UNIVERSIDAD NACIONAL AUTÓNOMA DE NICARAGUA
FACULTAD DE CIENCIAS
DEPARTAMENTO DE MATEMÁTICA Y ESTADÍSTICA

MINIMIZACIÓN NUMÉRICA DE FUNCIONES DE UNA Y VARIAS VARIABLES

MONOGRAFÍA

PARA OPTAR AL TÍTULO DE LICENCIADO EN MATEMÁTICA

PRESENTADA POR:

BRA. MARTHA LIGIA GARCÍA CABRERA
BRA. KAREN JOSÉ SEVILLA ARÁUZ
BR. ELVIN DE JESÚS PÉREZ RIVERA

TUTORA: MSc. ANGELA ALTAMIRANO SLINGER

LEÓN, SEPTIEMBRE 2007

DEDICATORIA

A DIOS sobre todas las cosas por haberme dado la oportunidad y la fortaleza que he tenido, porque nunca me ha desamparado siempre ha estado conmigo en los momentos mas difíciles y cruciales a través del estudio en esta investigación y gracias le doy a él por la capacidad que ha dado de poder entenderle y comprenderles a todas los profesores que me brindaron su apoyo para hoy estar optando a mi título.

A mis padres por estar en esta lucha conmigo, porque juntos hemos luchado armónicamente venciendo todo obstáculo que se atravesó en nuestro camino y por lo cual bendigo a DIOS por haberme dado unos padres maravillosos.

Martha Ligia García Cabrera

DEDICATORIA

A Dios por ser tan bello y maravilloso, mi guía y protector en los momentos buenos y malos porque, sin tu ayuda no sé como hubiera terminado este trabajo de investigación. ¡Gracias Padre, por tenerme tanto aprecio!

A mis padres por ser los más importantes en mi vida, las personas que me han motivado, acompañado y ayudado en los momentos más difíciles. ¡Gracias Dios mío, por habérmelos dado tal y como son!

A mis hermanas por ser muy especiales para mí, mis amigas y consejeras a quien también les debo lo que ahora soy, gracias por brindarme su apoyo incondicional.

Karen José Sevilla Aráuz

DEDICATORIA

Dedico esta tesis a **Dios** todo poderoso, a nuestro **Señor Jesucristo** y a la **Santísima Virgen María** por haberme dado el tiempo y la vida para hacer posible la culminación de mi carrera profesional.

A mi hermana **Sonia Luz Pérez Rivera** a quien considero como una madre y un ejemplo a seguir y que con su sacrificio, cariño y comprensión fue la guía de mis pasos durante todo el transcurso de mi vida como estudiante y persona; llegando así alcanzar la meta propuesta de ser un hombre de bien y un profesional exitoso.

A mi amada esposa **María Leonor Astacio Roque** la persona más importante de mi vida, que con su amor, ternura y apoyo constante me alienta todos los días a seguir luchando y así poder compartir el logro final en mi carrera que es obtener el título de Licenciatura en Matemática.

A mi madre **María Teresa Rivera** y mi hermana **Juanita** quienes también han sido un apoyo incondicional y me han brindado su cariño en los momentos difíciles.

Elvin de Jesús Pérez Rivera

AGRADECIMIENTO

Agradezco a **Dios** por haber estado como luz en nuestro camino, guiándonos y acompañándonos en cada momento con nuestro trabajo de investigación que gracias a él, pudimos concluirlo.

A la **Msc. Angela Altamirano Slinger** por habernos dedicado un poco de su valioso tiempo, ya que sin su orientación no hubiéramos podido concluir este trabajo.

A todas las personas que de una u otra manera nos apoyaron en la realización de este trabajo monográfico, en particular al **MSc. Alberto Cerda**, **Sr. Alejandro Vázquez** y a todos los profesores que nos brindaron sus conocimientos durante la carrera.

INDICE

INTRODUCCIÓN

En la actualidad las ciencias matemáticas se enfrentan al reto de mostrar su gran valor en la búsqueda de soluciones a problemas reales y la optimización es una de las áreas de estudio que ha avanzado ampliamente en esta dirección.

En los cursos de Optimización que se imparten en nuestra Universidad se hace énfasis en el estudio de métodos analíticos para resolver diversos problemas. En el tratamiento de problemas reales, estos métodos no son suficientes para obtener una solución exacta por lo que es necesario hacer uso de métodos numéricos para alcanzar un resultado aproximado.

En el caso particular de problemas de optimización de funciones, en el que se requiere maximizar o minimizar funciones reales de una o varias variables, no es la excepción puesto que se tiene que emplear la minimización numérica en problemas tales como hacer una estimación por ajustes en Estadísticas, calcular el flujo potencial en Biología, estudiar la inversión sísmica en Geofísica o estudiar el desplazamiento de placas vibrantes en Ingeniería Mecánica.

Lo anterior plantea la necesidad del conocimiento de diversos métodos numéricos de minimización de funciones reales de una o varias variables para el abordaje de problemas como los señalados anteriormente. Por tal razón hemos considerado de mucho interés el desarrollar un trabajo monográfico que permita disponer de métodos numéricos que aproxime el mínimo de funciones reales de una o varias variables.

El resultado de nuestro trabajo puede ser utilizado como material de consulta por estudiantes, docentes e investigadores en la aplicación de la optimización numérica en la solución de problemas específicos.

El presente trabajo consta de tres capítulos. En el primer capítulo se incluyen los elementos básicos de las funciones de una o varias variables, terminología, conceptos y teoremas básicos que se requiere para el desarrollo de los siguientes capítulos. En el segundo capítulo se plantean algunos de los métodos numéricos para localizar el mínimo de una función real de una variable, se utilizan dos tipos de métodos de minimización: el método de búsqueda y método mediante derivadas. En el tercer capítulo se trata de minimizar una función de varias variables, utilizando dos tipos de métodos: el método de Nelder-Mead y el método del gradiente.

Los métodos se describen de una forma concisa y clara, se ilustran con ejemplos y se incluyen los algoritmos de dichos métodos en un seudo código de fácil comprensión. En los anexos se presentan los programas en lenguaje C, de los diferentes métodos utilizados en el desarrollo del trabajo así como sus corridas correspondientes.

OBJETIVOS

OBJETIVO GENERAL

- Aproximar mínimos de funciones reales de una o varias variables mediante métodos numéricos.

OBJETIVOS ESPECÍFICOS

- Minimizar funciones reales de una o varias variables sin restricciones.

- Implementar los diferentes métodos numéricos que minimizan funciones de una o varias variables en una computadora.

- Comparar la precisión en soluciones obtenidas al utilizar métodos numéricos para minimizar funciones.

- Analizar la utilidad en la aplicación de un método de Optimización numérica en comparación con un método analítico.

CAPÍTULO 1
ELEMENTOS BÁSICOS

1.1 INTRODUCCIÓN

El presente capítulo tiene un doble propósito introducir la terminología básica que se usa en todo el trabajo y revisar algunos de los métodos básicos usados en cálculo para localizar extremos de funciones de una o varias variables con el fin de compararlos de los métodos numéricos de minimización a estudiar.

1.2 FUNCIONES DE UNA VARIABLE

La siguiente definición presenta los términos usados para describir la variación que se da en las funciones.

Definición 1.1

Sea f una función definida en un intervalo I y sean x_1, x_2 dos números $x_1 < x_2$ que están en I.

(i) f es creciente en I si $f(x_1) < f(x_2)$ siempre que $x_1 < x_2$.

(ii) f es decreciente en I si $f(x_1) > f(x_2)$ siempre que $x_1 < x_2$.

(iii) f es constante en I si $f(x_1) = f(x_2)$ para todo x_1 y x_2.

A continuación se da la terminología que se usa para denotar los valores más grandes y los mas pequeños de una función en un intervalo.

Definición 1.2

Sea f una función definida en un intervalo I y sea c un número en I.

(i) f tiene un mínimo local en $x = c$, si $f(c) \leq f(x)$ para todo $x \in$ I. Se dice que $f(c)$ es un valor mínimo local de f.

(ii) f tiene un máximo local en $x = c$, si $f(c) \geq f(x)$ para todo x ϵ I. Se dice que $f(c)$ es un valor máximo local de f.

(iii) Si una función f tiene un máximo o un mínimo local en $x = c$ entonces se dice que f tiene un extremo local en $x = c$.

Teorema 1.1

Sea f una función continua en I = $[a, b]$ y derivable en (a, b).

(i) Si $f'(x) > 0$ para todo x en (a, b), entonces f es creciente en I.

(ii) Si $f'(x) < 0$ para todo x en (a, b), entonces f es decreciente en I.

Los números en los que la derivada es 0 ó no existe desempeñan un papel importante en la determinación de los máximos y mínimos de una función. Por este motivo se da un nombre especial a estos números.

Definición 1.3

Un número c en el dominio de una función f se llama punto crítico de f si $f'(c) = 0$ ó $f'(c)$ no existe.

Si una función tiene un extremo local, este debe darse en un punto crítico, pero no todo punto crítico corresponde a un extremo local.

A continuación enunciamos dos criterios para distinguir máximos y mínimos locales.

Teorema 1.2 (Criterio de la primera derivada).

Sea f una función continua en I = $[a, b]$ y derivable en todo punto x ϵ (a, b) excepto quizás en $x = c$ un punto crítico de f.

(i) Si $f'(x) < 0$ en (a, c) y $f'(x) > 0$ en (c, b) entonces, f tiene un mínimo local en c.

(ii) Si $f'(x) > 0$ en (a, c) y $f'(x) < 0$ en (c, b) entonces, f tiene un máximo local en c.

(iii) Si $f'(x)>0$ ó si $f'(x)<0$ para todo x en I excepto para $x=c$, entonces f no tiene un extremo local en c.

