cases}$ y resolvemos el sistema y hallamos los puntos críticos $P(a,b)$, $Q(a',b')$ etc.

3) Calculamos el HESSIANO $H = \begin{bmatrix} f_{xx} & f_{xy} \\ f_{yx} & f_{yy} \end{bmatrix} = f_{xx} \cdot f_{yy} - f_{xy}^2 = \frac{\partial(f_{(x)}, f_{(y)})}{\partial(x, y)}$

4) Y reemplazamos en H las coordenadas de los puntos críticos P, Q, etc.

- Si $H(a,b) > 0$ existe máximo o mínimo ? Recordamos el teorema de las formas cuadráticas que nos aseguraba que F es definida positiva si H_1 y H_2 son positivo $\Delta F > 0$ y por lo tanto existe un **MINIMO RELATIVO**.
- Si $H_1 = f_{xx}(a,b) < 0$ en (a,b) existe un **MÁXIMO RELATIVO**.
- Si $H(a,b) = 0$ la forma cuadrática es semidefinida y tendremos en (a,b) un caso **DUDOSO**.
- Si $H(a,b) < 0$ la forma cuadrática es indefinida y tendremos en (a,b) un **PUNTO DE ENSILLADURA**.

EXTREMOS DE FUNCIONES CON VARIABLES LIGADAS O EXTREMOS CONDICIONADOS

INTRODUCCION

Los problemas de máximos y mínimos locales se presentan con frecuencia en forma tal que las variables no son todas independientes. Sin embargo, en muchos casos las condiciones de vínculo no permiten despejar unas variables en función de otras.

Es por ello que se buscan métodos que puedan utilizarse también cuando las condiciones de vínculo solamente definen funciones en forma implícita.

Sea en general $z = f(x,y)$ una función diferenciable de dos variables, estando ligadas estas por una ecuación diferenciable $g(x,y) = 0$.

Si se cumple la condición $\frac{\partial g}{\partial y} \neq 0$, se podrá considerar $y = y(x)$ definida en forma implícita por $g(x,y) = 0$, resultando $z = f[x, y(x)] = H(x)$.

Buscamos un método que permita determinar los extremos de $H(x)$ mediante f y g , sin necesidad de conocer la expresión explícita de $H(x)$ ni la de $y = y(x)$.

Se puede interpretar geométricamente el problema en el plano si se supone que en él se busca extremar una función $f(x,y)$ sobre una determinada curva $g(x,y) = 0$.

Supuestas satisfechas las condiciones de existencia, continuidad y derivabilidad de $y = y(x)$, la condición necesaria (pero no suficiente) de existencia de extremo local será la anulación de la derivada total de $f(x,y)$ siendo $y = y(x)$ esto es de $H(x) = f[x, y(x)]$, obteniendo $y'(x)$ mediante $g(x,y) = 0$. (o sea $H'(x) = 0$)

o bien al:

Llegamos así al sistema :

$$\begin{cases} f_x + f_y y' = 0 \\ g_x + g_y y' = 0 \end{cases} \quad (2)$$

$$\begin{cases} f_x dx + f_y dy = 0 \\ g_x dx + g_y dy = 0 \end{cases} \quad (2')$$

$$(ya \text{ que es } y' = -\frac{g_x}{g_y}),$$

Notemos que en la primera de las relaciones (2') dx es incremento independiente, mientras que dy es diferencial funcional, determinada por la segunda relación, pudiéndose intercambiar los papeles si así conviene para las condiciones de existencia o de contorno.

Esta última observación es importante puesto que las condiciones (2) pueden no dar el extremo buscado si éste se alcanza en un punto de contorno.

Ahora, de la segunda de las relaciones (2'), tenemos

$$dy = -\frac{g_x}{g_y} dx \quad (\text{supuesto } dx \text{ incremento independiente})$$

y reemplazando esta expresión de dy en la primera de las relaciones (2') resulta

$$0 = f_x dx + f_y dy = f_x dx - f_y \frac{g_x}{g_y} dx = \frac{f_x g_y - f_y g_x}{g_y} \cdot dx$$

Luego, como dx es incremento independiente, necesariamente debe ser

$$\frac{f_x g_y - f_y g_x}{g_y} = 0, \quad \text{esto es} \quad \boxed{f_x g_y - f_y g_x = 0} \quad (3)$$

Tenemos entonces que la expresión (3) junto con la ecuación de enlace $g(x,y)=0$ forman el sistema de dos ecuaciones con dos incógnitas:

$$\begin{cases} f_x g_y - f_y g_x = 0 \\ g(x, y) = 0 \end{cases} \quad (4)$$

Las soluciones del sistema (4) son los puntos críticos o extremantes en los cuales puede existir extremo local en dichas condiciones de continuidad y diferenciabilidad.

Debemos notar que sin estas últimas restricciones mencionadas acaso existen otros extremos singulares, sin olvidar que deben considerarse los puntos del contorno.

Finalmente, observemos que el sistema (4) da condiciones necesarias pero no suficientes para la existencia de extremo local. La discusión de suficiencia se hace mediante el estudio del signo de d^2f , estando aquí ligadas dx, dy, d^2y por las condiciones $dg = 0$, $d^2g = 0$, si consideramos dx incremento independiente y a dy diferencial funcional.

En la práctica, al resolver problemas concretos, se logra a veces determinar la naturaleza del punto crítico en base al carácter del problema mismo, mediante consideraciones auxiliares (por ejemplo, consideraciones físicas o geométricas).

Optimización (matemática)

La humanidad hace tiempo que busca, o profesa buscar, mejores maneras de realizar las tareas cotidianas de la vida. A lo largo de la historia de la humanidad, se puede observar la larga búsqueda de fuentes más efectivas de alimentos al comienzo y luego de materiales, energía y manejo del entorno físico. Sin embargo, relativamente tarde en la historia de la humanidad, comenzaron a formularse ciertas clases de preguntas generales de manera cuantitativa, primero en palabras y después en notaciones simbólicas. Un aspecto predominante de estas preguntas generales era la búsqueda de lo "mejor" o lo "óptimo".

Generalmente, los gerentes buscan simplemente lograr alguna mejora en el nivel de rendimiento, es decir, un problema de "búsqueda de objetivo". Cabe destacar que estas palabras normalmente no tienen un significado preciso

Se han realizado grandes esfuerzos por describir complejas situaciones humanas y sociales. Para tener significado, esto debería escribirse en una expresión matemática que contenga una o más variables, cuyos valores deben determinarse. La pregunta que se formula, en términos generales, es qué valores deberían tener estas variables para que la expresión matemática tenga el mayor valor numérico posible (maximización) o el menor valor numérico posible (minimización). A este proceso general de maximización o minimización se lo denomina optimización.

La optimización, también denominada programación matemática, sirve para encontrar la respuesta que proporciona el mejor resultado, la que logra mayores ganancias, mayor producción o felicidad o la que logra el menor costo, desperdicio o malestar. Con frecuencia, estos problemas implican utilizar de la manera más eficiente los recursos, tales como dinero, tiempo, maquinaria, personal, existencias, etc. Los problemas de optimización generalmente se clasifican en lineales y no lineales, según las relaciones del problema sean lineales con respecto a las variables. Existe una serie de paquetes de software para resolver problemas de optimización.

La Programación Matemática, en general, aborda el problema de determinar asignaciones óptimas de recursos limitados para cumplir un objetivo dado. El objetivo debe representar la meta del decisor. Los recursos pueden corresponder, por ejemplo, a personas, materiales, dinero o terrenos. Entre todas las asignaciones de recursos admisibles, queremos encontrar la/s que maximiza/n o minimiza/n alguna cantidad numérica tal como ganancias o costos.

El objetivo de la optimización global es encontrar la mejor solución de modelos de decisiones difíciles, frente a las múltiples soluciones locales.

La optimización intenta dar respuesta a un tipo general de problemas de la forma:

$$\begin{array}{l} \max(\min)f(x) \\ x \in \Omega \subset R^n \end{array}$$

Donde $x = (x_1, x_2, \dots, x_n)$ es un vector y representa variables de decisión, $f(x)$ es llamada función objetivo y representa o mide la calidad de las decisiones (usualmente números enteros o reales) y Ω es el conjunto de decisiones factibles o restricciones del problema.

Algunas veces es posible expresar el conjunto de restricciones Ω como solución de un sistema de igualdades o desigualdades.

$$\begin{array}{l} g(x_1, \dots, x_n) \leq 0 \\ h(x_1, \dots, x_n) \equiv 0 \end{array}$$

Un problema de optimización trata entonces de tomar una decisión óptima para maximizar (ganancias, velocidad, eficiencia, etc.) o minimizar (costos, tiempo, riesgo, error, etc.) un criterio determinado. Las restricciones significan que no cualquier decisión es posible.

Desde la década de los 60 la programación lineal (PL) ha sido aplicada en diversas áreas de la vida como por ejemplo: sistemas militares, agrícolas, económicos, de transporte y de salud. La PL ofrece bases importantes en el desarrollo de métodos de solución de otras técnicas de la Investigación de operaciones, como lo son la programación entera, la estocástica y la no lineal [Taha 1991]. La PL juega un papel muy importante en el estudio de los problemas continuos de optimización considerados como la frontera de los problemas de optimización combinatoria, ya que en los continuos se tienen las características necesarias para que sean considerados dentro del tipo combinatorio [Papadimitriou and Steiglitz, 1982]: Un problema de optimización combinatoria siempre se le involucra un conjunto de instancias, donde cada una de ellas cuenta con un conjunto finito de posibles soluciones (característica imprescindible de los problemas continuos).

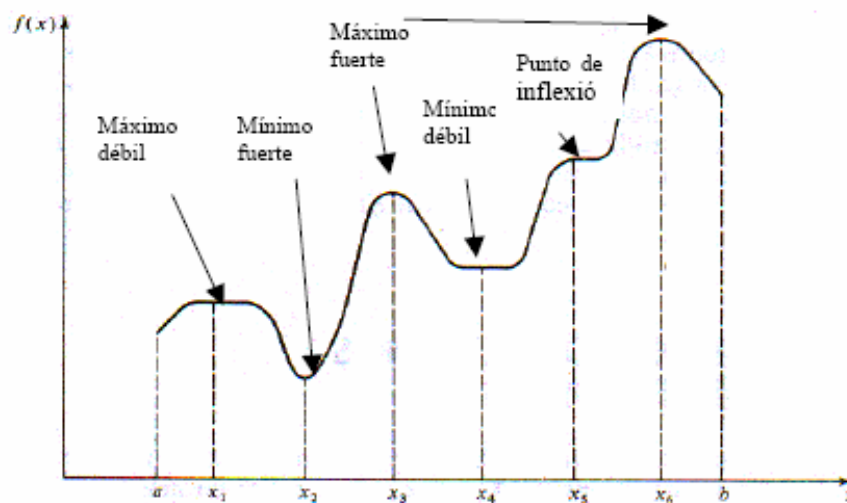
Por otra parte la teoría de optimización clásica se usa para la obtención de los máximos y mínimos de funciones no lineales restringidas y no restringidas, en los que se hace uso del calculo diferencial.

MAXIMOS Y MINIMOS

Mínimo (fuerte): Un punto extremo x_0 de una función $f(x_0)$ define un mínimo de la función si $f(x_0 + h) > f(x_0)$, donde x_0 es cualquier punto de la función y h en valor absoluto es suficientemente pequeña.

Máximo (fuerte): Un punto extremo x_0 de una función $f(x_0)$ define un máximo de la función si $f(x_0 + h) < f(x_0)$, donde x_0 es cualquier punto de la función y h en valor absoluto es suficientemente pequeña.

Una función puede contener varios máximos y mínimos, identificados por los puntos extremos de la función. En la figura 1 se puede observar esto, los puntos x_1 , x_3 y x_6 son máximos, de la figura notamos que $f(x_6)$ es el mayor que $f(x_1)$ y $f(x_3)$, a este punto se le conoce como máximo global de la función y a los restantes como máximos locales. Lo mismo se puede ver para los mínimos, en los que también existe un mínimo global $f(x_2)$ y un mínimo local $f(x_4)$. Como es de lógico, solo puede existir un solo global y posiblemente varios locales.



Una condición necesaria pero no suficiente para que x_0 sea un punto extremo, es que para una función con mas de una variable, el gradiente $\tilde{\nabla} f(x_0) = 0$. Si es cierto esto entonces x_0 será conocido como punto estacionario.