Teorema 1.3 (Criterio de la segunda derivada).

Sea f una función continua en $[a,b]$ y dos veces derivable en (a,b) que contiene a un punto crítico c y tal que $f'(c)=0$.

(i) Si $f''(c)>0$, entonces f tiene un mínimo local en c.

(ii) Si $f''(c)<0$, entones f tiene un máximo local en c.

(iii) Si $f''(c)=0$, entonces no se puede afirmar nada.

Si una función f cumple que $f(c)\leq f(x)$ para toda x en el dominio de f, entonces el minimo que se da en c además de ser local es un mínimo global o absoluto.

De forma similar si f cumple que $f(c)\geq f(x)$ para toda x en el dominio de f, entonces el máximo que se da en c además de ser local es un máximo global o absoluto.

Definición 1.4 (Función unimodal).

Se dice que una función f es unimodal en I = $[a,b]$, si existe un único número c є I tal que:

(i) f es decreciente en $[a,c]$

(ii) f es creciente en $[c,b]$

Lo que implica, en particular que f alcanza su mínimo global en c.

Si se sabe que f es unimodal en (a,b) entonces es posible sustituir el intervalo inicial por un subintervalo en el que f alcanza su mínimo.

Ejemplo:

Determinar los extremos locales de la función $f(x)=x^3+3x^2-1$ en $[-4,0]$

Solución:

La primera derivada es $f'(x)=3x^2+6x$
La segunda derivada es $f''(x)=6x+6$

Haciendo $f'(x)=0$ se tiene que
$$3x^2+6x=0$$
$$3x(x+2)=0$$

$$\begin{array}{c|c} 3x=0 & x+2=0 \\ x=\dfrac{0}{3}=0 & x=-2 \end{array}$$

Los puntos críticos son $x=0$ y $x=-2$, ambos pertenecen al intervalo $[-4,0]$.

Aplicando el criterio de la segunda derivada se tiene que $f''(0)=6(0)+6=6>0$. Entonces f tiene un mínimo local en $x=0$ y $f''(-2)=6(-2)+6=-12+6=-6<0$ entonces f tiene máximo local en $x=-2$.

Presentamos a continuación un ejemplo donde se requiere el uso de un método numérico de cálculo de raíces para determinar los puntos críticos.

Ejemplo:

Determinar los extremos locales de la función $g(x)=senx-\dfrac{x^2}{2}$ en $[0,\pi/2]$.

Solución:

La primera derivada es $g'(x)=\cos x-x$
La segunda derivada es $g''(x)=-senx-1$

Haciendo $g'(x)=0$ se tiene que $\cos x-x=0$.

Para encontrar las raíces de esta ecuación usaremos el método de la secante (ver el algoritmo y el programa en el Anexo 1).

Sea $f(x)=\cos x - x$

Como $f(0)=\cos 0 - 0 = 1 - 0 = 1$ y $f\left(\frac{\pi}{2}\right)=\cos\left(\frac{\pi}{2}\right)-\frac{\pi}{2}=0-\frac{\pi}{2}=-\frac{\pi}{2}$,

entonces existe una raíz de la ecuación $\cos x - x = 0$ en $[0,\pi/2]$.

Aplicando el método de la secante con $p_0 = 0.5$ y $p_1 = \frac{\pi}{4}$, con tres iteraciones se obtiene que $p = 0.739085$ es una raíz de la ecuación mencionada, o sea un punto critico de la función $g(x)= senx - \frac{x^2}{2}$.

Aplicando el criterio de la segunda derivada se tiene que: $g''(0.739085)=-sen(0.739085)-1=-0.673612-1=-1.673612<0$. Entonces g tiene un maximo local en $x = 0.739085$.

1.3 **FUNCIONES DE VARIAS VARIABLES**

La definición 1.2 puede extenderse a funciones de varias variables. Para funciones de dos variables se usa la siguiente definición:

Definición 1.5

Sea $f(x,y)$ una función definida en una región R en el plano x,y. Se dice que $f(x,y)$ tiene un mínimo local en el punto (p,q) cuando $f(p,q)\leq f(x,y)$ para cada punto (x,y) ϵ R.

Se dice que $f(x,y)$ tiene un máximo local en el punto (p,q) cuando $f(x,y)\leq f(p,q)$ para cada punto (x,y) ϵ R.

Si una función f tiene un máximo ó un mínimo local en (p,q) entonces se dice que f tiene un extremo local en (p,q).

A continuación se presentan notaciones comunes para las primeras derivadas parciales de una función $f(x,y)$.

Si $w = f(x,y)$ entonces

$$f_x = \frac{\partial f}{\partial x} \qquad f_y = \frac{\partial f}{\partial y}$$

$$f_x(x, y) = \frac{\partial}{\partial x} f(x, y) = \frac{\partial w}{\partial x} = w_x$$

$$f_y(x, y) = \frac{\partial}{\partial y} f(x, y) = \frac{\partial w}{\partial y} = w_y$$

Dado que f_x y f_y son también funciones de dos variables se pueden considerar sus primeras derivadas parciales.

Estas son las segundas derivadas parciales de f y se denota como sigue:

$$\frac{\partial}{\partial x} f_x = (f_x)_x = f_{xx} = \frac{\partial}{\partial x}\left(\frac{\partial f}{\partial x}\right) = \frac{\partial^2 f}{\partial x^2}$$

$$\frac{\partial}{\partial y} f_x = (f_x)_y = f_{xy} = \frac{\partial}{\partial y}\left(\frac{\partial f}{\partial x}\right) = \frac{\partial^2 f}{\partial y \partial x}$$

$$\frac{\partial}{\partial x} f_y = (f_y)_x = f_{yx} = \frac{\partial}{\partial x}\left(\frac{\partial f}{\partial y}\right) = \frac{\partial^2 f}{\partial x \partial y}$$

$$\frac{\partial}{\partial y} f_y = (f_y)_y = f_{yy} = \frac{\partial}{\partial y}\left(\frac{\partial f}{\partial y}\right) = \frac{\partial^2 f}{\partial y^2}$$

Como en el caso de las funciones de una variable los máximos y mínimos locales se pueden alcanzar en pares de números críticos.

Definición 1.6

Sea $z = f(x, y)$ tiene primeras derivadas parciales, entonces las soluciones de:

$$f_x(x, y) = 0 \qquad \text{y} \qquad f_y(x, y) = 0$$

se llaman punto crítico.

Para determinar los extremos locales de una función de dos variables se usa el criterio de la segunda derivada que es una extensión del teorema 1.3.

Teorema 1.4 (Criterio de la segunda derivada)

Supongamos que $f(x,y)$ así como sus derivadas parciales primera y segunda son continúa en la región R. Supongamos que (p,q) ϵ R es un punto crítico.

(i) Si $f_{xx}(p,q)\,f_{yy}(p,q)-f^2{}_{xy}(p,q)>0$ y $f_{xx}(p,q)>0$, entonces f tiene un mínimo local en (p,q).

(ii) Si $f_{xx}(p,q)\,f_{yy}(p,q)-f^2{}_{xy}(p,q)>0$ y $f_{xx}(p,q)<0$, entonces f tiene un máximo local en (p,q).

(iii) Si $f_{xx}(p,q)\,f_{yy}(p,q)-f^2{}_{xy}(p,q)<0$, entonces f no tiene un extremo local en (p,q); este punto es un punto de silla.

(iv) Si $f_{xx}(p,q)\,f_{yy}(p,q)-f^2{}_{xy}(p,q)=0$, entonces no se puede afirmar nada.

Ejemplo:

Sea $f(x,y)=x^2-4xy+y^3+4y$. Encontrar los extremos locales de *f*.

Solución:

Las primeras derivadas parciales de *f* son:

$$f_x(x,y)=2x-4y$$
$$f_y(x,y)=-4x+3y^2+4$$

Los puntos críticos son soluciones del sistema de ecuaciones:

$$\begin{aligned} 2x-4y &= 0 \\ -4x+3y^2+4 &= 0 \end{aligned}$$

Resolviendo el sistema se obtiene los puntos $(4,2)$ y $\left(\frac{4}{3},\frac{2}{3}\right)$.

Las segundas derivadas parciales de f son:

$$f_{xx}(x,y)=2$$
$$f_{yy}(x,y)=6y$$
$$f_{xy}(x,y)=-4$$

Luego $f_{xx}(4,2)f_{yy}(4,2)-f^2{}_{xy}(4,2)=(2)(12)-(-4)^2=8>0$ y $fxx(4,2)=2>0$ entonces f tiene un mínimo local en $(4,2)$ y su valor es $f(4,2)=0$.

Como $f_{xx}\left(\frac{4}{3},\frac{2}{3}\right)f_{yy}\left(\frac{4}{3},\frac{2}{3}\right)-f^2{}_{xy}\left(\frac{4}{3},\frac{2}{3}\right)=(2)(4)-(-4)^2=-8<0$, entonces f

no tiene un extremo local en $\left(\frac{4}{3},\frac{2}{3}\right)$.

Definición 1.7 (Gradiente)

Si $u=u(x,y)$ es una función de dos variables, determinemos un vector cuyas proyecciones sobre los ejes de coordenadas son los valores de las derivadas parciales $\frac{\partial u}{\partial x},\frac{\partial u}{\partial y}$ de esta función en el punto correspondiente:

$$gradu=\frac{\partial u}{\partial x}i+\frac{\partial u}{\partial y}j$$

Este vector se llama gradiente de la función $u=u(x,y)$.

1.4 INTERPOLACION Y POLINOMIO DE LAGRANGE

Una de las más importantes clases de funciones de R en R es la clase de los polinomios algebraicos.

Un polinomio es una función de la forma $p(x)=a_nx^n+a_{n-1}x^{n-1}+\ldots+a_1x+a_0$ donde n es un entero no negativo y $a_0,a_1,\ldots a_n$ son constantes reales.

Una razón de su importancia es que los polinomios aproximan uniformemente funciones continuas.

Teorema 1.5 (Teorema de aproximación de Weierstrass).

Supongamos que *f* es una función definida y continua [a, b]. Para cada $\epsilon > 0$ existe un polinomio $p(x)$ con la propiedad que $|f(x) - p(x)| < \epsilon$ para todo x en $[a,b]$.

A continuación se describe un polinomio aproximante que puede determinarse especificando algunos puntos en el plano por donde pasa.

Teorema 1.6 (Polinomio Interpolante de lagrange).

Si $x_{0,} x_{1,} \ldots, x_{n.}$ son $(n+1)$ números diferentes y f es una función cuyos valores están dados en estos números entonces existe un único polinomio *p* de grado a la más *n* con la propiedad de que: $f(x_k) = p(x_k)$. Para $k = 0,1,\Lambda\ ,n$.

$$p(x) = f(x_0) l_{n,0}(x) + f(x_1) l_{n,1} + \ldots + f(x_n) l_{n,n}(x) = \sum_{k=0}^{n} f(x_k) l_{n,k}(x).$$ Donde

$$l_{n,k}(x) = \frac{(x - x_0)(x - x_1)\ldots(x - x_{k-1})(x - x_{k+1})\ldots(x - x_n)}{(x_k - x_0)(x_k - x_1)\ldots(x_k - x_{k-1})(x_k - x_{k+1})\ldots(x_k - x_n)} = \prod_{\substack{i=0 \\ i \neq k}}^{n} \frac{(x - x_i)}{(x_k - x_i)}.$$

Para $k = 0,1,\Lambda\ ,n$.

Escribiremos $l_{n,k}(x)$ simplemente como $l_k(x)$ cuando no haya confusión de grado.

En el caso particular del polinomio Interpolante de grado dos, se tiene que este polinomio está dado por:

$$f(x) = f(x_0) l_0(x) + f(x_1) l_1(x) + f(x_2) l_2(x).$$ Donde

$$l_0(x) = \frac{(x - x_1)(x - x_2)}{(x_0 - x_1)(x_0 - x_2)}$$

$$l_1(x) = \frac{(x - x_0)(x - x_2)}{(x_1 - x_0)(x_1 - x_2)}$$

$$l_2(x)=\frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)}$$

Ejemplo:

Usando los números o nodos $x_0=2, x_1=2.5, x_2=4$. Encontrar el polinomio Interpolante de segundo grado para $f(x)=\frac{1}{x}$

$$l_0(x)=\frac{(x-2.5)(x-4)}{(2-2.5)(2-4)}=x^2-6.5x+10$$

$$l_1(x)=\frac{(x-2)(x-4)}{(2.5-2)(2.5-4)}=\frac{1}{3}(-4x^2+24x-32)$$

$$l_2(x)=\frac{(x-2)(x-2.5)}{(4-2)(4-2.5)}=\frac{1}{3}(x^2-4.5x+5)$$. Ya que

$$f(x_0)=f(2)=0.5$$
$$f(x_1)=f(2.5)=0.4$$
$$f(x_2)=f(4)=0.25$$

$$\begin{aligned}
f(x)&=f(x_0)l_0(x)+f(x_1)l_1(x)+f(x_2)l_2(x)\\
&=0.5\left(x^2-6.5x+10\right)+\frac{0.4}{3}\left(-4x^2+24x-32\right)+\frac{0.25}{3}\left(x^2-4.5x+5\right)\\
&=0.05x^2-0.425x+1.15\\
&=\left(0.05x-0.425\right)x+1.15
\end{aligned}$$

Una aproximación de $f(3)=\frac{1}{3}$ usando el polinomio hallado sería $f(3)\approx p(3)=0.325$.

El error real en la aproximación es: $f(3)-p(3)$

$$= 1/3-0.325$$
$$= 0.00833=8.33\times 10^{-3}$$

Para calcular el término residual o cota para el error involucrado en la aproximación de una función mediante un polinomio Interpolante de Lagrange, se usa el siguiente teorema.

Teorema 1.7

Suponga que $x_0, x_1, ..., x_n$ son números distintos en el intervalo $[a,b]$. Entonces para cada $x \in [a,b]$ existe un número $\xi(x)$ en (a,b) con:

$f(x) = p(x) + \dfrac{f^{(n+1)}(\xi(x))}{(n+1)!}(x-x_0)(x-x_1)...(x-x_n)$. donde *p(x)* es el polinomio interpolante de Lagrange.

Ejemplo:

Calcular la cota del error en la aproximación $f(3)$ del ejemplo anterior.

Solución:

La cota del error esta dada por la expresión:

$$\left|\frac{f^{(n+1)}(\xi(x))}{(n+1)!}(x-x_0)(x-x_1)...(x-x_n)\right|$$

$\left|\dfrac{f'''(\xi(x))}{3!}(x-x_0)(x-x_1)(x-x_2)\right|$ donde $2 < \xi(x) < 4$

Ahora, $f(x) = \dfrac{1}{x}, f'(x) = -\dfrac{1}{x^2}, f''(x) = \dfrac{2}{x^3}, f'''(x) = -\dfrac{6}{x^4}, f'''(\xi(x)) = -\dfrac{6}{(\xi(x))^4}$

Luego $\left|-\dfrac{6}{6(\xi(x))^4}(3-2)(3-2.5)(3-4)\right| = \left|-\dfrac{1}{(\xi(x))^4}(-0.5)\right| = \left|\dfrac{0.5}{(\xi(x))^4}\right| < 0.0625 = 6.25\times10^{-2}$

El error resultante es menor que la cota obtenida, por lo que la aproximación es aceptable.

1.5 VECTORES Y MATRICES

Definición 1.8

El sistema ordenado de n números reales $(x_1, x_2 ..., x_n); x_i \in R, i = 1,2,...n$ se llama vector real n-dimensional x y los números x_i, $i = 1, 2, ...n$ se denominarán componentes del vector, el número n se llama dimensión del vector.

Como ejemplo de vectores señalamos los siguientes:

(i) Los coeficientes de cualquier ecuación lineal con n incógnita forman un vector de dimensión n.

(ii) Toda solución de cualquier sistema de ecuaciones lineales con n incógnitas es un vector de dimensión n.

Se llama suma de los vectores $x = (x_1, x_2, ...x_n)$ y $y = (y_1, y_2, ...y_n)$ al vector $x + y = (x_1 + y_1, x_2 + y_2, ..., x_n + y_n)$ cuyas componentes son iguales a la suma de las componentes, correspondientes de los vectores que se suman y se llama producto del vector x por el número $k \in R$ al vector $kx = (kx_1, kx_2, ..., kx_n)$, cuyas componentes son iguales al producto de las componentes del vector x por el número k.

Definición 1.9

Una matriz es un arreglo rectangular de números dispuestos en filas y columnas, los números en el arreglo son llamados los elementos de la matriz.

Usaremos letras mayúsculas para denotar las matrices A, B, C,…..y letras minúsculas para denotar sus elementos a, b, c,… Por ejemplo,

$$A = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix}$$

El tamaño de una matriz se describe especificando el número de filas y el número de columnas, así por ejemplo, sea la matriz $A=\begin{bmatrix} 1 & -5 & 7 & 2 \\ 3 & 0 & 4 & -1 \end{bmatrix}$. La matriz A tiene 2 filas y 4 columnas, es decir, A es una matriz 2×4.

Definición 1.10 (Norma de un vector)

Sea V un espacio vectorial con producto interno, la norma de $u \in V$ denotada por $\|u\|$ se define como $\|u\| = < u, u >^{1/2}$.

Si u y v $\in V$, la distancia entre u y v se define por $d(u,v) = \|u - v\| = < u - v, u - v >^{1/2}$.

1.6 PROGRAMACÓN LINEAL

En esta sección se presentan los conceptos básicos y las ideas fundamentales de una técnica algebraica conocida como método simplex, utilizada para resolver problemas de programación lineal independientemente del número de variables.

El problema general de la programación lineal con *n* variable se describe de la siguiente manera:

Obtener los valores de $x_1, x_2, ...x_n$ que maximizan o minimizan a:

$$Z = c_1 x_1 + c_2 x_2 + ...c_n x_n \qquad (1)$$

Sujeta a

$$\begin{aligned} &a_{11}x_1 + a_{12}x_2 + ... + a_{1n}x_n (\leq)(\geq) b_1 \\ &a_{21}x_1 + a_{22}x_2 + ... + a_{2n}n_n (\leq)(\geq) b_2 \qquad (2) \\ &\vdots \qquad \vdots \qquad \vdots \\ &a_{m1}x_1 + a_{m2}x_2 + ... + a_{mn}x_n (\leq)(\geq) b_m \end{aligned}$$

y $x_i \geq 0$, para $i = 1,2,...,n$ (3)

A la función lineal Z se le llama función objetivo y a las condiciones (2) y (3) se les llama restricciones.

El problema general de la programación lineal escrito en la forma normal se describe de la siguiente manera:

Obtener los valores de $x_1, x_2, \ldots, x_n$ que maximizan a :

$$Z = c_1x_1 + c_2x_2 + \ldots\ldots + c_nx_n \qquad (4)$$

Sujeta a

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \ldots\ldots + a_{1n}x_n &= b_1 \\ a_{21}x_2 + a_{22}x_2 + \ldots\ldots + a_{2n}x_n &= b_2 \qquad (5) \\ \vdots \qquad \vdots \qquad\qquad \vdots \quad & \\ a_{m1}x_1 + a_{m2}x_2 + \ldots\ldots + a_{mn}x_n &= b_m \end{aligned}$$

y $x_i \geq 0$, para $i = 1,2,\ldots n$

Siempre se puede escribir un problema de programación lineal en forma normal si se siguen los siguientes pasos:

Paso 1: Un problema de minimización se convierte en uno de maximización, definiendo una nueva función objetivo.

$$Z' = -Z$$

Paso 2: Una restricción con $\geq$ se convierte en una restricción con $\leq$, multiplicando la desigualdad por -1.