Una condición suficiente para que un punto estacionario sea extremo es que la matriz Hessiana H obtenida en x_0 del sistema de ecuaciones sea positiva cuando x_0 es un punto extremo de mínimo. Y negativa cuando x_0 es un punto extremo de máximo.

Un máximo débil implica un numero finito de máximos alternativos (ver figura 1) y se define como x_0 es un máximo débil, si $f(x_0 + h) \leq f(x_0)$. Un análisis similar es para los mínimos débiles.

Un punto de inflexión se encuentra cuando la evaluación del gradiente da cero y no es un extremo, esto es, se debe de cumplir la condición de la matriz Hessiana.

TÉCNICAS DE OPTIMIZACION CLÁSICA.

Método de Newton

El encontrar la solución a un sistema de ecuaciones no lineal es mucho más difícil que el de un sistema lineal. El método de Newton es un procedimiento iterativo y permite la linealización de un sistema de ecuaciones no lineal, para posteriormente darle solución por cualquier método numérico de ecuaciones lineales simultáneas, esta forma parte de los métodos conocidos como métodos de gradiente.

Un sistema de n ecuaciones con n incógnitas $(x_1, x_2, x_3, \dots, x_n)$, se conoce como no lineal, si una o más de estas no es lineal.

De manera general, la solución de un sistema de “n” ecuaciones no lineales aplicando el método de Newton (Conte y Boor, 1987) se plantea como sigue:

$$\begin{bmatrix} \frac{\partial f_1(x, y, z, \dots)}{\partial x} & \frac{\partial f_1(x, y, z, \dots)}{\partial y} & \frac{\partial f_1(x, y, z, \dots)}{\partial z} & \dots \\ \frac{\partial f_2(x, y, z, \dots)}{\partial x} & \frac{\partial f_2(x, y, z, \dots)}{\partial y} & \frac{\partial f_2(x, y, z, \dots)}{\partial z} & \dots \\ \frac{\partial f_3(x, y, z, \dots)}{\partial x} & \frac{\partial f_3(x, y, z, \dots)}{\partial y} & \frac{\partial f_3(x, y, z, \dots)}{\partial z} & \dots \\ \vdots & \vdots & \vdots & \dots \\ \frac{\partial f_n(x, y, z, \dots)}{\partial x} & \frac{\partial f_n(x, y, z, \dots)}{\partial y} & \frac{\partial f_n(x, y, z, \dots)}{\partial z} & \dots \end{bmatrix} \begin{bmatrix} \Delta x_1 \\ \Delta y_1 \\ \Delta z_1 \\ \vdots \end{bmatrix} = \begin{bmatrix} -f_1 \\ -f_2 \\ -f_3 \\ \vdots \\ -f_n \end{bmatrix} \quad 1. (1)$$

evaluado en (x_i, y_i, z_i, \dots) y resolviendo se tiene:

$$x_{i+1} = x_i + \Delta x_i, y_{i+1} = y_i + \Delta y_i, z_{i+1} = z_i + \Delta z_i, \dots$$

Sea la expresión (2) la suma de todas las ecuaciones no lineales que representan el sistema en estudio

$$f_{2n}(X^{k+1}) = D_n(X^{k+1}) + E_n(X^{k+1}) \quad (2)$$

La iteración de Newton se podrá obtener para el conjunto de ecuaciones a partir de (2), usando la serie de Taylor en una aproximación de primer orden

$$f_{2n}^{k+1}(X_{i+1}) \approx f_{2n}^{k+1}(X_i) + f'_{2n}{}^{k+1}(X_i)(X_{i+1} - X_i) = 0 \quad (3)$$

Las ecuaciones (3) representan un conjunto de $2n$ ecuaciones lineales acopladas para el vector de incógnitas X en el paso de tiempo $k+1$, en la iteración $i+1$. Los términos en derivadas darán lugar a la formación de la matriz Jacobiana:

$$f'_{2n}{}^{k+1}(X_i) = \sum \frac{\partial f_{2n}{}^{k+1}(X_i)}{\partial X_i} \quad (4)$$

La solución del conjunto de ecuaciones lineales encontradas en cada iteración (matriz Jacobiana) será resuelta por cualquier método lineal de matrices hasta que los residuales sean menores a una tolerancia designada, muy próxima a cero, ecuación (3).

Método de derivadas restringidas (Jacobiano).

Se puede considerar como una generalización del método simplex para programación lineal.

Considere el problema

$$\begin{aligned} &\text{Minimizar } z = f(X) \\ &\text{Sujeto a } g(X) = 0 \end{aligned}$$

$$\begin{aligned} &\text{Donde } X = (x_1, x_2, \dots, x_n) \\ &g = (g_1, g_2, \dots, g_m)^T \end{aligned}$$

Las funciones $f(X)$ y $g_i(X)$, donde $i = 1, 2, \dots, m$, se suponen diferenciables y doblemente continuas. El utilizar derivadas restringidas es encontrar una expresión de forma cerrada para las primeras derivadas parciales de $f(X)$ en todos los puntos que satisfacen las restricciones $g(X) = 0$. Estos puntos estacionarios entonces se sabe que son donde dichas derivadas parciales se anulan.

Por el teorema de Taylor, para los puntos $X + \Delta X$ en el entorno factible de X , se deduce que

$$f(X + \Delta X) - f(X) = \nabla f(X) \Delta X + 0(\Delta X^2_j)$$

y

$$g(X + \Delta X) - g(X) = \nabla g(X) \Delta X + 0(\Delta X^2_j)$$

Cuando $\Delta x_j \rightarrow 0$ se tiene que

$$\partial f(X) = \nabla f(X) \Delta X \quad \text{y} \quad \partial g(X) = \nabla g(X) \Delta X \quad (5)$$

ya que $g(X) = 0$, entonces $\partial g(X) = 0$, se deduce que,

$\begin{aligned} \partial f(X) - \nabla f(X) \partial X &= 0 \\ \nabla g(X) \partial X &= 0 \end{aligned}$
--

con esto se tienen m ecuaciones con n incógnitas, cuando $m < n$ tenemos, si definimos a X ahora como:

$$X = (Y, Z)$$

Donde

$$Y = (y_1, y_2, \dots, y_n) \text{ y } Z = (z_1, z_2, \dots, z_{n-m})$$

Indican a las variables dependientes e independientes, respectivamente y que corresponden al vector X. Volviendo a escribir los vectores gradiente de f y g en términos de Y y Z se encuentra que

$$\begin{aligned}\nabla f(Y, Z) &= (\nabla_y f, \nabla_z f) \\ \nabla g(Y, Z) &= (\nabla_y g, \nabla_z g)\end{aligned}$$

Donde

$$J = \nabla_y g = \begin{pmatrix} \nabla_y g_1 \\ \vdots \\ \nabla_y g_m \end{pmatrix}$$

$$C = \nabla_z g = \begin{pmatrix} \nabla_z g_1 \\ \vdots \\ \nabla_z g_m \end{pmatrix}$$

J_{mn} se conoce como la matriz jacobiana y $C_{m \times n-m}$ es la matriz de control. Utilizando las definiciones anteriores en (4), se puede escribir el conjunto original de ecuaciones como

$$\partial f(Y, Z) = \nabla_y f \partial Y + \nabla_z f \partial Z \quad (6)$$

y

$$J \partial Y = -C \partial Z$$

Como J se supone que no es singular, existirá su inversa J^{-1} . Por lo tanto,

$$\partial Y = -J^{-1} C \partial Z$$

sustituyendo ∂Y en la ecuación (6) se obtiene a ∂f como una función de ∂Z

$$\partial f(Y, Z) = (\Delta_z f + \Delta_y f J^{-1} C) \partial Z$$

aplicando a esta ecuación la derivada restringida con respecto a al vector independiente Z, se tiene

$$\nabla_c f = \partial_c f(Y, Z) / \partial_c Z = \Delta_z f + \Delta_y f J^{-1} C$$

donde $\nabla_c f$ representa el vector gradiente restringido de f con respecto a Z . Por consiguiente f debe de ser nulo en los puntos estacionarios.

Método Lagrangiano.

Esta muy relacionado con el método jacobiano y se puede desarrollar a partir de aquel. Anteriormente se demostró que:

$$\partial f(Y, Z) = \nabla_y f \partial Y + \nabla_z f \partial Z \quad (7)$$

$$\partial g = J \partial Y + C \partial Z$$

Supongamos que ∂g no es igual a 0 entonces despejamos ∂Y

$$\partial Y = J^{-1} \partial g - J^{-1} C \partial Z$$

sustituyendo en la ecuación (6) tenemos

$$\partial f(Y, Z) = \nabla_y f J^{-1} \partial g + \nabla_z f \partial Z \quad (8)$$

Donde

$$\nabla_c f = \nabla_z f - \nabla_y f J^{-1} C$$

Considerando que en el punto extremo (y para cualquier punto estacionario) $X_0 = (Y_0, Z_0)$, el gradiente restringido $\nabla_c f$ debe anularse, entonces de (8)

$$\partial f(Y_0, Z_0) = \nabla_{y_0} f J^{-1} \partial g(Y_0, Z_0)$$

o bien,

$$\frac{\partial f}{\partial g} = \nabla_{y_0} f J^{-1} \quad (9)$$

Esta relación (9) permite estudiar el efecto de variaciones pequeñas de g sobre el valor óptimo de f evaluando la tasa de cambio de f con respecto a g . Estas tasas usualmente se conocen como coeficientes de sensibilidad.

De (9) si tomamos

$$\lambda = \nabla_y \partial f J^{-1} = \frac{\partial f}{\partial g}$$

Tendremos

$$\partial f - \lambda \partial g = 0 \quad (10)$$

Esta ecuación satisface las condiciones necesarias para los puntos estacionarios ya que la expresión para $\partial f / \partial g$ se calcula con $\nabla_c f = 0$. Otra representación mas conveniente de la ecuación (10) es tomando sus derivadas parciales con respecto a todas las x_j . Lo anterior da

$$\frac{\partial}{\partial X_j} (f - \lambda g) = 0 \quad j = 1, 2, \dots, n$$

Las ecuaciones resultantes junto con las de restricciones $g = 0$ proporcionan los valores factibles de X y λ que satisfacen las condiciones necesarias para puntos estacionarios. Este procedimiento define al método de Lagrange para identificar los puntos estacionarios de problemas de optimización con restricciones de igualdad.

EL METODO SIMPLEX

Para resolver un modelo PL por medio del método simplex, este se deberá de poner en forma estándar aplicando los siguientes puntos:

1. Todas las restricciones son ecuaciones con un segundo miembro positivo.
2. Una restricción se convierte en ecuación adicionando una variable de holgura (o restando una variable de exceso) al primer miembro de la ecuación. El segundo miembro de la ecuación se puede hacer positivo, multiplicando ambos lados por -1 . Y la dirección de una desigualdad se invierte multiplicando ambos miembros de la ecuación por -1 .
3. Todas las variables son positivas.
4. En el caso de que una variable y_i cualquiera, sea irrestricta (no restringida), esta pudiera expresarse en términos de dos variables positivas mediante el uso de la sustitución.
 $y_i = y'_i - y''_i$ donde $y'_i, y''_i \geq 0$
5. La función objetivo puede ser de maximización o de minimización.

LA PROGRAMACIÓN NO LINEAL: METODOS DE GRADIENTE.

El método del gradiente se utiliza para la optimización de funciones que son dos veces diferenciables. La idea general para generar puntos sucesivos, iniciando de un punto inicial dado, en la dirección del incremento más rápido (maximización) de la función. La técnica es conocida como método de gradiente debido a que gradiente de la función en un punto es indicativo de la rapidez del incremento.

Método Newton-Raphson

Uno de los métodos de gradiente es el Newton-Raphson, este método está basado en la resolución de ecuaciones simultáneas que representan la condición necesaria para optimalidad, llamada $\nabla f(\mathbf{X}) = 0$.

La desventaja de utilizar la condición necesaria $\nabla f(\mathbf{X}) = 0$ para determinar los puntos estacionarios es la dificultad para resolver las ecuaciones simultáneas resultantes numéricamente. El método Newton-Raphson es un procedimiento iterativo para resolver ecuaciones simultáneas no lineales.