Paso 3: Una restricción con $\leq$ se convierte en una igualdad, sumando una variable de holgura no negativa al primer miembro de la desigualdad.

Ejemplo:

Escribir en forma normal el siguiente problema de programación lineal.

Minimizar $Z = 2x_1 + 5x_2$

Sujeta a

$$\begin{aligned} 3x_1 - 6x_2 &\leq 2 \\ x_1 + x_2 &\leq 3 \\ x_1 \qquad &\geq 6 \\ x_2 &\leq 5 \\ x_1, x_2 &\geq 0 \end{aligned}$$

Solución:

La forma normal del problema dado será:

Maximizar $Z' = -2x_1 - 5x_2 + 0x_3 + 0x_4 + 0x_5 + 0x_6$

Sujeta a

$$
\begin{array}{lllllll}
3x_1 - 6x_2 + x_3 & & & & = & 2 \\
x_1 + x_2 & + x_4 & & & = & 3 \\
-x_1 & & + x_5 & & = & -6 \\
x_2 & & & + x_6 & = & 5 \\
\multicolumn{7}{l}{x_1, x_2, x_3, x_4, x_5, x_6 \geq 0}
\end{array}
$$

Donde las variables de holgura son: x_3, x_4, x_5 y x_6. La notación matricial es conveniente para compactar el problema general normalizado, el que se escribe como:

Maximizar $Z = C^t x$

Sujeta a $Ax = b$ y $x \geq 0$

Donde $x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$, $C = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_3 \end{bmatrix}$, $b = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}$, $A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ . & . & & . \\ . & . & & . \\ . & . & & . \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{bmatrix}$

La expresión $x \geq 0$ establece la no negatividad de las entradas del vector x.

Planteado de esta forma, el problema consiste en encontrar el vector x no negativo, x en el espacio vectorial R^n que satisfaga la condición $Ax = b$ y que haga que la función $Z = C^t x$ tenga su valor máximo.

Todo vector x no negativo y que satisfaga la restricción $Ax = b$, recibe el nombre de **Solución Factible** del problema de programación lineal.

Al conjunto de todas las soluciones factibles en R^n, se le llama **Conjunto Factible** o **Región Factible** del problema.

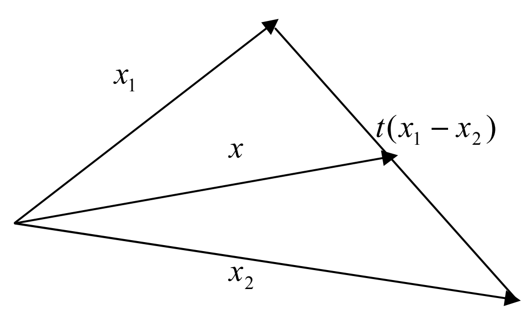
A una solución factible que maximice la función objetivo, se le llama **Solución Óptima.**

A continuación se presentan las definiciones y teoremas requeridos para el análisis de la naturaleza del conjunto factible de un problema de programación lineal.

Definición 1.11

A un conjunto de vectores en R^n se le llama convexo si para dos vectores cualquiera x_1, x_2 del conjunto el vector $x = tx_1 + (1-t)x_2$ también pertenece al conjunto, para cualquier número t del intervalo (0,1).

Geométricamente el vector $x = tx_1 + (1-t)x_2$ se muestra en la figura siguiente:



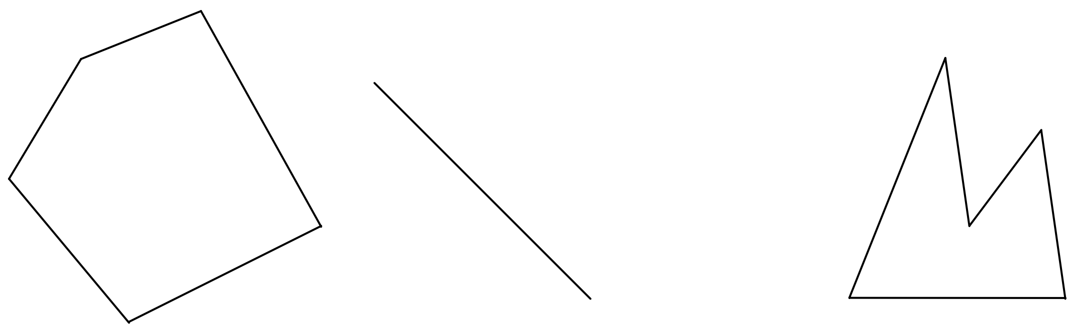
Observe que $x = x_2 + t(x_1 - x_2) = tx_1 + (1-t)x_2$

Figura 1.1

Se puede describir al conjunto convexo como aquel en el que el segmento de recta que une a dos puntos cualesquiera del conjunto pertenece también al conjunto.

Ejemplo:

Las figuras (a) y (b) muestran conjunto convexo en R^2 y la figura (c) es un conjunto en R^2 que no es convexo.



Convexo (a) Convexo (b) No Convexo (c)

Figura1.2

Teorema 1.8

El conjunto factible de un problema normal de programación lineal es convexo.

Ejemplo:

Mostrar si el conjunto factible del siguiente problema de programación lineal es convexo.

Maximizar $Z = 3x_1 + x_2 - 2x_3$

Sujeta a
$$
\begin{aligned}
3x_1 + x_2 + x_3 &= 10 \\
2x_1 - x_2 + 2x_3 &= 10 \\
x_1, x_2, x_3 &\geq 0
\end{aligned}
$$

Solución:

El conjunto factible esta formado por la intersección de los planos

$$\begin{cases} 3x_1 + x_2 + x_3 = 10 \\ 2x_1 - x_2 + 2x_3 = 10 \end{cases}$$

en el primer octante del espacio x_1, x_2, x_3. Para encontrar la intersección, hacemos primero $x_2 = 0$ y resolvemos el sistema resultante.

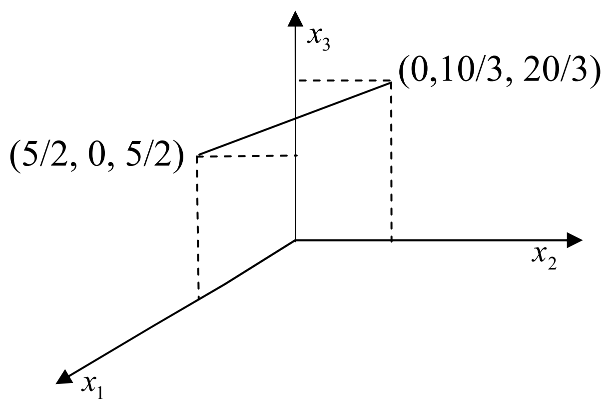
$$\begin{cases} 3x_1 + x_3 = 10 \\ 2x_1 + 2x_3 = 10 \end{cases}$$

Obtenemos $x_1 = \frac{5}{2}$ y $x_3 = \frac{5}{2}$, luego el punto $p\left(\frac{5}{2}, 0, \frac{5}{2}\right)$ es un punto de intersección con el plano x_1, x_3. A continuación hacemos $x_1 = 0$ y resolvemos el sistema resultante.

$$\begin{cases} x_2 + x_3 = 10 \\ -x_2 + 2x_3 = 10 \end{cases}$$

Obtenemos $x_2 = \frac{10}{3}$ y $x_3 = \frac{20}{3}$, luego el punto $Q\left(0, \frac{10}{3}, \frac{20}{3}\right)$ es un punto de intersección con el plano x_2, x_3.

El segmento que une los puntos p y q es la intersección buscada



Obsérvese que el conjunto factible es convexo

Figura 1.3

Definición 1.12

Un vector x de un conjunto convexo en R es un punto extremo si $x \neq \frac{1}{2}(x_1 + x_2)$ para dos vectores cualesquiera x_1 y x_2 del conjunto.

En otras palabras los puntos extremos de un conjunto convexo son aquellos que no están a la mitad de la distancia entre dos puntos cualesquiera del conjunto.

Por ejemplo los puntos extremos del conjunto convexo que se muestra en el ejemplo anterior son los extremos del segmento de recta, o sea los

puntos $\left(\frac{5}{2}, 0, \frac{5}{2}\right)$ y $\left(0, \frac{10}{3}, \frac{20}{3}\right)$.

Para el conjunto convexo que se muestra en la Figura 1.4 los puntos extremos son los vértices de la región triangular.

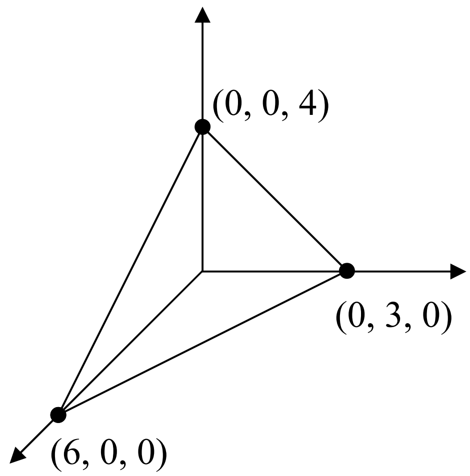


Figura 1.4

En forma semejante se puede definir un conjunto confinado en R^n como sigue:

Definición 1.13

Se dice que un conjunto en R^n es confinado si hay un número positivo r tal que para cada punto del conjunto $x = \{\ x_1, x_2, \Lambda\ \Lambda\ , x_n\ \}$ se sastiface:

$$\|x\| = \sqrt{x_1^2 + x_2^2 + \Lambda\ \Lambda\ x_n^2} \leq r$$

El siguiente teorema establece que si un problema de programación lineal tiene solución óptima, ésta se encuentra en alguno de los puntos extremos del conjunto factible.