Considere las ecuaciones simultáneas:

$$f_i(X) = 0, \quad i=1,2,\dots, m$$

donde X^K es un punto dado. Por expansión de Taylor

$$f_i(X) \cong f_i(X^K) + \nabla f_i(X^K)(X - X^K), \quad i=1,2,\dots,m$$

Así, la ecuación original puede aproximarse a

$$f_i(X^K) + \nabla f_i(X^K)(X - X^K) = 0, \quad i=1,2,\dots,m$$

Estas ecuaciones pueden escribirse en notación matricial como

$$A_K + B_K(X - X^K) = 0$$

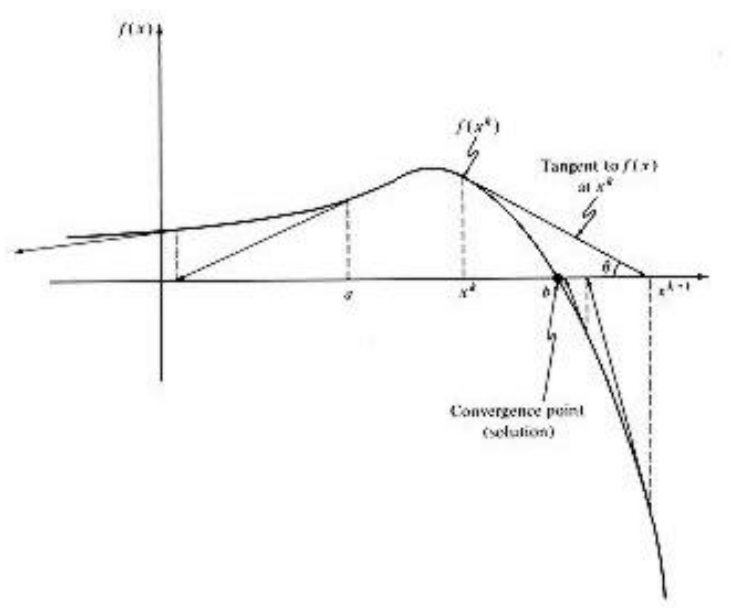
Tomando en cuenta la condición de que toda $f_i(\mathbf{X})$ es independiente, B_K es necesariamente no singular. Así, la ecuación anterior resulta

$$X = X^K - B_K^{-1} A_K$$

La idea del método es iniciar de un punto inicial X_0 . Utilizando la ecuación anterior, un nuevo punto X^{K+1} puede determinarse de X^K . El procedimiento termina con X^m como solución cuando $X^m \cong X^{m-1}$.

Una interpretación geométrica de método se muestra con una función de una sola variable en la figura 2. La relación entre x^K y x^{K+1} para una función de una sola variable $f(x)$ se reduce a

$$X^{K+1} = X^K - \frac{f(x^K)}{f'(x^K)} \quad \text{o} \quad f'(x^K) = \frac{f(x^K)}{x^K - x^{K+1}}$$



Representación grafica de Newton Raphson.

Analizando la figura 2 se puede ver que x^{K+1} se determinan de la pendiente de $f(x)$ en x^K , donde $\tan\theta = f'(x^K)$.

Un problema con este método es que la convergencia no esta garantizada siempre a menos que la función se comporte bien. En la figura, si el punto inicial x_0 es a el método divergirá. No existe una manera fácil para localizar un “buen” punto inicial x_0 . Tal vez una solución para este problema es usar prueba y error.

Obtención del algoritmo

De forma similar a como sucede con una sola ecuación, se toma $g(x) = x - A(x) f(x)$ siendo $A(x)$ una matriz $n \times n$ no singular $\forall x$.

El algoritmo resulta,

$$x_{m+1} = x_m - A(x_m) f(x_m)$$

Si se toman los elementos a_{ij} de $A(x)$ constantes se tiene que,

$$g_i = x_i - \sum_{k=1}^n a_{ik} f_k(x)$$

Algoritmo de Newton-Raphson

Por tanto, el algoritmo de Newton-Raphson para sistemas de ecuaciones no lineales resulta,

$$x_{m+1} = x_m - J^{-1}(x_m)f(x_m)$$

Obtención de la matriz A

Para elegir a_{ik} , derivamos respecto a x_j

$$\frac{\partial g_i(x)}{\partial x_j} = \delta_{ij} - \sum_{k=1}^n a_{ik} \frac{\partial f_k(x)}{\partial x_j}$$

$\delta_{ii} = 1$ y $\delta_{ij} = 0$ si $i \neq j$. De forma matricial,

$$G(x) = I - A J(x)$$

siendo $G(x)$ y $J(x)$ las matrices jacobianas de g y f , respectivamente.

Para que exista convergencia

$$G(s) = I - A J(s) = 0 \quad \gg \quad A = J^{-1}(s)$$

Como sucedía en el caso de una sola ecuación, no conocemos la solución s y definimos A en función de x ,

$$A(x) = J^{-1}(x)$$

El cálculo de $J^{-1}(x_m)$ no se realiza explícitamente, sino que se plantea un sistema de ecuaciones lineales en cada iteración del método de Newton-Raphson,

$$J(x_m)(x_{m+1} - x_m) = -f(x_m)$$

tomando como incógnita,

$$y_m = x_{m+1} - x_m$$

Una vez resuelto el sistema lineal $J(x_m) y_m = -f(x_m)$, la solución es actualizada.

Algoritmo de Newton Modificado

Si no actualizamos la matriz Jacobiana $J(X_0)$ y la mantenemos constante a lo largo de todas las iteraciones del proceso de Newton-Raphson, se obtiene el método de Newton-Modificado,

$$\begin{aligned} J(x_0)y_m &= -f(x_m) \\ x_{m+1} &= x_m + y_m \end{aligned}$$

Método de ascenso pronunciado (steepest ascent)

La terminación del método del gradiente se realiza en el punto donde el gradiente de vuelve nulo. Esta es solamente una condición necesaria para la optimalidad. Así se enfatiza que la optimalidad no puede verificarse al menos que se conozca a priori que $f(\mathbf{X})$ es cóncava o convexa.

Suponga que $f(\mathbf{X})$ es maximizada. Tomemos a X_0 como el punto inicial en el cual el procedimiento inicia y define a $\nabla f(X^K)$ como el gradiente de f en el k ésimo punto de X^K . La idea del método es determinar el PATH p a través del cual df/dp es maximizada en un punto dado. Este resultado es alcanzado si los puntos sucesivos X^K y X^{K+1} se seleccionan tales que

$$X^{K+1} = X^K + r^K \nabla f(X^K)$$

donde r^K es un parámetro llamado **tamaño de paso optimo**.

El parámetro r^K se determina tal que X^{K+1} de cómo resultado una gran mejora de f . En otras palabras, si la función $h(r)$ se define como

$$h(r) = f(X^K + r \nabla f(X^K))$$

r^K es el valor de r maximizando

El procedimiento propuesto termina cuando dos puntos de prueba sucesivos X^K y X^{K+1} son aproximadamente iguales. Esto es equivalente a tener $r^K \nabla f(X^K) \cong 0$

Bajo el supuesto de que $r^K \neq 0$, el cual siempre será verdadero al menos que X_0 se convierta en optimo de $f(\mathbf{X})$, esto es equivalente a la condición necesaria

$$\nabla f(X^K) = 0$$

Tipos de optimizaciones

Según el nivel de generalidad que tome el problema, será la resolución que se plantee.

Optimización clásica

Si la restricción no existe, o es una restricción de igualdad, con menor o igual número de VARIABLES que la función objetivo entonces, el CÁLCULO DIFERENCIAL, da la respuesta, ya que solo se trata de buscar los valores extremos de una FUNCIÓN.

Optimización con restricciones de desigualdad - optimización no clásica

Si la restricción contiene mayor cantidad de variables que la función objetivo, o la restricción contiene restricciones de desigualdad, existen métodos en los que en algunos casos se pueden encontrar los valores máximos o mínimos.

Si tanto restricciones como función objetivo son lineales (PROGRAMACIÓN LINEAL o PL), la existencia de máximo (mínimo), esta asegurada, y el problema se reduce a la aplicación de unos simples ALGORITMOS de ÁLGEBRA LINEAL elemental los llamados MÉTODO SIMPLEX; y MÉTODO DUAL. Sin embargo, si estas condiciones no se cumplen, existen, las llamadas condiciones de Khun -Tucker, las cuales en algunos casos, pueden ser utilizables, para probar encontrar puntos críticos, máximos o mínimos. Sin embargo, esta es un área aún muy poco desarrollada de la matemática, frecuentemente, las condiciones de Khun y Tucker fallan, o no son suficientes, para la existencia de extremos.

Optimización estocástica: Cuando las variables del problema (función objetivo y/o restricciones) son VARIABLES ALEATORIAS el tipo de optimización realizada es optimización estocástica.

Programación Lineal (PL)

La programación lineal muchas veces es uno de los temas preferidos tanto de profesores como de alumnos. La capacidad de introducir la PL utilizando un abordaje gráfico, la facilidad relativa del método de solución, la gran disponibilidad de paquetes de software de PL y la amplia gama de aplicaciones hacen que la PL sea accesible incluso para estudiantes con poco conocimiento de matemática. Además, la PL brinda una excelente oportunidad para presentar la idea del análisis what-if o análisis de hipótesis ya que se han desarrollado herramientas poderosas para el análisis de post optimalidad para el modelo de PL.

La Programación Lineal (PL) es un procedimiento matemático para determinar la asignación óptima de recursos escasos. La PL es un procedimiento que encuentra su aplicación práctica en casi todas las facetas de los negocios, desde la publicidad hasta la planificación de la producción. Problemas de transporte, distribución, y planificación global de la producción son los objetos más comunes del análisis de PL. La industria petrolera parece ser el usuario más frecuente de la PL. Un gerente de procesamiento de datos de una importante empresa petrolera recientemente calculó que del 5% al 10% del tiempo de procesamiento informático de la empresa es destinado al procesamiento de modelos de PL y similares.

La programación lineal aborda una clase de problemas de programación donde tanto la función objetivo a optimizar como todas las relaciones entre las variables correspondientes a los recursos son lineales. Este problema fue formulado y resuelto por primera vez a fines de la década del 40. Rara vez una nueva técnica matemática encuentra una gama tan diversa de aplicaciones prácticas de negocios, comerciales e industriales y a la vez recibe un desarrollo teórico tan exhaustivo en un período tan corto. Hoy en día, esta teoría se aplica con éxito a problemas de presupuestos de capital, diseño de dietas, conservación de recursos, juegos de estrategias, predicción de crecimiento económico y sistemas de transporte. Recientemente la teoría de la programación lineal también contribuyó a la resolución y unificación de diversas aplicaciones.

Es importante que el lector entienda desde el comienzo que el término "programación" tiene un significado distinto cuando se refiere a Programación Lineal que cuando hablamos de Programación Informática. En el primer caso, significa planificar y organizar mientras que en el segundo caso, significa escribir las instrucciones para realizar cálculos. La capacitación en una clase de programación tiene muy poca relevancia directa con la otra clase de programación. De hecho, el término "programación lineal" se acuñó antes de que la palabra programación se relacionara con el software de computación. A veces se evita esta confusión utilizando el término optimización lineal como sinónimo de programación lineal.

Cualquier problema de PL consta de una función objetivo y un conjunto de restricciones. En la mayoría de los casos, las restricciones provienen del entorno en el cual usted trabaja para lograr su objetivo. Cuando usted quiere lograr el objetivo deseado, se dará cuenta de que el entorno fija ciertas restricciones (es decir, dificultades, limitaciones) para cumplir con su deseo (vale decir, el objetivo).

Qué es una función: una función es una cosa que hace algo. Por ejemplo, una máquina de moler café es una función que transforma los granos de café en polvo. La función (objetivo) traza, traduce el dominio de entrada (denominado región factible) en un rango de salida con dos valores finales denominados valores máximo y mínimo.

Cuando se formula un problema de toma de decisiones como un programa lineal, se deben verificar las siguientes condiciones:

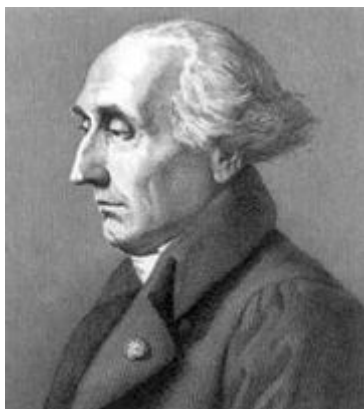
1. La función objetivo debe ser lineal. Vale decir que se debe verificar que todas las variables estén elevadas a la primera potencia y que sean sumadas o restadas (no divididas ni multiplicadas);
2. El objetivo debe ser ya sea la maximización o minimización de una función lineal. El objetivo debe representar la meta del decisor; y
3. Las restricciones también deben ser lineales. Asimismo, la restricción debe adoptar alguna de las siguientes formas (las restricciones de PL siempre están *cerradas*).

Optimización con información no perfecta

En este caso la cantidad de variables, o más aún la función objetivo puede ser desconocida o también variable. En este campo, la matemática conocida como matemática borrosa, está realizando esfuerzos, por resolver el problema. Sin embargo, como el desarrollo

de esta área de la matemática es aún demasiado incipiente, son escasos los resultados obtenidos.

Joseph-Louis de Lagrange



Joseph Louis Lagrange (25 DE ENERO de 1736 - 10 DE ABRIL de 1813) fue un MATEMÁTICO, FÍSICO y ASTRÓNOMO ITALIANO que después vivió en PRUSIA y FRANCIA. Lagrange trabajó para FEDERICO II DE PRUSIA, en BERLÍN, durante veinte AÑOS. Lagrange demostró el TEOREMA DEL VALOR MEDIO, desarrolló la MECÁNICA LAGRANGIANA y tuvo una importante contribución en ASTRONOMÍA.

MÉTODO DE LOS MULTIPLICADORES DE LAGRANGE

Muchos problemas de optimización tienen restricciones o ligaduras en los valores que pueden usarse para lograr la solución óptima. Tales ligaduras tienden a complicar los problemas de optimización ya que la solución óptima puede ocurrir fácilmente en un punto frontera del dominio.

Para resolver tales problemas se utiliza una técnica ingeniosa que se conoce como método de los multiplicadores de lagrange en honor al Matemático Francés Joseph Louis LAGRANGE (1736-1813) quien introdujo el método en su famoso artículo de mecánica cuando tenía diecinueve años.

Este método implica la introducción de nuevos parámetros llamados multiplicadores de Lagrange, mediante el cual se logra mayor sencillez y simetría en los cálculos y se elimina la necesidad de distinguir previamente entre variables dependientes e independientes, así como la intervención de diferenciales segundas de variables dependientes en la discusión de suficiencia.

Supongamos entonces que buscamos extremar una función $z = f(x,y)$ con la condición $g(x,y) = 0$, donde tanto f como g son diferenciables y $g_y \neq 0$ (o bien $g_x \neq 0$, si $g_y = 0$) Si en el sistema (2') visto anteriormente, a la primera ecuación le sumamos la segunda multiplicada por un parámetro λ , ordenando respecto de dx y dy se obtiene:

$$\begin{cases} f_x dx + f_y dy = 0 \\ g_x dx + g_y dy = 0 \end{cases} \quad (f_x dx + f_y dy) + \lambda(g_x dx + g_y dy) = 0 \quad \boxed{(f_x + \lambda g_x) dx + (f_y + \lambda g_y) dy = 0} \quad (5)$$

Como en (5) las diferenciales dx , dy no son independientes, no podemos deducir más que los paréntesis deben anularse. Ahora, si se elige λ de modo que se anule el coeficiente de dy , resulta:

$$(6) \quad \begin{cases} f_x + \lambda g_x = 0 \\ f_y + \lambda g_y = 0 \end{cases}$$

que conjuntamente con la condición de vínculo $g(x,y) = 0$ dan tres ecuaciones para determinar los puntos críticos y el correspondientemente parámetro λ para cada uno de ellos:

$$\begin{cases} f_x + \lambda g_x = 0 \\ f_y + \lambda g_y = 0 \\ g(x,y) = 0 \end{cases}$$

Lagrange observó que las ecuaciones (6) corresponden o se establecen cuando se buscan los extremos locales libres de una función que la llamaremos función auxiliar de Lagrange

$$\boxed{F(x,y) = f(x,y) + \lambda g(x,y)} \quad (7)$$

considerando en ella x e y como variables independientes y λ como constante.

Por otra parte, para discutir el signo de d^2f basta considerar el de d^2F , ya que sobre $g(x,y) = 0$ es $F(x,y) = f(x,y)$, como se ve en (7), y con esa condición $g(x,y) = 0$, será también $d^2F = d^2f$.

Recordando como se calcula la diferencial segunda de funciones compuestas, tendremos para d^2F la expresión

$$d^2F = \left(\frac{\partial}{\partial x} dx + \frac{\partial}{\partial y} dy \right)^2 F + \frac{\partial F}{\partial y} d^2y$$

Ahora, la segunda ecuación del sistema (6) es, según la expresión (7) $F_y = 0$, y como sobre $g(x,y) = 0$ es $d^2F = d^2f$, tendremos $d^2f = \left(\frac{\partial}{\partial x} dx + \frac{\partial}{\partial y} dy \right)^2 F$

Así, vemos que el cálculo formal de $d^2F = d^2f$ se efectúa considerando también en F a x e y como variables independientes.

Cabe aclarar que si $g_y = 0$, no podrá hallarse un λ tal que anule el coeficiente de dy en (5).

Entonces, si $g_x \neq 0$, el método sigue siendo válido con solo permutar los roles de x e y . Para los puntos singulares de la curva $g(x,y) = 0$ (puntos de la curva $g(x,y) = 0$ que además satisfacen $g_x = g_y = 0$), no podrán obtenerse las ecuaciones (6) obviamente, ni aún permutando los roles de x e y , y entonces el método de Lagrange no es aplicable.

MULTIPLICADORES DE LAGRANGE

Si f y g satisfacen las hipótesis del Teorema de LAGRANGE y f tiene un máximo o mínimo sujeto a la ligadura $g(x,y) = 0$, entonces dicho extremo se producirá en uno de los puntos críticos de la función F dada por

$$F(x, y, \lambda) = f(x, y) - \lambda g(x, y)$$

Para hallar los puntos críticos se resuelve el siguiente sistema

$$\begin{cases} F_x(x, y, \lambda) = \frac{\partial f(x, y)}{\partial x} - \lambda \frac{\partial g(x, y)}{\partial x} = 0 \\ F_y(x, y, \lambda) = \frac{\partial f(x, y)}{\partial y} - \lambda \frac{\partial g(x, y)}{\partial y} = 0 \\ F_\lambda(x, y, \lambda) = g(x, y) = 0 \end{cases}$$

El método de Multiplicadores de Lagrange dice que dadas $f: U \subset \mathbb{R}^n \rightarrow \mathbb{R}$ $g: U \subset \mathbb{R}^n \rightarrow \mathbb{R}$ dos funciones con valores reales de clase $C^1(U)$, con $x_0 \in U$, $g(x_0) = c$ y $S = \{x \in \mathbb{R}^n : g(x) = c\}$. Asumiendo que $\nabla g(x_0) \neq 0$. Si f restringida a S tiene un máximo y/o mínimo relativo en $x_0 \in S$, entonces existe un número real λ tal que

$$\nabla f(x_0) = \lambda \nabla g(x_0)$$

El Método de los Multiplicadores de Lagrange nos permite encontrar los puntos $x \in \mathbb{R}^n$ que optimizan (producen máximos y/o mínimos) una función dada $f(x)$, sujeta a la restricción $g(x) = 0$.

Esta idea es esencialmente la extensión natural del método usual para funciones de una variable, buscar máximos y/o mínimos entre sus puntos críticos, es decir, los puntos x en que $f'(x) = 0$. En este caso consideramos $F(x) = f(x) - \lambda g(x)$. Es claro que $\max F(x) = \max f(x)$. Así basta buscar valores de x and λ en los cuales $\nabla F = 0$, es decir,

$$\nabla f(x) = \lambda \nabla g(x)$$

DEFINICIÓN DEL PROBLEMA

La realización de nuestro trabajo se centra en la resolución de un problema de optimización de una función f , esta función puede tener hasta N variables, las cuales estarán sometidas a M restricciones de tipo igualdad. Lógicamente las restricciones deberán ser como máximo $N-1$, ya que de lo contrario nos encontraríamos con un sistema de ecuaciones de única solución, lo cual sería absurdo.

La optimización de la función se hará mediante los multiplicadores de Lagrange. De esta forma construiremos primeramente la función Φ constituida por la función f más las restricciones g multiplicadas por cada uno de los λ (multiplicadores de Lagrange).

$$\Phi = f(x_1 \dots x_n) + \lambda_1 g_1(x_1 \dots x_n) + \dots + \lambda_n g_n(x_1 \dots x_n)$$

Derivando la función de Lagrange respecto de cada una de las variables, obtenemos un sistema de ecuaciones que posteriormente resolveremos mediante el método de Newton Modificado.

$$J \times x = -D\Phi$$

Donde J es la matriz Jacobiana compuesta por las derivadas segundas de la función de Lagrange, que para dicho método permanece constante durante todo el cálculo. Sin embargo, la matriz columna $D\Phi$ es el sistema de ecuaciones que obtenemos a partir de derivar la función de Lagrange, la cual va a ir modificándose en cada iteración hasta que el error sea menor que una cierta tolerancia.

Elaboración del Trabajo

El problema más laborioso de nuestro trabajo es el cálculo de las derivadas, ya que las debe hacer el programa en vez de pedir las como habíamos visto en clase. Dicho cálculo, sobre todo de las derivadas segundas cruzadas, no es tan sencillo como se puede pensar a priori.

Cálculo de las derivadas primeras

Una vez obtenida la función de Lagrange, la primera derivada respecto a una variable se calcula de la siguiente manera; siguiendo el mismo proceso para todas las variables que presenta el Lagrangiano, osea N+M variables, por tanto N+M derivadas.

$$\frac{\partial \Phi}{\partial x_1} = \frac{\Phi(x_1 + h, x_2, \dots, x_n) - \Phi(x_1 - h, x_2, \dots, x_n)}{2h}$$

...

$$\frac{\partial \Phi}{\partial \lambda_n} = \frac{\Phi(x_1, \dots, x_n + h) - \Phi(x_1, \dots, x_n - h)}{2h}$$

Cuando hayamos finalizado el cálculo de todas las derivadas primeras, ya tendremos el segundo miembro de la ecuación del método de Newton Modificado.

Cálculo de las segundas derivadas

Vamos a proceder al cálculo de las segundas derivadas que nos van a hacer falta introducirlas en la matriz Jacobiana.

Las derivadas respecto a cada una de las variables no dan excesivos problemas ya que es aplicar la fórmula que sigue y repetirla para todas las variables. Los términos que vamos a calcular ahora, conformarán la diagonal principal de la matriz.

$$\frac{\partial^2 \Phi}{\partial x_1^2} = \frac{\Phi(x_1 + h, \dots, \lambda_n) - 2\Phi(x_1, \dots, \lambda_n) + \Phi(x_1 - h, \dots, \lambda_n)}{2h}$$

...

$$\frac{\partial^2 \Phi}{\partial \lambda_n^2} = \frac{\Phi(x_1, \dots, \lambda_n + h) - 2\Phi(x_1, \dots, \lambda_n) + \Phi(x_1, \dots, \lambda_n - h)}{2h}$$

Como vemos el cálculo de las derivadas segundas se hace siempre de la misma forma, no es excesivamente complicado.