Teorema 1.9

Si el conjunto factible de un problema de programación lineal no es vacío y es confinado, la función objetivo alcanza su valor máximo en un punto extremo del conjunto. Si no es confinado la función objetivo puede o no tener valor máximo, si lo tiene, lo alcanza en un punto extremo.

Ejemplo:

Encuentre el valor máximo de la función objetivo $Z = 3x_1 + x_2 - 2x_3$ y la solución óptima del problema correspondiente a esta función objetivo.

Solución:

Como se muestra en la Figura 1.3 el conjunto factible del problema correspondiente a esta función objetivo no es vacío y es confinado entonces por el Teorema 1.9 la función objetivo $Z = 3x_1 + x_2 - 2x_3$ alcanza su valor máximo en una de los dos puntos extremos que tiene el conjunto.

En el punto extremo $(5/2, 0, 5/2)$ se tiene $Z = 5/2$ y en el punto extremo $(0, 10/3, 20/3)$ se tiene $Z = -10$. Por tanto, el valor máximo de la función objetivo es $Z = 5/2$ y la solución óptima del problema es $x_1 = 5/2$, $x_2 = 0$ y $x_3 = 5/2$.

Sea el sistema lineal $Ax = b$ donde A es una matriz de *m x n*, aunque no es esencial por simplicidad se supondrá que $m \leq n$, es decir, que el número de ecuaciones no es mayor que el de la variable.

Definición 1.14

Un vector $x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$ recibe el nombre de solución básica del sistema

lineal $Ax = b$, si $n - m$ de las variables $x_1, x_2, \ldots, x_n$ son 0 y las m restantes corresponden a los vectores columna de A que son linealmente independiente.

Las $n - m$ variables que son ceros reciben el nombre de **no básicas** y los m correspondientes a los vectores columna linealmente independiente son las llamadas **variables básicas** de x.

Un sistema lineal de m ecuaciones con n variables tiene tantas soluciones básicas como conjuntos de m columnas linealmente independientes se pueden formar con los n columnas de la matriz de coeficiente A.

Relacionando la condición de no negatividad $(x \geq 0)$ con el concepto de solución básica, se llega a la siguiente definición.

Definición 1.15

En un programa normal de programación lineal, una solución factible que sea también una solución básica del sistema $Ax = b$ recibe el nombre de **solución factible básica**.

Ahora se puede enunciar el teorema fundamental en la teoría de la programación lineal.

Teorema 1.10

Un vector x es un punto extremo del conjunto factible de un problema de programación lineal si y solo si, es una solución factible básica del problema. Este teorema e
s la base de una técnica algebraica que conduce al conocimiento de los puntos extremos.

Ejemplo:

Obtener la solución óptima del problema de programación lineal.

Maximizar $Z = 2x_1 + 3x_2 - x_3 + 4x_5 + x_6$

$$\begin{aligned} 2x_1 + 0x_2 + x_3 + 4x_4 + 2x_5 &= 20 \\ x_1 + x_2 - x_3 + x_4 + 3x_5 &= 10 \\ x_1, x_2, x_3, x_{4,} x_5 &\geq 0 \end{aligned}$$

Solución:

Primero se obtiene las soluciones básicas del sistema lineal:

$$\begin{cases} 2x_1 + x_3 + 4x_4 + 2x_5 = 20 \\ x_1 + x_2 - x_3 + x_4 + 3x_5 = 10 \end{cases}$$

La matriz de coeficiente del sistema es:

$$A = \begin{bmatrix} 2 & 0 & 1 & 4 & 2 \\ 1 & 1 & -1 & 1 & 3 \end{bmatrix}$$

Como se puede ver dos columnas cualesquiera son linealmente independientes, entonces para encontrar la solución básica, basta con escoger dos de ella, digamos la primera y tercera columna.

$$\begin{bmatrix} 2 \\ 1 \end{bmatrix} , \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

Se igualan a cero las variables no básicas x_2 , x_4, x_{5}, y se obtiene el sistema:

$$\begin{cases} 2x_1 + x_3 = 20 \\ x_1 - x_3 = 10 \end{cases}$$

Donde x_1 y x_3 son las variables básicas.

Fácilmente se llega a $x_1 = 10$, $x_3 = 0$ y la solución básica del problema

original es: $x = \begin{bmatrix} 10 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$

Las diez posibles combinaciones tomando dos columnas de A conducen a las siguientes soluciones básicas en las que han sido subrayadas las variables básicas de cada posibilidad.

$$\begin{bmatrix} \underline{10} \\ \underline{0} \\ 0 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} \underline{10} \\ 0 \\ \underline{0} \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} \underline{10} \\ 0 \\ 0 \\ \underline{0} \\ 0 \end{bmatrix} \begin{bmatrix} 10 \\ 0 \\ 0 \\ 0 \\ \underline{0} \end{bmatrix} \begin{bmatrix} 0 \\ \underline{30} \\ \underline{20} \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ \underline{5} \\ 0 \\ \underline{5} \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ \underline{-20} \\ 0 \\ 0 \\ \underline{10} \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ \underline{-4} \\ \underline{6} \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ \underline{8} \\ 0 \\ \underline{6} \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \\ \underline{4} \\ \underline{2} \end{bmatrix}$$

Y de estas diez las ocho que aparecen en la siguiente lista son las soluciones factibles básicas, ya que satisfacen la condición de no negatividad. Ahora por el Teorema 1.10 los puntos extremos del conjunto factible también están dados por estas soluciones:

$$\begin{bmatrix} \underline{10} \\ \underline{0} \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad \begin{bmatrix} \underline{10} \\ 0 \\ \underline{0} \\ 0 \\ 0 \end{bmatrix} \quad \begin{bmatrix} \underline{10} \\ 0 \\ 0 \\ \underline{0} \\ 0 \end{bmatrix} \quad \begin{bmatrix} 10 \\ 0 \\ 0 \\ 0 \\ \underline{0} \end{bmatrix} \quad \begin{bmatrix} 0 \\ \underline{30} \\ \underline{20} \\ 0 \\ 0 \end{bmatrix} \quad \begin{bmatrix} 0 \\ \underline{5} \\ 0 \\ \underline{5} \\ 0 \end{bmatrix} \quad \begin{bmatrix} 0 \\ 0 \\ \underline{8} \\ 0 \\ \underline{6} \end{bmatrix} \quad \begin{bmatrix} 0 \\ 0 \\ 0 \\ \underline{4} \\ \underline{2} \end{bmatrix}$$

$$z = 20 \quad z = 20 \quad z = 20 \quad z = 0 \quad z = 70 \quad z = 35 \quad z = 2 \quad z = 18$$

(Debajo de cada uno de las ochos soluciones factible básicas se ha escrito el valor correspondiente de la función objetivo.)

Se ve que $z = 70$ es el valor máximo de la función objetivo y que la solución óptima es $x_1 = 0$, $x_2 = 30$, $x_3 = 20$, $x_4 = 0$, $x_5 = 0$.

La técnica del ejemplo anterior en principio es aplicable a cualquier problema de programación lineal, pero el número de soluciones factibles básicas aumenta rápidamente cuando aumenta el número de variable.

Por ejemplo un problema con cuarenta variables y veinte restricciones podría tener más de 130 millones de soluciones factibles básicas y sería impráctico encontrarles todas aún con la computadora más rápida.

A continuación describiremos una alternativa práctica a esta técnica que se conoce con el nombre de Método Simplex, el cual es un procedimiento que parte de una solución factible a otra adyacente, de forma que nunca disminuya el valor de la función objetivo.

El método consta de los siguientes pasos:

Paso 1: Construir la tabla inicial de la matriz aumentada.

Paso 2: Probar la condición óptima, si la tabla la da, se termina el proceso, si no se pasa al paso tres.

Paso 3: Determinar la columna pivote.

Paso 4: Determinar la fila pivote.

Paso 5: Con las operaciones elementales de fila obtener los 0 y el 1 de la columna pivote, de forma que el 1 esté en la fila pivote, y regresar al paso 2.

Los detalles de los pasos 2, 3 y 4 son los siguientes:

(i) Para la condición óptima: Si ninguna de las entradas de las filas objetivo es negativa (no tomar en cuenta la de la extrema derecha) la tabla da la solución óptima.

(ii) Determinación de la columna pivote: La columna pivote debe ser aquella que contenga la entrada más negativa de la fila objetivo (sin tomar en cuenta la situada más a la derecha).

(iii) Haciendo caso omiso de la fila objetivo, se divide cada entrada positiva de la columna pivote entre la última de su fila y se escoge como fila pivote aquella en la que es mayor el resultado de la división.

Ejemplo:

Resolver por el Método Simplex, el problema de programación lineal siguiente.