Ahora vamos a calcular las derivadas cruzadas. Para ello haremos uso de las siguientes ecuaciones en las que las derivadas cruzadas aparecen implícitas:

$$\Phi(x+h, y) = \Phi(x, y) + h \frac{\partial \Phi}{\partial x} + \frac{h^2}{2} \frac{\partial^2 \Phi}{\partial x^2} + \frac{h^3}{3!} \frac{\partial^3 \Phi}{\partial x^3} + \dots$$

$$\Phi(x-h, y) = \Phi(x, y) - h \frac{\partial \Phi}{\partial x} + \frac{h^2}{2} \frac{\partial^2 \Phi}{\partial x^2} - \frac{h^3}{3!} \frac{\partial^3 \Phi}{\partial x^3} + \dots$$

$$\Phi(x, y+h) = \Phi(x, y) + h \frac{\partial \Phi}{\partial y} + \frac{h^2}{2} \frac{\partial^2 \Phi}{\partial y^2} + \frac{h^3}{3!} \frac{\partial^3 \Phi}{\partial y^3} + \dots$$

$$\Phi(x, y-h) = \Phi(x, y) - h \frac{\partial \Phi}{\partial y} + \frac{h^2}{2} \frac{\partial^2 \Phi}{\partial y^2} - \frac{h^3}{3!} \frac{\partial^3 \Phi}{\partial y^3} + \dots$$

$$\Phi(x+h, y+h) = \Phi(x, y) + h \frac{\partial \Phi}{\partial x} + h \frac{\partial \Phi}{\partial y} + \frac{h^2}{2} \left[\frac{\partial^2 \Phi}{\partial x^2} + 2 \frac{\partial^2 \Phi}{\partial x \partial y} + \frac{\partial^2 \Phi}{\partial y^2} \right] + \dots$$

$$\Phi(x-h, y-h) = \Phi(x, y) - h \frac{\partial \Phi}{\partial x} - h \frac{\partial \Phi}{\partial y} + \frac{h^2}{2} \left[\frac{\partial^2 \Phi}{\partial x^2} + 2 \frac{\partial^2 \Phi}{\partial x \partial y} + \frac{\partial^2 \Phi}{\partial y^2} \right] - \dots$$

$$\Phi(x-h, y+h) = \Phi(x, y) - h \frac{\partial \Phi}{\partial x} + h \frac{\partial \Phi}{\partial y} + \frac{h^2}{2} \left[\frac{\partial^2 \Phi}{\partial x^2} - 2 \frac{\partial^2 \Phi}{\partial x \partial y} + \frac{\partial^2 \Phi}{\partial y^2} \right] + \dots$$

$$\Phi(x+h, y-h) = \Phi(x, y) + h \frac{\partial \Phi}{\partial x} - h \frac{\partial \Phi}{\partial y} + \frac{h^2}{2} \left[\frac{\partial^2 \Phi}{\partial x^2} - 2 \frac{\partial^2 \Phi}{\partial x \partial y} + \frac{\partial^2 \Phi}{\partial y^2} \right] + \dots$$

Para ello multiplicaremos todas las funciones por constantes de la A a la H, de forma que los valores de estas constantes hagan que se anule la suma de todos los términos menos los que contienen a las derivadas cruzadas. Así:

$$A\Phi(x+h, y) = A\Phi(x, y) + Ah \frac{\partial \Phi}{\partial x} + A \frac{h^2}{2} \frac{\partial^2 \Phi}{\partial x^2} + A \frac{h^3}{3!} \frac{\partial^3 \Phi}{\partial x^3} + \dots$$

$$B\Phi(x-h, y) = B\Phi(x, y) - Bh \frac{\partial \Phi}{\partial x} + B \frac{h^2}{2} \frac{\partial^2 \Phi}{\partial x^2} - B \frac{h^3}{3!} \frac{\partial^3 \Phi}{\partial x^3} + \dots$$

$$C\Phi(x, y+h) = C\Phi(x, y) + Ch \frac{\partial \Phi}{\partial y} + C \frac{h^2}{2} \frac{\partial^2 \Phi}{\partial y^2} + C \frac{h^3}{3!} \frac{\partial^3 \Phi}{\partial y^3} + \dots$$

$$D\Phi(x, y-h) = D\Phi(x, y) - Dh \frac{\partial \Phi}{\partial y} + D \frac{h^2}{2} \frac{\partial^2 \Phi}{\partial y^2} - D \frac{h^3}{3!} \frac{\partial^3 \Phi}{\partial y^3} + \dots$$

$$E\Phi(x+h, y+h) = E\Phi(x, y) + Eh \frac{\partial \Phi}{\partial x} + Eh \frac{\partial \Phi}{\partial y} + E \frac{h^2}{2} \left[\frac{\partial^2 \Phi}{\partial x^2} + 2 \frac{\partial^2 \Phi}{\partial x \partial y} + \frac{\partial^2 \Phi}{\partial y^2} \right] + \dots$$

$$F\Phi(x-h, y-h) = F\Phi(x, y) - Fh \frac{\partial \Phi}{\partial x} - Fh \frac{\partial \Phi}{\partial y} + F \frac{h^2}{2} \left[\frac{\partial^2 \Phi}{\partial x^2} + 2 \frac{\partial^2 \Phi}{\partial x \partial y} + \frac{\partial^2 \Phi}{\partial y^2} \right] - \dots$$

$$G\Phi(x-h, y+h) = G\Phi(x, y) - Gh \frac{\partial \Phi}{\partial x} + Gh \frac{\partial \Phi}{\partial y} + G \frac{h^2}{2} \left[\frac{\partial^2 \Phi}{\partial x^2} - 2 \frac{\partial^2 \Phi}{\partial x \partial y} + \frac{\partial^2 \Phi}{\partial y^2} \right] + \dots$$

$$H\Phi(x+h, y-h) = H\Phi(x, y) + Hh \frac{\partial \Phi}{\partial x} - Hh \frac{\partial \Phi}{\partial y} + H \frac{h^2}{2} \left[\frac{\partial^2 \Phi}{\partial x^2} - 2 \frac{\partial^2 \Phi}{\partial x \partial y} + \frac{\partial^2 \Phi}{\partial y^2} \right] + \dots$$

Como dijimos, sumando e igualando a cero los coeficientes de los términos que no contienen derivadas cruzadas (también los de las derivadas terceras, que no se han puesto en las ecuaciones de arriba porque no cabían), obtenemos el siguiente sistema de ecuaciones:

$$1) A - B + E - F - G + H = 0$$

$$2) C - D + E - F + G - H = 0$$

$$3) A + B + E + F + G + H = 0$$

$$4) C + D + E + F + G + H = 0$$

$$5) E - F + G - H = 0$$

$$6) E - F - G + H = 0$$

Puesto que tenemos más variables que ecuaciones, como era de esperar, debemos resolver el sistema dando valores a dos de las incógnitas. Para nuestro caso lo más razonable es da valor $A=2$ y $E=2$, de esta forma obtenemos el resto de los valores fácilmente:

$$A = 2 \qquad C = 2 \qquad E = 2 \qquad G = -4$$

$$B = 2 \qquad D = 2 \qquad F = 2 \qquad H = -4$$

Una vez comprobados los valores obtenidos, los sustituimos en la suma de todas las ecuaciones:

$$\begin{aligned} & A\Phi(x+h, y) + B\Phi(x-h, y) + C\Phi(x, y+h) + D\Phi(x, y-h) + E\Phi(x+h, y+h) + \\ & + F\Phi(x-h, y-h) + G\Phi(x-h, y+h) + H\Phi(x+h, y-h) = \\ & (A+B+C+D+E+F+G+H)\Phi(x, y) - Gh^2 \frac{\partial^2 \Phi}{\partial x \partial y} - 2Hh^2 \frac{\partial^2 \Phi}{\partial x \partial y} \end{aligned}$$

$$2\Phi(x+h, y) + 2\Phi(x-h, y) + 2\Phi(x, y+h) + 2\Phi(x, y-h) + 2\Phi(x+h, y+h) + 2\Phi(x-h, y-h) + \\ - 4\Phi(x-h, y+h) - 4\Phi(x+h, y-h) = (4)\Phi(x, y) + 4h^2 \frac{\partial^2 \Phi}{\partial x \partial y} + 8h^2 \frac{\partial^2 \Phi}{\partial x \partial y}$$

Ahora despejamos el término de la derivada cruzada:

$$\frac{\partial^2 \Phi}{\partial x \partial y} = \frac{1}{6h^2} (\Phi(x+h, y) + \Phi(x-h, y) + \Phi(x, y+h) + \Phi(x, y-h) + \Phi(x+h, y+h) + \Phi(x-h, y-h) + \\ - 2\Phi(x-h, y+h) - 2\Phi(x+h, y-h) - 2\Phi(x, y))$$

Este procedimiento es para el caso particular de las variables x e y , pero en nuestro problema tenemos $N+M$ variables, así que utilizaremos esta fórmula para calcular todas las derivadas cruzadas que posteriormente vamos a introducir en la matriz Jacobiana.

Volviendo al método de Newton Modificado, construimos la matriz Jacobiana que quedará fija durante todo el proceso:

$$J = \begin{bmatrix} \frac{\partial^2 \Phi}{\partial x_1^2} & \frac{\partial^2 \Phi}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 \Phi}{\partial x_1 \partial \lambda_n} \\ \frac{\partial^2 \Phi}{\partial x_2 \partial x_1} & \frac{\partial^2 \Phi}{\partial x_2^2} & \cdots & \frac{\partial^2 \Phi}{\partial x_2 \partial \lambda_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 \Phi}{\partial \lambda_n \partial x_1} & \frac{\partial^2 \Phi}{\partial \lambda_n \partial x_2} & \cdots & \frac{\partial^2 \Phi}{\partial \lambda_n^2} \end{bmatrix}$$

Ahora construimos la matriz columna $D\Phi_0$, compuesta por las derivadas primeras iniciales de la función Φ , que se irán modificando en cada iteración por los resultados obtenidos en x :

$$D\Phi = \begin{bmatrix} \frac{\partial \Phi}{\partial x_1} \\ \vdots \\ \frac{\partial \Phi}{\partial \lambda_n} \end{bmatrix}$$

Por último, el método de Newton Modificado se basa en la siguiente igualdad:

$$J \times x = -D\Phi$$

$$\begin{bmatrix} \frac{\partial^2 \Phi}{\partial x_1^2} & \frac{\partial^2 \Phi}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 \Phi}{\partial x_1 \partial \lambda_n} \\ \frac{\partial^2 \Phi}{\partial x_2 \partial x_1} & \frac{\partial^2 \Phi}{\partial x_2^2} & \cdots & \frac{\partial^2 \Phi}{\partial x_2 \partial \lambda_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 \Phi}{\partial \lambda_n \partial x_1} & \frac{\partial^2 \Phi}{\partial \lambda_n \partial x_2} & \cdots & \frac{\partial^2 \Phi}{\partial \lambda_n^2} \end{bmatrix} \times \begin{bmatrix} x_1 \\ \vdots \\ \lambda_n \end{bmatrix} = - \begin{bmatrix} \frac{\partial \Phi}{\partial x_1} \\ \vdots \\ \frac{\partial \Phi}{\partial \lambda_n} \end{bmatrix}$$

La solución del problema será el vector $\begin{bmatrix} x_1 \\ \vdots \\ \lambda_n \end{bmatrix}$, que serán los valores de las variables

para los cuales la función inicial f que se tiene, es óptima y además cumple las M restricciones de tipo igualdad que se dan en el enunciado. Dado que el método de Newton Modificado es un método iterativo, el programa nos pedirá que introduzcamos una tolerancia para la cual, consideremos como bueno el resultado obtenido.

A continuación mostramos el programa mediante el cual hemos elaborado la subrutina que nos permite encontrar la solución al problema que se nos ha asignado

SUBRRUTINA:

MÉTODO DE OPTIMIZACION POR MULTIPLICADORES DE LAGRANGE

```
% METODO DE OPTIMIZACION POR MULTIPLICADORES DE LAGRANGE
%
% Optimizar la funcion no lineal F(X), dada una
% aproximacion inicial X:
%
% ENTRADA: Numero de variables de la ecuacion N; aproximacion inicial
%          X=(X(1),...,X(n)); tolerancia h; numero de
%          restricciones N; numero de iteraciones NN.
%
% SALIDA:  Solucion aproximada X=(X(1),...,X(n)) o un mensaje
%          de que el numero de iteraciones fue sobrepasado.
syms('OK', 'N', 'I', 'J', 'P', 'h', 'NN', 'X', 'ZZ', 'g', 'q',
'F', 'FF', 'RR');
syms('FLAG', 'NAME', 'OUP', 'K', 'A', 'R', 's', 'ss', 'Res', 'fs',
'f', 'Q', 'T');
syms('K1', 'I1', 'Z1', 'IR1', 'IA1', 'J1', 'C1', 'L1', 'JA1',
'EP', 'EE', 'BB');
TRUE = 1;
FALSE = 0;
fprintf(1, 'Este es el metodo de Optimizacion funciones no lineales por
Multiplicadores de Lagrange.\n');
fprintf(1, 'La funcion y las restricciones pueden ser escritas ahora o
leidas de un fichero.\n');
fprintf(1, 'El numero de variables de la funcion no puede exceder de
5.\n');
fprintf(1, 'En otro caso habria que modificar el programa.\n');

OK = FALSE;
while OK == FALSE
    fprintf(1, 'Escriba el numero de variables N.\n');
    N = input(' ');
    if N <= 5
        OK = TRUE;
    else
        fprintf(1, 'El numero de variables N debe ser un entero menor que
6.\n');
    end;
end;
OK = FALSE;
while OK == FALSE
    fprintf(1, 'Escriba el numero de restricciones M.\n');
    M = input(' ');
    if M <= N-1
        OK = TRUE;
    else
        fprintf(1, 'El numero de restricciones M debe ser un entero menor
que el numero de variables N.\n');
    end;
end;
```

```

A = zeros(N+M,N+M+1);
X = zeros(1,N+M);
Y = zeros(1,N+M);
FF = zeros (1,N+M);
fprintf(1,'Entrada de la funcion\n');
fprintf(1,'1. Pantalla\n');
fprintf(1,'2. Fichero de texto\n');
fprintf(1,'Escriba 1 o 2\n');
BB = input(' ');
if BB == 2
    fprintf(1,'Escriba el nombre del fichero en la forma -
disco:\\nombre.ext\n');
    fprintf(1,'Por ejemplo:   A:\\DATA.DTA\n');
    NAME = input(' ','s');
    INP = fopen(NAME,'rt');

    f = fscanf(INP,'%s',1);

    for I = 1 : M
        Res(I) = fscanf(INP,'%s',1);
    end;

    for I = 1 : M+N
        X(I) = fscanf(INP,'%f',1);
    end;

    fclose(INP);

else

    fprintf(1,'Escriba la funcion F en terminos de y1 ... y%d \n',N);
    f = input(' ','s');

    for I = 1 : M
        fprintf(1,'Escriba la restriccion Res(%d) igualada a cero (solo
el primer miembro)\n',I);
        fprintf(1,'en terminos de y1, ..., y%d \n',N);

        Res(I) = input(' ','s');
    end;

    for I = 1 : M+N

        fprintf(1,'Escriba la aproximacion inicial X(%d)\n',I);
        X(I) = input(' ');
    end;
end;

OK = FALSE;
while OK == FALSE
    fprintf(1,'Escriba la tolerancia T.\n');
    T = input(' ');
    if T > 0
        OK = TRUE;
    else
        fprintf(1,'La tolerancia T debe ser positiva.\n');
    end;
end;

```

```

end;
OK = FALSE;
while OK == FALSE
    fprintf(1,'Escriba el numero maximo de iteraciones NN.\n');
    NN = input(' ');
    if NN > 0
        OK = TRUE;
    else
        fprintf(1,'El numero maximo de iteraciones NN debe ser un
entero positivo.\n');
    end;
end;

EP = zeros(1,NN);
fprintf(1,'Salida de datos\n');
fprintf(1,'1. Pantalla\n');
fprintf(1,'2. Fichero de texto\n');
fprintf(1,'Escriba 1 o 2\n');
FLAG = input(' ');
if FLAG == 2
    fprintf(1,'Escriba el nombre del fichero de la forma -
disco\\:nombre.ext\n');
    fprintf(1,'Por ejemplo   A:\\OUTPUT.DTA\n');
    NAME = input(' ','s');
    OUP = fopen(NAME,'wt');
else
    OUP = 1;
end;
fprintf(1,'Seleccione la cantidad de resultados que desea
visualizar\n');
fprintf(1,'1. Solo la respuesta \n');
fprintf(1,'2. Todas las aproximaciones intermedias\n');
fprintf(1,'Escriba 1 o 2\n');
FLAG = input(' ');
fprintf(OUP, 'METODO DE OPTIMIZACION POR MULTIPLICADORES DE
LAGRANGE\n\n');
if FLAG == 2
    fprintf(OUP, 'Iteracion, Aproximacion, Error\n');
end;

% PASO 1
OK = TRUE;
K = 1;
h = 1e-4;
% PASO 2
while OK == TRUE & K <= NN
    A = zeros(N+M,N+M+1);
    FF = zeros (1,N+M);
% PASO 3
%     if K == 1
%         for I = 1 : N+M

%                 for J = 1 : N+M

%                     if N == 2

%                         if I == J

```

```

if I <=N
    q = inline(char(f), 'y1', 'y2');
    X(I) = X(I)+h;
    A(I,J) = q(X(1),X(2));
    X(I) = X(I)-2*h;
    EE = q(X(1),X(2));
    A(I,J) = A(I,J)+EE;
    X(I) = X(I)+h;
    EE = q(X(1),X(2));
    A(I,J) = (A(I,J)-2*EE)/(h^2);
end;

for Q = 1 : M

    q = inline(char(Res(Q)), 'y1', 'y2');
    X(I) = X(I)+h;
    RR = q(X(1),X(2));
    RR = X(N+Q)*RR;
    X(I) = X(I)-2*h;
    EE = q(X(1),X(2));
    EE = X(N+Q)*EE;
    X(I) = X(I)+h;
    A(I,J) = A(I,J) + ((RR + EE)/(h^2));
    EE = q(X(1),X(2));
    EE = X(N+Q)*EE;
    A(I,J) = A(I,J) - (2*EE/(h^2));

end;

else
    if I<=N & J<=N

        q = inline(char(f), 'y1', 'y2');
        EE = q(X(1),X(2));
        A(I,J) = -2*EE;
        X(I) = X(I)+h;
        EE = q(X(1),X(2));
        A(I,J) = A(I,J) + EE;
        X(J) = X(J)+h;
        EE = q(X(1),X(2));
        A(I,J) = A(I,J) + EE;
        X(J) = X(J) - 2*h;
        EE = q(X(1),X(2));
        A(I,J) = A(I,J) -2*EE;
        X(I) = X(I) - h;
        EE = q(X(1),X(2));
        A(I,J) = A(I,J) + EE;
        X(I) = X(I) - h;
        EE = q(X(1),X(2));
        A(I,J) = A(I,J) + EE;
        X(J) = X(J) + h;
        EE = q(X(1),X(2));
        A(I,J) = A(I,J) + EE;
        X(J) = X(J) + h;
        EE = q(X(1),X(2));
    end
end

```



```

        A(I,J) = A(I,J) -2*EE;
        X(I) = X(I)+ h;
        EE = q(X(1),X(2));
        A(I,J) = (A(I,J) + EE)/(6*(h^2));
        X(J)=X(J)-h;
    end;

    for Q = 1 : M

        q = inline(char(Res(Q)), 'y1', 'y2');
        EE = X(N+Q)*q(X(1),X(2));
        A(I,J) = A(I,J)*(6*(h^2))-2*EE;
        X(I) = X(I)+h;
        EE = X(N+Q)*q(X(1),X(2));
        A(I,J) = A(I,J) + EE;
        X(J) = X(J)+h;
        EE = X(N+Q)*q(X(1),X(2));
        A(I,J) = A(I,J) + EE;
        X(J) = X(J) - 2*h;
        EE = X(N+Q)*q(X(1),X(2));
        A(I,J) = A(I,J) -2*EE;
        X(I) = X(I) - h;
        EE = X(N+Q)*q(X(1),X(2));
        A(I,J) = A(I,J) + EE;
        X(I) = X(I) - h;
        EE = X(N+Q)*q(X(1),X(2));
        A(I,J) = A(I,J) + EE;
        X(J) = X(J) + h;
        EE = X(N+Q)*q(X(1),X(2));
        A(I,J) = A(I,J) + EE;
        X(J) = X(J) + h;
        EE = X(N+Q)*q(X(1),X(2));
        A(I,J) = A(I,J) -2*EE;
        X(I) = X(I)+ h;
        EE = X(N+Q)*q(X(1),X(2));
        A(I,J) = (A(I,J) + EE)/(6*(h^2));
        X(J)=X(J)-h;
    end;
end;

elseif N == 3

    if I == J
        if I<=N
            q = inline(char(f), 'y1', 'y2', 'y3');
            X(I) = X(I)+h;
            A(I,J) = q(X(1),X(2),X(3));
            X(I) = X(I)-2*h;
            EE = q(X(1),X(2),X(3));
            A(I,J) = A(I,J)+EE;
            X(I) = X(I)+h;
            EE = q(X(1),X(2),X(3));
            A(I,J) = A(I,J)-2*EE;
        end;

        for Q = 1 : M

```

```

                                q =
inline(char(Res(Q)), 'y1', 'y2', 'y3');
                                X(I) = X(I)+h;
                                RR = q(X(1),X(2),X(3));
                                RR = X(N+Q)*RR;
                                X(I) = X(I)-2*h;
                                EE = q(X(1),X(2),X(3));
                                EE = X(N+Q)*EE;
                                X(I) = X(I)+h;
                                A(I,J) = A(I,J) + RR + EE;
                                EE = q(X(1),X(2),X(3));
                                EE = X(N+Q)*EE;
                                A(I,J) = (A(I,J) - 2*EE)/(h^2);

                                end;

                                else
                                if I<=N & J<=N
                                q = inline(char(f), 'y1', 'y2', 'y3');
                                EE = q(X(1),X(2),X(3));
                                A(I,J) = -2*EE;
                                X(I) = X(I)+h;
                                EE = q(X(1),X(2),X(3));
                                A(I,J) = A(I,J) + EE;
                                X(J) = X(J)+h;
                                EE = q(X(1),X(2),X(3));
                                A(I,J) = A(I,J) + EE;
                                X(J) = X(J) - 2*h;
                                EE = q(X(1),X(2),X(3));
                                A(I,J) = A(I,J) -2*EE;
                                X(I) = X(I) - h;
                                EE = q(X(1),X(2),X(3));
                                A(I,J) = A(I,J) + EE;
                                X(I) = X(I) - h;
                                EE = q(X(1),X(2),X(3));
                                A(I,J) = A(I,J) + EE;
                                X(J) = X(J) + h;
                                EE = q(X(1),X(2),X(3));
                                A(I,J) = A(I,J) + EE;
                                X(J) = X(J) + h;
                                EE = q(X(1),X(2),X(3));
                                A(I,J) = A(I,J) -2*EE;
                                X(I) = X(I)+ h;
                                EE = q(X(1),X(2),X(3));
                                A(I,J) = (A(I,J) + EE)/(6*(h^2));
                                X(J)=X(J)-h;

                                end;

                                for Q = 1 : M

                                q =
inline(char(Res(Q)), 'y1', 'y2', 'y3');
                                EE = X(N+Q)*q(X(1),X(2),X(3));
                                A(I,J) = A(I,J)*(6*(h^2))-2*EE;
                                X(I) = X(I)+h;
                                EE = X(N+Q)*q(X(1),X(2),X(3));

```

```

A(I,J) = A(I,J) + EE;
X(J) = X(J)+h;
EE = X(N+Q)*q(X(1),X(2),X(3));
A(I,J) = A(I,J) + EE;
X(J) = X(J) - 2*h;
EE = X(N+Q)*q(X(1),X(2),X(3));
A(I,J) = A(I,J) -2*EE;
X(I) = X(I) - h;
EE = X(N+Q)*q(X(1),X(2),X(3));
A(I,J) = A(I,J) + EE;
X(I) = X(I) - h;
EE = X(N+Q)*q(X(1),X(2),X(3));
A(I,J) = A(I,J) + EE;
X(J) = X(J) + h;
EE = X(N+Q)*q(X(1),X(2),X(3));
A(I,J) = A(I,J) + EE;
X(J) = X(J) + h;
EE = X(N+Q)*q(X(1),X(2),X(3));
A(I,J) = A(I,J) -2*EE;
X(I) = X(I)+ h;
EE = X(N+Q)*q(X(1),X(2),X(3));
A(I,J) = (A(I,J) + EE)/(6*(h^2));
X(J)=X(J)-h;

end;

end;

elseif N == 4

    if I == J
        if I <=N
            q =
inline(char(f),'y1','y2','y3','y4');
            X(I) = X(I)+h;
            A(I,J) = q(X(1),X(2),X(3),X(4));
            X(I) = X(I)-2*h;
            EE = q(X(1),X(2),X(3),X(4));
            A(I,J) = A(I,J)+EE;
            X(I) = X(I)+h;
            EE = q(X(1),X(2),X(3),X(4));
            A(I,J) = (A(I,J)-2*EE)/(h^2);
        end;

    for Q = 1 : M

        q =
inline(char(Res(Q)),'y1','y2','y3','y4');
        X(I) = X(I)+h;
        RR = q(X(1),X(2),X(3),X(4));
        RR = X(N+Q)*RR;
        X(I) = X(I)-2*h;
        EE = q(X(1),X(2),X(3),X(4));
        EE = X(N+Q)*EE;
        X(I) = X(I)+h;