Maximizar $Z = 2x_1 + 2x_2$

Sujeta a
$$\begin{aligned} 3x_1 + 2x_2 &\le 4 \\ 3x_1 + x_2 &\le 3 \\ 2x_1 \qquad &\le 3 \\ x_1, x_2 &\ge 0 \end{aligned}$$

Solución:

En forma normal este problema se hace así:

Maximizar $z = 2x_1 + 2x_2 + 0x_3 + 0x_4 + 0x_5$

Sujeta a
$$\begin{aligned} 3x_1 + 2x_2 + x_3 \quad\quad\quad &= 4 \\ 3x_1 + x_2 \quad + x_4 \quad &= 3 \\ 2x_1 \quad\quad\quad + x_5 &= 3 \end{aligned}$$

Entonces la tabla inicial para resolver el problema es:

	x_1	x_2	x_3	x_4	x_5	z		
	3	2	1	0	0	0	4	
x_4	3	1	0	1	0	0	3	
	2	0	0	0	1	0	3	
	-2	-1	0	0	0	1	0	
x_3	0	1	1	-1	0	0	1	
	1	$1/3$	0	$1/3$	0	0	1	
	0	$-2/3$	0	$-2/3$	1	0	1	
	0	$-1/3$	0	$2/3$	0	1	2	
	0	1	1	-1	0	0	1	x_2
	1	0	$-1/3$	$2/3$	0	0	$2/3$	x_1
	0	0	$2/3$	$-4/3$	1	0	$5/3$	x_5
	0	0	$1/3$	$1/3$	0	1	$7/3$	z

Para el problema original propuesto se desechan las variables de holgura x_3, x_4 y x_5 y simplemente se escribe $x_1 = 2/3$, $x_2 = 1$, $x_5 = 5/3$ y $z = 7/3$.

La solución óptima es $x_1 = 2/3$, $x_2 = 1$, $x_3 = 0$, $x_4 = 0$, $x_5 = 5/3$ y $z = 7/3$.

Son tres los posibles resultados de un problema de programación lineal:

(i) Las restricciones son inconsistentes y no existen soluciones factibles.

(ii) El conjunto factible no es confinado y la función objetivo toma valores tan grandes como se quiera.

(iii) Existe por lo menos una solución óptima.

En la mayoría de las aplicaciones prácticas, el que se presenta es el caso *(iii)*.

CAPÍTULO 2
MINIMIZACIÓN NUMÉRICA DE FUNCIONES DE UNA VARIABLE

2.1 INTRODUCCIÓN

En este capítulo estudiaremos algunos de los métodos numéricos para localizar el mínimo de una función de una variable.

Revisaremos dos tipos de métodos de minimización: Métodos de búsqueda y Métodos mediante derivadas.

2.2 MINIMIZACIÓN DE FUNCIONES DE UNA VARIABLE

Un método de búsqueda para hallar el mínimo de $f(x)$ consiste en evaluar la función en muchos puntos y buscar un mínimo local entre ellos.

Para reducir el número de evaluaciones de la función se debe tener una estrategia que determine donde tenemos que evaluar $f(x)$. Abordaremos en detalle el método de búsqueda de la sección áurea.

En el caso de los métodos de minimización mediante derivadas se estudia un método de aproximación cuadrática para calcular p, el método de interpolación cuadrática.

2.3 MÉTODO DE LA SECCIÓN ÁUREA

En el método de búsqueda de la sección áurea, la estrategia de selección de los puntos en los que se evaluará la función $f(x)$ depende de la llamada proporción áurea r, que nos ayuda a dividir el intervalo inicial $[a,b]$ de búsqueda por un subintervalo en el que $f(x)$ alcanza su mínimo p.

Para hallar el mínimo de $f(x)$, hay una condición que nos asegura que existe solo un mínimo y que el método converge realmente a dicho mínimo, esto es que $f(x)$ sea unimodal en $[a,b]$.

Si $f(x)$ es una función unimodal en $[a,b]$. Dividimos el intervalo inicial $[a,b]$ en tres subintervalo $[a,c]$, $[c,d]$ y $[d,b]$ utilizando dos puntos interiores $c = a + (1-r)(b-a)$ y $d = a + r(b-a)$ siendo $r = \frac{\sqrt{5}-1}{2}$ la proporción áurea llamada así por los matemáticos de la Grecia clásica.

De manera que $a < c < d < b$ y los valores de $f(c)$ y $f(d)$ son ambos menores que $\max\{f(a), f(b)\}$.

Hay que considerar dos casos (ver Figura 2.1):

(i) Si $f(c) \leq f(d)$ entonces el mínimo debe de estar en el subintervalo $[a,d]$ así que reemplazamos b por d y continuamos la búsqueda.

(ii) Si $f(d) < f(c)$ entonces el mínimo debe de estar en el subintervalo $[c,b]$ así que reemplazamos a por c y continuamos la búsqueda.

En cada paso los valores de la función $f(c)$ y $f(d)$ se comparan y se decide si la búsqueda continúa en $[a,d]$ o bien si continúa en $[c,b]$. La iteración continúa hasta que se obtiene el mínimo p con la precisión deseada.

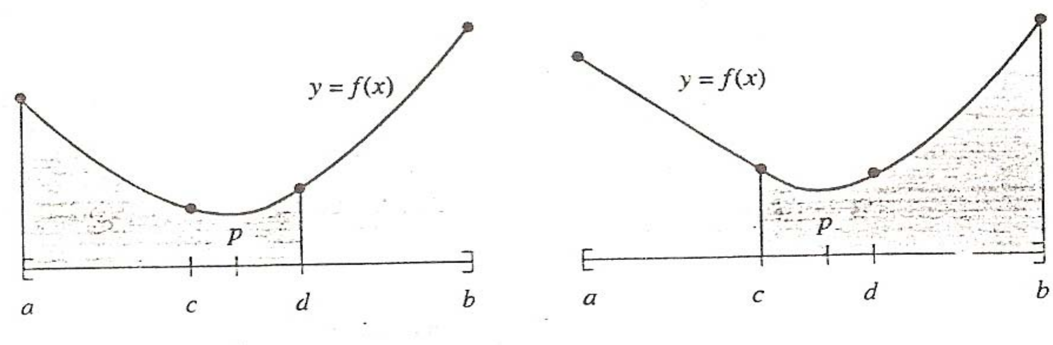


Figura 2.1

Ejemplo:

Encontrar el mínimo de la función $f(x) = 4x^3 - 8x^2 - 11x + 5$ en $[0,2]$.

Solución:

En este ejemplo $a = 0, b = 2$; c y d, se calculan con las fórmulas:

$$c = a + (1 - r)(b - a)$$

, donde $r = \frac{\sqrt{5} - 1}{2}$

$$d = a + r(b - a)$$

De modo que $c = 0.7640$ y $d = 1.2360$.

Considerando 4 decimales, en la tabla siguiente se muestran algunos de los cálculos usando el método de la sección áurea. En cada paso se calculan c, d, $f(c)$, $f(d)$ y se determina el subintervalo de búsqueda.

	a_k	c_k	d_k	b_k	$f(c_k)$	$f(d_k)$
1	0.0000	0.7640	1.2360	2	-6.2897	-13.2646
2	0.7640	1.2361	1.5278	2	-13.2658	-16.2145
3	1.2361	1.5279	1.7081	2	-16.2153	-17.1956
4	1.5279	1.7082	1.8196	2	-16.4002	-17.4047
5	1.7082	1.8196	1.8885	2	-17.4047	-17.3641
6	1.7082	1.7770	1.8156	1.8885	-17.3636	-17.4047
7	1.7770	1.8195	1.8459	1.8885	-17.4047	-17.4051
8	1.8195	1.8458	1.8621	1.8885	-17.4052	-17.3957
9	1.8195	1.8357	1.8458	1.8621	-17.4073	-17.4052
10	1.8195	1.8295	1.8357	1.8458	-17.4072	-17.40723
11	1.8295	1.8357	1.8395	1.8458	-17.4073	-17.4068
12	1.8295	1.8333	1.8356	1.8395	-17.4074	-17.4073
13	1.8295	1.8318	1.8332	1.8356	-17.4073	-17.4074
14	1.8328	1.8332	1.8341	1.8356	-17.4074	-17.4073
15	1.8318	1.8326	1.8332	1.8341	-17.4073	-17.4074
16	1.8326	1.8331	1.8335	1.8341	-17.4074	-17.4074

En la iteración décimo sexta el intervalo se va estrechando hasta ser $[a_{16}, b_{16}] = [1.8326, 1.8341]$, los valores de la función calculados en sus extremos coinciden en cuatro cifras decimales $f(a_{16}) \approx -17.4074 \approx f(b_{16})$ pues:

$$f(a_{16}) = 4(1.8326)^3 - 8(1.8326)^2 - 11(1.8326) + 5 = -17.40739988$$

$$f(b_{16}) = 4(1.8341)^3 - 8(1.8341)^2 - 11(1.8341) + 5 = -17.40739918$$

Como $f(a_{16}) < f(b_{16})$ tomamos el mínimo $p = 1.8326$.

El procedimiento detallado del método de la sección áurea está descrito en el siguiente Algoritmo.

ALGORITMO DE LA SECCIÓN ÁUREA

Para encontrar el mínimo de una función unimodal $f(x)$ en el intervalo $[a,b]$ donde $a < c < d < b$ y los valores $f(c)$ y $f(d)$ son ambos menores que $\max\{f(a), f(b)\}$.

ENTRADA: Extremos a, b; tolerancias para las abscisas ta; tolerancias para las ordenadas to y número máximo de iteraciones m.

SALIDA: Mínimo aproximado p ó mensaje de fracaso.

PASO 1: Tomar $r = \frac{\sqrt{5}-1}{2}$ (proporción áurea)

PASO 2: Calcular $c = a + (1-r)(b-a)$

$$d = a + r(b-a)$$

PASO 3: Tomar $k = 1$

PASO 4: Mientras $k \le m$ seguir paso 5-7

PASO 5: Si $f(c) \le f(d)$ entonces tomar $b = d$.

$f(b) = f(d)$
$d = c$
$f(d) = f(c)$
$c = a + (1-r)*(b-a)$
$fc = f(c)$

Sino tomar $a = c$.

$f(a) = f(c)$
$c = d$
$f(c) = f(d)$
$d = a + r*(b-a)$
$fd = f(d)$

PASO 6: Si $(|b-a| < ta$ ó $|f(b) - f(a)| < to)$ entonces

$p = a$
$f(p) = f(a)$

Salida (p)
Parar

PASO 7: $k = k + 1$

PASO 8: Salida (“El método fracasó después de *m* iteraciones, $m =$”, m)
Parar.

En el Anexo 2, se presenta el programa en lenguaje C correspondiente al algoritmo de la sección áurea para encontrar el mínimo de una función unimodal $f(x)$ en $[a,b]$.

Comparemos ahora, el método de búsqueda de la sección áurea con el método de hallar mínimos locales usando la derivada, estudiada en la sección 1.2 del capítulo 1. Para ello usaremos el ejemplo anterior.

Dado que $f(x) = 4x^3 - 8x^2 - 11x + 5$ se tiene que $f'(x) = 12x^2 - 16x - 11$ y $f''(x) = 24x - 16$ hacemos $f'(x) = 0$ ó sea $f'(x) = 12x^2 - 16x - 11 = 0$ puesto que:

$f'(0) = 12(0)^2 - 16(0) - 11 = -11$

$f'(2) = 12(2)^2 - 16(2) - 11 = 5$

Existe una raíz para la ecuación $f'(x) = 0$ en el intervalo $[0,2]$.

Aplicando el método de la secante con $p_o = 0$ y $p_1 = 2$, la tabla siguiente muestra las iteraciones realizadas.

	p_k	$12p_k^2 - 16p_k - 11$	$\lvert p_k - p_{k-1} \rvert$
0	0.00000000	-11.0000000	
1	2.00000000	5.00000000	
2	1.37500000	-10.3125000	0.62500000
3	1.79591836	-1.03082069	0.42091836
4	1.84266544	0.26234404	0.04674708
5	1.83318232	-0.00422809	0.00948312
6	1.83333273	-0.00001689	0.00015041
7	1.83333333	-0.00000009	0.00000006

La raíz encontrada con seis cifras de precisión es 1.83333333 y $f''(1.83333333) = 27.99999999 > 0$.

El valor mínimo de f se alcanza en $p = 1.83333333$ y es $f(p) = -17.40740741$.

Uno de los problemas del método de búsqueda es que la función suele ser bastante plana cerca del mínimo y esto limita la exactitud que podemos obtener. En este ejemplo el método de la secante nos proporciona una respuesta más exacta $p_7 = 1.83333333$.

Aunque el método de búsqueda de la sección áurea es más lento en este ejemplo, tiene un aspecto deseable y es que puede usarse cuando $f(x)$ no es derivable ó cuando su derivada no puede calcularse fácilmente.

2.4 MÉTODO DE INTERPOLACIÓN CUADRÁTICA

Supongamos que $f(x)$ una función unimodal en $[a,b]$ y que su único mínimo se encuentra en $x = p$. Sea $f(x)$ derivable en (a,b) tomemos un valor inicial p_0 en (a,b).

(i) Si $f'(p_0) < 0$ entonces el mínimo p está en el intervalo $[p_0, b]$.

(ii) Si $f'(p_0) > 0$ entonces p está en el intervalo $[a, p_0]$.

(Ver Figura 2.2)

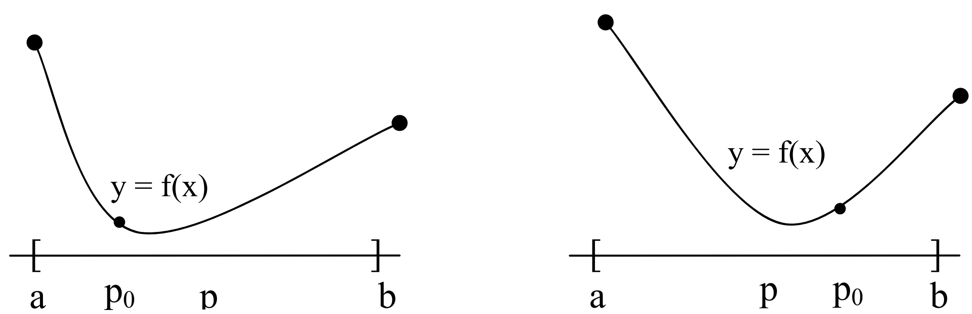


Figura 2.2

Para localizar el mínimo se obtiene primero tres valores de partida

$$p_0, \qquad p_1 = p_0 + h, \qquad p_2 = p_0 + 2h \qquad (1)$$ donde h es un incremento positivo o negativo,

tales que cumplen:

$$f(p_0) > f(p_1) \text{ y } f(p_1) < f(p_2) \qquad (2)$$

Si $f'(p_0) < 0$, entonces $p_0 < p$ y elegimos un incremento h positivo.

Empezamos con $h = 1$ en la fórmula (1), se debe tener p_0, p_1 y p_2 en $[a,b]$, si esto no ocurre se toma $h = \frac{1}{2}$ y así sucesivamente.

Caso 1: Si cumple con (2) hemos terminado.

Caso 2: Si $f(p_0) > f(p_1)$ y $f(p_1) > f(p_2)$ entonces $p_2 < p$ y tenemos que buscar más a la derecha, se dobla el incremento a la mitad y repetimos el proceso.

Caso 3: Si $f(p_0) \leq f(p\ )$ entonces hemos ido demasiado lejos a la derecha de p y h es demasiado grande necesitamos valores más cercano a p_0, se reduce el incremento a la mitad y repetimos el proceso.

Cuando $f'(p_0) > 0$, entonces se toma el incremento h negativo y se repite un proceso similar al de los casos 1 a 3 anterior.

Con los tres puntos p_0, p_1 y p_2 ya encontrados, se usa Interpolación Cuadrática para hallar una aproximación a p que denotamos $p_{\min}$.

La Interpolación Cuadrática nos proporciona un polinomio $Q(x)$ de grado dos que aproxime a la función $f(x)$ en los puntos p_0, p_1 y p_2.

Hallar una aproximación al mínimo p para el polinomio es equivalente a encontrarla para la función $f(x)$.

El Polinomio interpolador de lagrange para los puntos de segundo grado p_0, p_1 y p_2 es:

$$Q(x) = f(p_0)l_0(x) + f(p_1)l_1(x) + f(p_2)l_2(x)$$

$$Q(x) = y_0 \frac{(x-p_1)(x-p_2)}{(p_0-p_1)(p_0-p_2)} + y_1 \frac{(x-p_0)(x-p_2)}{(p_1-p_0)(p_1-p_2)} + y_2 \frac{(x-p_0)(x-p_1)}{(p_2-p_0)(p_2-p_1)}$$

$$Q(x) = y_0 \frac{(x-p_1)(x-p_2)}{(-h)(-2h)} + y_1 \frac{(x-p_0)(x-p_2)}{(h)(-h)} + y_2 \frac{(x-p_0)(x-p_1)}{(2h)(h)}$$

$$Q(x) = y_0 \frac{(x-p_1)(x-p_2)}{2h^2} - y_1 \frac{(x-p_0)(x-p_2)}{h^2} + y_2 \frac{(x-p_0)(x-p_1)}{2h^2}$$

$$Q(x) = y_0 \frac{(x^2 - p_1x - p_2x + p_1p_2)}{2h^2} - y_1 \frac{(x^2 - p_0x - p_2x + p_0p_2)}{h^2} + y_2 \frac{(x^2 - p_0x - p_1x + p_0p_1)}{2h^2}$$

La derivada de $Q(x)$ es:

$$Q'(x) = y_0 \frac{(2x - p_1 - p_2)}{2h^2} - y_1 \frac{(2x - p_0 - p_2)}{2h^2} + y_2 \frac{(2x - p_0 - p_1)}{2h^2}$$

Hacemos $x = p_0 + h_{\min}$

$$Q'(p_0 + h_{\min}) = y_0 \frac{(2(p_0 + h_{\min}) - p_1 - p_2)}{2h^2} - y_1 \frac{(2(p_0 + h_{\min}) - p_0 - p_2)}{h^2} + y_2 \frac{(2(p_0 + h_{\min}) - p_0 - p_1)}{2h^2}$$

$$Q'(p_0 + h_{\min}) = y_0 \frac{(2p_0 + 2h_{\min} - p_1 - p_2)}{2h^2} - y_1 \frac{(2p_0 + 2h_{\min} - p_0 - p_2)}{h^2} + y_2 \frac{(2p_0 + 2h_{\min} - p_0 - p_1)}{2h^2}$$

Ahora resolvemos $Q'(p_0 + h_{\min}) = 0$ o sea:

$$O = \left[y_0 \left(\frac{2p_0 + 2h_{\min} - p_1 - p_2}{2h^2} \right) - y_1 \left(\frac{2p_0 + 2h_{\min} - p_0 - p_2}{h^2} \right) + y_2 \left(\frac{2p_0 + 2h_{\min} - p_0 - p_1}{2h^2} \right) \right] \quad (3)$$

Multipliquemos cada término de (3) por $2h^2$

$$O=\left[y_0\left(\frac{2p_0+2h_{\min}-p_1-p_2}{2h^2}\right)-y_1\left(\frac{2p_0+2h_{\min}-p_0-p_2}{h^2}\right)+y_2\left(\frac{2p_0+2h_{\min}-p_0-p_1}{2h^2}\right)\right]2h^2$$

Agrupamos los términos que contiene el factor $h_{\min}$, y obtenemos:

$$-\left[-2y_0+4y_1-2y_2\right]h_{\min}=\left[y_0\left(2p_0-p_1-p_2\right)-y_1\left(4p_0-2p_0-2p_2\right)+y_2\left(2p_0-p_0-p_1\right)\right]$$

$$\left[-2y_0+4y_1-2y_2\right]h_{\min}=\left[y_0\left(2p_0-p_0-h-p_0-2h\right)-y_1\left(4p_0-2p_0-2p_0-4h\right)+y_2\left(2p_0-p_0-p_0-h\right)\right]$$

$$\left[-2y_0+4y_1-2y_2\right]h_{\min}=\left[y_0\left(-3h\right)-y_1\left(-4h\right)+y_2\left(-h\right)\right]$$

$$-\left[-2y_0+4y_1-2y_2\right]h_{\min}=\left[-3y_0h+4y_1h-y_2h\right]$$

Despejando $h_{\min}$:

$$h_{\min}=\frac{h(4y_1-3y_0-y_2)}{4y_1-2y_0-2y_2}$$

El valor $p_{\min}=p_0+h_{\min}$ es una aproximación a p mejor que p_0. Ahora reemplazamos p_0 por $p_{\min}$ y repetimos los dos procesos descritos antes, obteniendo un nuevo incremento h y un nuevo $h_{\min}$. La iteración continúa hasta que se obtiene el mínimo con la precisión deseada.