```

```

A(I,J) = A(I,J) + (RR + EE)/(h^2);
EE = q(X(1),X(2),X(3),X(4));
EE = X(N+Q)*EE;
A(I,J) = A(I,J) - 2*(EE/(h^2));

end;

else
    if I<=N & J<=N
        q =
inline(char(f), 'y1', 'y2', 'y3', 'y4');
        EE = q(X(1),X(2),X(3),X(4));
        A(I,J) = -2*EE;
        X(I) = X(I)+h;
        EE = q(X(1),X(2),X(3),X(4));
        A(I,J) = A(I,J) + EE;
        X(J) = X(J)+h;
        EE = q(X(1),X(2),X(3),X(4));
        A(I,J) = A(I,J) + EE;
        X(J) = X(J) - 2*h;
        EE = q(X(1),X(2),X(3),X(4));
        A(I,J) = A(I,J) -2*EE;
        X(I) = X(I) - h;
        EE = q(X(1),X(2),X(3),X(4));
        A(I,J) = A(I,J) + EE;
        X(I) = X(I) - h;
        EE = q(X(1),X(2),X(3),X(4));
        A(I,J) = A(I,J) + EE;
        X(J) = X(J) + h;
        EE = q(X(1),X(2),X(3),X(4));
        A(I,J) = A(I,J) + EE;
        X(J) = X(J) + h;
        EE = q(X(1),X(2),X(3),X(4));
        A(I,J) = A(I,J) -2*EE;
        X(I) = X(I)+ h;
        EE = q(X(1),X(2),X(3),X(4));
        A(I,J) = (A(I,J) + EE)/(6*(h^2));
        X(J)=X(J)-h;
    end;

    for Q = 1 : M
        q =
inline(char(Res(Q)), 'y1', 'y2', 'y3', 'y4');
        EE = X(N+Q)*q(X(1),X(2),X(3),X(4));
        A(I,J) = A(I,J)*(6*(h^2))-2*EE;
        X(I) = X(I)+h;
        EE = X(N+Q)*q(X(1),X(2),X(3),X(4));
        A(I,J) = A(I,J) + EE;
        X(J) = X(J)+h;
        EE = X(N+Q)*q(X(1),X(2),X(3),X(4));
        A(I,J) = A(I,J) + EE;
        X(J) = X(J) - 2*h;
        EE = X(N+Q)*q(X(1),X(2),X(3),X(4));
        A(I,J) = A(I,J) -2*EE;
        X(I) = X(I) - h;
        EE = X(N+Q)*q(X(1),X(2),X(3),X(4));
        A(I,J) = A(I,J) + EE;
    end;
end;

```

```

X(I) = X(I) - h;
EE = X(N+Q)*q(X(1),X(2),X(3),X(4));
A(I,J) = A(I,J) + EE;
X(J) = X(J) + h;
EE = X(N+Q)*q(X(1),X(2),X(3),X(4));
A(I,J) = A(I,J) + EE;
X(J) = X(J) + h;
EE = X(N+Q)*q(X(1),X(2),X(3),X(4));
A(I,J) = A(I,J) -2*EE;
X(I) = X(I)+ h;
EE = X(N+Q)*q(X(1),X(2),X(3),X(4));
A(I,J) = (A(I,J) + EE)/(6*(h^2));
X(J)=X(J)-h;

end;

end;

else

    if I == J
        if I<=N
            q =
inline(char(f),'y1','y2','y3','y4','y5');
X(I) = X(I)+h;
A(I,J) = q(X(1),X(2),X(3),X(4),X(5));
X(I) = X(I)-2*h;
EE = q(X(1),X(2),X(3),X(4),X(5));
A(I,J) = A(I,J)+EE;
X(I) = X(I)+h;
EE = q(X(1),X(2),X(3),X(4),X(5));
A(I,J) = (A(I,J)-2*EE)/(h^2);
end;

        for Q = 1 : M

            q =
inline(char(Res(Q)),'y1','y2','y3','y4','y5');
X(I) = X(I)+h;
RR = q(X(1),X(2),X(3),X(4),X(5));
RR = X(N+Q)*RR;
X(I) = X(I)-2*h;
EE = q(X(1),X(2),X(3),X(4),X(5));
EE = X(N+Q)*EE;
X(I) = X(I)+h;
A(I,J) = A(I,J) + (RR + EE)/(h^2);
EE = q(X(1),X(2),X(3),X(4),X(5));
EE = X(N+Q)*EE;
A(I,J) = A(I,J) - 2*(EE/(h^2));

end;

else

```

```

        if I<=N & J<=N
            q =
inline(char(f), 'y1', 'y2', 'y3', 'y4', 'y5');
            EE = q(X(1),X(2),X(3),X(4),X(5));
            A(I,J) = -2*EE;
            X(I) = X(I)+h;
            EE = q(X(1),X(2),X(3),X(4),X(5));
            A(I,J) = A(I,J) + EE;
            X(J) = X(J)+h;
            EE = q(X(1),X(2),X(3),X(4),X(5));
            A(I,J) = A(I,J) + EE;
            X(J) = X(J) - 2*h;
            EE = q(X(1),X(2),X(3),X(4),X(5));
            A(I,J) = A(I,J) -2*EE;
            X(I) = X(I) - h;
            EE = q(X(1),X(2),X(3),X(4),X(5));
            A(I,J) = A(I,J) + EE;
            X(I) = X(I) - h;
            EE = q(X(1),X(2),X(3),X(4),X(5));
            A(I,J) = A(I,J) + EE;
            X(J) = X(J) + h;
            EE = q(X(1),X(2),X(3),X(4),X(5));
            A(I,J) = A(I,J) + EE;
            X(J) = X(J) + h;
            EE = q(X(1),X(2),X(3),X(4),X(5));
            A(I,J) = A(I,J) -2*EE;
            X(I) = X(I)+ h;
            EE = q(X(1),X(2),X(3),X(4),X(5));
            A(I,J) = (A(I,J) + EE)/(6*(h^2));
            X(J)=X(J)-h;
        end;

        for Q = 1 : M

            q =
inline(char(Res(Q)), 'y1', 'y2', 'y3', 'y4', 'y5');
            EE =
X(N+Q)*q(X(1),X(2),X(3),X(4),X(5));
            A(I,J) = A(I,J)*(6*(h^2))-2*EE;
            X(I) = X(I)+h;
            EE =
X(N+Q)*q(X(1),X(2),X(3),X(4),X(5));
            A(I,J) = A(I,J) + EE;
            X(J) = X(J)+h;
            EE =
X(N+Q)*q(X(1),X(2),X(3),X(4),X(5));
            A(I,J) = A(I,J) + EE;
            X(J) = X(J) - 2*h;
            EE =
X(N+Q)*q(X(1),X(2),X(3),X(4),X(5));
            A(I,J) = A(I,J) -2*EE;
            X(I) = X(I) - h;
            EE =
X(N+Q)*q(X(1),X(2),X(3),X(4),X(5));
            A(I,J) = A(I,J) + EE;
            X(I) = X(I) - h;
            EE =
X(N+Q)*q(X(1),X(2),X(3),X(4),X(5));

```



```

elseif N == 3

    for I = 1 : N

        q = inline(char(f), 'y1', 'y2', 'y3');
        X(I) = X(I)+h;
        FF(I) = q(X(1),X(2),X(3));
        X(I) = X(I)-2*h;
        EE = q(X(1),X(2),X(3));
        X(I) = X(I)+h;
        FF(I) = (FF(I) - EE)/(2*h);
    end;

    for J = 1 : M

        for I = 1 : N+M

            q = inline(char(Res(J)), 'y1', 'y2', 'y3');
            X(I) = X(I)+h;
            RR = q(X(1),X(2),X(3));
            RR = X(N+J)*RR;
            X(I) = X(I)-2*h;
            EE = q(X(1),X(2),X(3));
            EE = X(N+J)*EE;
            X(I) = X(I)+h;
            FF(I) = FF(I) + (RR - EE)/(2*h);

        end;
    end;

elseif N == 4

    for I = 1 : N

        q = inline(char(f), 'y1', 'y2', 'y3', 'y4');
        X(I) = X(I)+h;
        FF(I) = q(X(1),X(2),X(3),X(4));
        X(I) = X(I)-2*h;
        EE = q(X(1),X(2),X(3),X(4));
        X(I) = X(I)+h;
        FF(I) = (FF(I) - EE)/(2*h);
    end;

    for J = 1 : M

        for I = 1 : N+M

            q = inline(char(Res(J)), 'y1', 'y2', 'y3', 'y4');
            X(I) = X(I)+h;
            RR = q(X(1),X(2),X(3),X(4));
            RR = X(N+J)*RR;
            X(I) = X(I)-2*h;
            EE = q(X(1),X(2),X(3),X(4));
            EE = X(N+J)*EE;
            X(I) = X(I)+h;

```



```

        FF(I) = FF(I) + (RR - EE)/(2*h);

    end;
end;

else

    for I = 1 : N

        q = inline(char(f), 'y1', 'y2', 'y3', 'y4', 'y5');
        X(I) = X(I)+h;
        FF(I) = q(X(1),X(2),X(3),X(4),X(5));
        X(I) = X(I)-2*h;
        EE = q(X(1),X(2),X(3),X(4),X(5));
        X(I) = X(I)+h;
        FF(I) = (FF(I) - EE)/(2*h);
    end;

    for J = 1 : M

        for I = 1 : N+M

            q = inline(char(Res(J)), 'y1', 'y2', 'y3', 'y4', 'y5');
            X(I) = X(I)+h;
            RR = q(X(1),X(2),X(3),X(4),X(5));
            RR = X(N+J)*RR;
            X(I) = X(I)-2*h;
            EE = q(X(1),X(2),X(3),X(4),X(5));
            EE = X(N+J)*EE;
            X(I) = X(I)+h;
            FF(I) = FF(I) + (RR - EE)/(2*h);

        end;
    end;
end;

for I = 1 : N+M

    A(I,N+M+1) = -FF(I);

end;

if K==1
    R = 0;
    for I = 1 : N+M
        if abs(FF(I)) > R
            R = abs(FF(I));
        end;
    end;
    EP(K)=R;
    if FLAG == 2
        fprintf(OUP, ' %2d', K-1);
        for I = 1 : N+M
            fprintf(OUP, ' %11.8f', X(I));
        end;
        fprintf(OUP, '\n%12.6e\n', R);
    end;
end;

```

```

        end;

% PASO 4
K1 = N+M-1;
OK = TRUE;
I1 = 1;
while OK == TRUE & I1 <= K1
    Z1 = abs(A(I1,I1));
    IR1 = I1;
    IA1 = I1+1;
    for J1 = IA1 : N+M
        if abs(A(J1,I1)) > Z1
            IR1 = J1;
            Z1 = abs(A(J1,I1));
        end;
    end;

    if Z1 <= 1.0e-20
        OK = FALSE;
    else
        if IR1 ~= I1
            for J1 = I1 : N+M+1
                C1 = A(I1,J1);
                A(I1,J1) = A(IR1,J1);
                A(IR1,J1) = C1;
            end;
        end;
        for J1 = IA1 : N+M
            C1 = A(J1,I1)/A(I1,I1);
            if abs(C1) <= 1.0e-20
                C1 = 0;
            end;
            for L1 = I1 : N+M+1
                A(J1,L1) = A(J1,L1)-C1*A(I1,L1);
            end;
        end;
    end;
    I1 = I1+1;
end;

if OK == TRUE
    if abs(A(N+M,N+M)) <= 1.0e-20
        OK = FALSE;
    else
        Y(N+M) = A(N+M,N+M+1)/A(N+M,N+M);
        for I1 = 1 : K1
            J1 = N+M-I1;
            JA1 = J1+1;
            C1 = A(J1,N+M+1);
            for L1 = JA1 : N+M
                C1 = C1-A(J1,L1)*Y(L1);
            end;
            Y(J1) = C1/A(J1,J1);
        end;
    end;
end;

if OK == FALSE
    fprintf(1,'El sistema lineal es singular\n');
end;

```

```

        end;

        if OK == TRUE
% PASO 5
            R = 0;
            for I = 1 : N+M
                if abs(Y(I)) > R
                    R = abs(Y(I));
                end;
                X(I) = X(I)+Y(I);
            end;
            EP(K)=R;
            if FLAG == 2
                fprintf(OUP, ' %2d', K);
                for I = 1 : N+M
                    fprintf(OUP, ' %11.8f', X(I));
                end;
                fprintf(OUP, '\n%12.6e\n', R);
            end;

% PASO 6
            if R <= T
                OK = FALSE;
                fprintf(OUP, 'La solucion en la iteracion %d es:\n\n', K);
                for I = 1 : N+M
                    fprintf(OUP, ' %11.8f', X(I));
                end;
                fprintf(OUP, '\n\n con una tolerancia %.10e\n', T);

% PASO 7
            else
                K = K+1;
            end;

        end;

        end;
        T=0;
        if K>NN
            T=K;
            K=NN;
        end;
        EP=EP(1:K);
        figure(1);
        semilogy(EP,'--
rs','LineWidth',2,'MarkerEdgeColor','k','MarkerFaceColor','g','MarkerSize',
5);
        title('METODO DE NEWTON-RAPHSON');
        xlabel('Iteraciones');
        ylabel('Error');
        if T > NN
% PASO 8
            fprintf(OUP, 'El proceso no converge en %d iteraciones\n', NN);
        end;
        if OUP ~= 1
            fclose(OUP);
            fprintf(1,'Fichero de salida %s creado satisfactoriamente
\n',NAME);
        end;

```

EJEMPLOS

1) Una ventana está formada por un rectángulo rematado con un semicírculo en la parte superior. Si el marco ha de tener una longitud de 10m, determinar sus dimensiones para que la superficie de la ventana sea máxima.

La función a optimizar es:

$$A = \pi \frac{D^2}{8} + Dx$$

s.a.

$$\pi \frac{D}{2} + 2x + D = 10$$

Las soluciones del problema son:

$$Y1 = D = 2.8\text{m}$$

$$Y2 = x = 1.4\text{m}$$

$$A = 7\text{m}$$

Para resolver el problema con nuestra subrutina, lo hacemos de la siguiente manera:

Este es el metodo de Optimizacion funciones no lineales por Multiplicadores de Lagrange.

La funcion y las restricciones pueden ser escritas ahora o leidas de un fichero.

El numero de variables de la funcion no puede exceder de 5.

En otro caso habria que modificar el programa.

Escriba el numero de variables N.

(Introducimos el número de variables que tiene la función a optimizar) en nuestro caso 2.

Escriba el numero de restricciones M.

(Introducimos el número de restricciones a las que está sujeta la función, siempre debe ser menor que N) en nuestro caso 1.

Entrada de la funcion

1. Pantalla

2. Fichero de texto

Escriba 1 o 2

(Nos da la opción de introducir los datos por fichero o por pantalla) en nuestro caso 1 (Pantalla).

Escriba la función F en términos de $y_1 \dots y_2$

(Escribimos la función a optimizar, sustituyendo las variables por y_1, y_2, \dots) en nuestro caso $\pi \cdot (y_1^2/8) + y_1 \cdot y_2$

Escriba la restricción $\text{Res}(1)$ igualada a cero (solo el primer miembro)

en términos de y_1, \dots, y_2

(Procedemos de la misma forma con la restricción, teniendo en cuenta que debe estar igualada a 0 y sólo colocaremos el primer miembro de la igualdad) en nuestro caso $\pi \cdot (y_1/2) + 2 \cdot y_2 + y_1 - 10$

Escriba la aproximación inicial $X(1)$

0

Escriba la aproximación inicial $X(2)$

0

Escriba la aproximación inicial $X(3)$

0

(Las aproximaciones iniciales son los valores que van a tomar inicialmente las variables para el método iterativo) en nuestro caso 0.

Escriba la tolerancia T .

(La tolerancia es el máximo error permitido en la aproximación de una solución) en nuestro caso $1e-3$.

Escriba el número máximo de iteraciones NN .

(Es el número de iteraciones en el que el programa finalizará si no ha encontrado solución anteriormente. Si el programa llega a este número máximo, el sistema no converge) en nuestro caso 100.

Salida de datos

1. Pantalla

2. Fichero de texto

Escriba 1 o 2

(El programa nos da la posibilidad de obtener los resultados en la pantalla de comandos de matlab o bien en un fichero cuya ubicación introducimos también) en nuestro caso 1 (Pantalla).

Seleccione la cantidad de resultados que desea visualizar

1. Solo la respuesta

2. Todas las aproximaciones intermedias

Escriba 1 o 2

(Podemos elegir si visualizar una por una todas las iteraciones o bien visualizar solamente la iteración final en la que se llega al resultado) en nuestro caso 2 (Todas las aproximaciones intermedias).

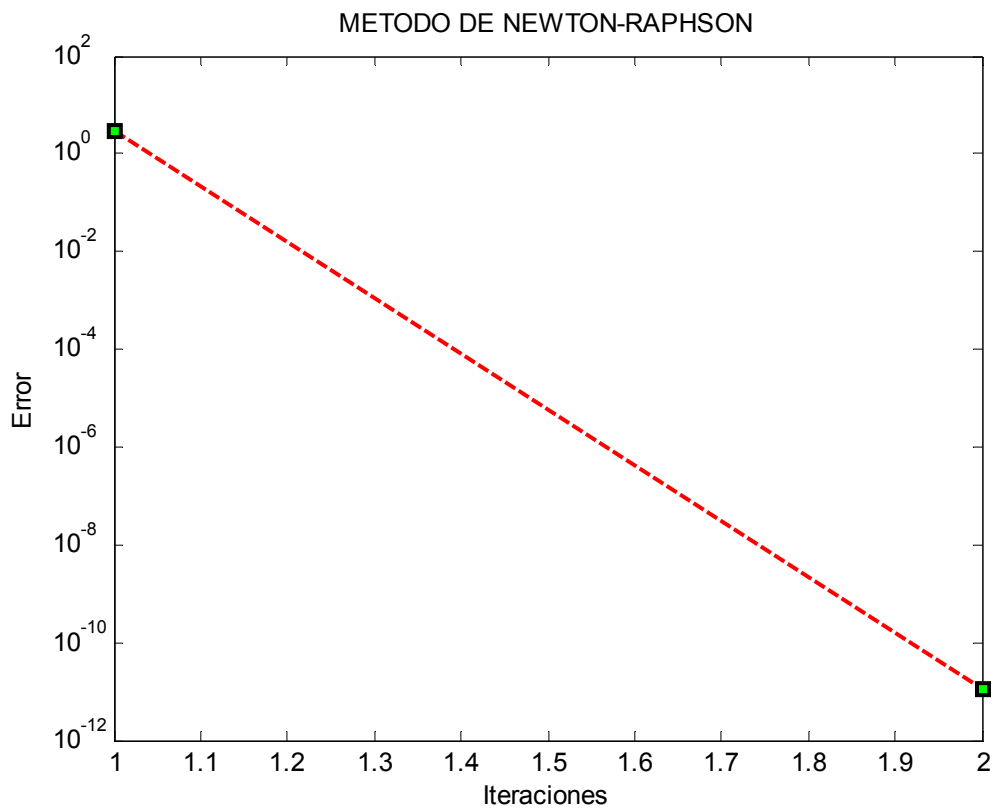
METODO DE OPTIMIZACION POR MULTIPLICADORES DE LAGRANGE

Iteracion, Aproximacion, Error
0 0.00000000 0.00000000 0.00000000
1.000000e+001
1 2.80049577 1.40024788 -1.40024788
2.800496e+000
2 2.80049577 1.40024788 -1.40024788
1.142447e-011

La solucion en la iteracion 2 es:

2.80049577 1.40024788 -1.40024788

con una tolerancia 1.0000000000e-003



Podemos ver que los resultados obtenidos son correctos.

2) Dado el programa: Minimizar $f = x^2 - y^2$ (calcular su mínimo local)

s.a: $x^2 - y = 1$

Las soluciones de este problema son:

$Y1=x=0$

$Y2=y=-1$

$F=-1$

El problema lo resolvemos de la siguiente manera:

Este es el metodo de Optimizacion funciones no lineales por Multiplicadores de Lagrange.

La funcion y las restricciones pueden ser escritas ahora o leidas de un fichero.

El numero de variables de la funcion no puede exceder de 5.

En otro caso habria que modificar el programa.

Escriba el numero de variables N.

2

Escriba el numero de restricciones M.

1

Entrada de la funcion

1. Pantalla

2. Fichero de texto

Escriba 1 o 2

1

Escriba la funcion F en terminos de $y1 \dots y2$

$y1^2 - y2^2$

Escriba la restriccion Res(1) igualada a cero (solo el primer miembro)
en terminos de $y1, \dots, y2$

$y1^2 - y2 - 1$

Escriba la aproximacion inicial X(1)

0

Escriba la aproximacion inicial X(2)

0

Escriba la aproximacion inicial X(3)

0

Escriba la tolerancia T.

1e-3

Escriba el numero maximo de iteraciones NN.

100

Salida de datos

1. Pantalla

2. Fichero de texto

Escriba 1 o 2

1

Seleccione la cantidad de resultados que desea visualizar

1. Solo la respuesta
2. Todas las aproximaciones intermedias

Escriba 1 o 2

2

METODO DE OPTIMIZACION POR MULTIPLICADORES DE LAGRANGE

Iteracion, Aproximacion, Error

0 0.00000000 0.00000000 0.00000000

1.000000e+000

1 0.00000000 -1.00000000 2.00000000

2.000000e+000

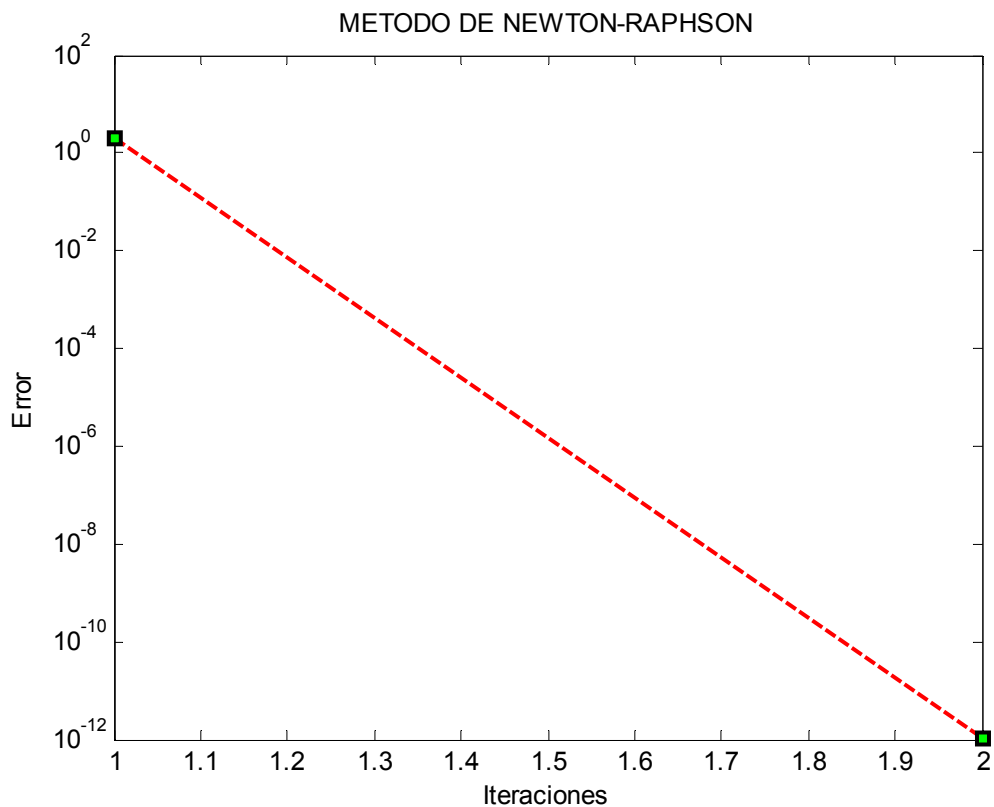
2 0.00000000 -1.00000000 2.00000000

1.095568e-012

La solucion en la iteracion 2 es:

0.00000000 -1.00000000 2.00000000

con una tolerancia 1.0000000000e-003



BIBLIOGRAFÍA

<http://sistemas.itlp.edu.mx/tutoriales/investoper2/>

<http://www.uaem.mx/posgrado/mcruz/maximominimo.pdf>

http://www.elprisma.com/apuntes/ingenieria_industrial/optimizacion/default.asp

<http://www.ica.luz.ve/~lzerpa/optimizacionparaingenieros/introduccion-repaso.pdf>

<http://home.ubalt.edu/ntsbarsh/opre640s/spanishd.htm>

<http://es.wikipedia.org/wiki/optimizaci%c3%b3n>

http://exa.unne.edu.ar/investigacion/calculo2/public_html/anamat1_doc/