Ejemplo:

Hallar el mínimo local de $f(x)=4x^3-8x^2-11x+5$ en $[0,2]$.

Solución:

Se tiene que $f'(x)=12x^2-16x-11$ y tomemos como valor inicial $p_0=0$.

Luego $f'(0)=12(0)^2-16(0)-11=-11<0$

Por tanto el mínimo p esta en el intervalo $[0,2]$

Eligiendo $h = 1$

$p_0 = 0$, $\quad p_1 = 0 + 1 = 1,$ $\quad p_2 = 0 + 2(1) = 2$

Como $f(x) = 4x^3 - 8x^2 - 11x + 5$ entonces:

$$f(p_0) = f(0) = 4(0)^3 - 8(0)^2 - 11(0) + 5 = 5 \qquad f(p_0) = 5$$
$$f(p_1) = f(1) = 4(1)^3 - 8(1)^2 - 11(1) + 5 = -10 \qquad f(p_1) = -10$$
$$f(p_2) = f(2) = 4(2)^3 - 8(2)^2 - 11(2) + 5 = -17 \qquad f(p_2) = -17$$

Obtenemos $f(p_0) > f(p_1)$ y $f(p_1) > f(p_2)$

Tomemos $h = 2$

$p_0 = 0$, $\quad p_1 = 0 + 2 = 2,$ $\quad p_2 = 0 + 2(2) = 4$

como $f(x) = 4x^3 - 8x^2 - 11x + 5$ entonces:

$$f(p_0) = f(0) = 4(0)^3 - 8(0)^2 - 11(0) + 5 = 5 \qquad f(p_0) = 5$$
$$f(p_1) = f(2) = 4(2)^3 - 8(2)^2 - 11(2) + 5 = -17 \qquad f(p_1) = -17$$
$$f(p_2) = f(4) = 4(4)^3 - 8(4)^2 - 11(4) + 5 = 89 \qquad f(p_2) = 89$$

Ahora como se cumple que $f(p_0) > f(p_1)$ y $f(p_1) < f(p_2)$, entonces hemos terminado el proceso.

Puesto que $f(p_0) = y_0$, $\quad f(p_1) = y_1$ y $\quad f(p_2) = y_2$

$y_0 = 5$, $\quad y_1 = -17$, $\quad y_2 = 89$

Así $$h_{\min} = \frac{h(4y_1 - 3y_0 - y_2)}{4y_1 - 2y_0 - 2y_2} = \frac{2(4(-17) - 3(5) - 89)}{(4(-17) - 2(5) - 2(89))} = 1.34375$$

$$p_{\min} = p_0 + h_{\min} = 0 + 1.34375 = 1.34375$$

Ahora hacemos que $p_0 = 1.34375$ y repetimos el proceso de encontrar h y $h_{\min}$. Para $h = 0.5$

$p_0 = 1.34375$, $p_1 = 1.34375 + 0.5 = 1.84375$, $p_2 = 1.34375 + 2(0.5) = 2.34375$
Como $f(x) = 4x^3 - 8x^2 - 11x + 5$ entonces:

$$f(p_0) = f(1.34375) = 4(1.34375)^3 - 8(1.34375)^2 - 11(1.34375) + 5 = -14.5211$$
$$f(p_1) = f(1.84375) = 4(1.84375)^3 - 8(1.84375)^2 - 11(1.84375) + 5 = -17.4039$$
$$f(p_2) = f(2.34375) = 4(2.34375)^3 - 8(2.34375)^2 - 11(2.34375) + 5 = -13.2281$$

Ya que $f(p_0) > f(p_1)$ y $f(p_1) < f(p_2)$; luego los tres puntos a considerar son:

$p_0 = 1.34375$ $\quad p_1 = 1.84375,$ $\quad p_2 = 2.34375$ y tenemos:

$$y_0 = f(p_0) = -14.5211$$
$$y_1 = f(p_1) = -17.4039$$
$$y_2 = f(p_2) = -13.2281$$

Así:

$$h_{\min} = \frac{h(4y_1 - 3y_0 - y_2)}{4y_1 - 2y_0 - 2y_2} = \frac{0.5(4(-17.4039) - 3(-14.5211) - (-13.2281))}{(4(-17.4039) - 2(-14.5211) - 2(-13.2281))} = 0.45420$$

$$p_{\min} = p_0 + h_{\min} = 1.34375 + 0.45420 = 1.79795$$

Ahora hacemos que $p_0 = 1.79795$ y repetimos el proceso de encontrar h y $h_{\min}$. Para $h = 0.0625$.

$p_0 = 1.79795$, $p_1 = 1.79795 + 0.0625 = 1.8604$, $p_2 = 1.79795 + 2(0.0625) = 1.92295$
Como $f(x) = 4x^3 - 8x^2 - 11x + 5$ entonces:

$$f(p_0) = f(1.79795) = 4(1.79795)^3 - 8(1.79795)^2 - 11(1.79795) + 5 = -17.3900$$
$$f(p_1) = f(1.86040) = 4(1.86040)^3 - 8(1.86040)^2 - 11(1.86040) + 5 = -17.3970$$
$$f(p_2) = f(1.92295) = 4(1.92295)^3 - 8(1.92295)^2 - 11(1.92295) + 5 = -17.2920$$

Ya que $f\left(p_0\right) > f\left(p_1\right)$ y $f\left(p_1\right) < f\left(p_2\right)$, luego los tres puntos a considerar son:

$p_0 = 1.79795$, $p_1 = 1.86040$, $p_2 = 1.92295$

$$y_0 = f(p_0) = -17.3900$$
$$y_1 = f(p_1) = -17.3970$$
$$y_2 = f(p_2) = -17.2920$$

Así:

$$h_{\min} = \frac{h(4y_1 - 3y_0 - y_2)}{4y_1 - 2y_0 - 2y_2} = \frac{0.0625(4(-17.3970) - 3(-17.3900) - (-17.2920))}{(4(-17.3970) - 2(-17.3900) - 2(-17.2920))} = 0.035156$$

$$p_{\min} = p_0 + h_{\min} = 1.79795 + 0.035156 = 1.833106\,.$$

ALGORITMO DE INTERPOLACIÓN CUADRÁTICA

Para encontrar un mínimo local de una función unimodal $f(x)$ en el intervalo $[a,b]$ y derivable en (a,b) tomando un valor inicial p_0 en $[a,b]$.

Entrada: Extremos a,b; tolerancias para las ordenadas t_0, t_1, t_2 y número máximo de iteraciones m.

Salida: Mínimo aproximado P o mensaje de fracaso.

Paso 1: Tomar $P_0 = a$

Tomar $Z_0 = f'(P_0)$

Paso 2: Si $f'(P_0) < 0 \quad h = 1$

Sino tomar $h = -1$

Paso 3: Mientras $r \leq m$ seguir pasos 4-11

Paso 4: Tomar $P_1 = P_0 + h,\ \ P_2 = P_0 + 2h$

Paso 5: Tomar $y_0 = f(P_0),\ \ y_1 = f(P_1),\ \ y_2 = f(P_2)$

Paso 6: Calcular h tal que

$y_0 > y_1$ y $y_1 < y_2$

Paso 7: Hacemos $h_{\min} = \dfrac{h(4y_1 - 3y_0 - y_2)}{4y_1 - 2y_0 - 2y_2}$

Paso 8: Tomar $P_{\min} = P_0 + h_{\min}$

$$y_{\min} = f(P_{\min})$$
$$E_0 = |y_0 - y_{\min}|$$
$$E_1 = |y_1 - y_{\min}|$$
$$E_2 = |y_2 - y_{\min}|$$

Paso 9: Si $((E_1 < t_0) \wedge (E_1 < t_1) \wedge (E_2 < t_2))$ entonces

Salida $(P_{\min})$

Parar

Paso 10: Tomar $P_0 = P_{\min}$

Paso 11: $k = k + 1$

Paso 12: ("El método fracaso después de *m* iteraciones, $m =$", m)
Parar

En el Anexo 3, se presenta el programa en lenguaje C correspondiente al algoritmo de Interpolación Cuadrática para encontrar el mínimo de una función $f(x)$ en $[a,b]$.

Comparando los resultados obtenidos, para la función $f(x) = 4x^3 - 8x^2 - 11x + 5$, con los métodos de la sección áurea y de la interpolación cuadrática se encuentra que para el primer método el mínimo es $p = 1.8359214$ con 12 iteraciones y para el segundo método, el mínimo es $p = 1.832422$ con 11 iteraciones. En ambos casos con una precisión de 0.0001.

El segundo resultado es más cercano al mínimo $p = 1.83333333$ proporcionado por el método analítico, que utiliza la derivada y que fue determinada en la sección 1.2.

CAPÍTULO 3
MINIMIZACIÓN NUMÉRICA DE FUNCIONES DE VARIAS VARIABLES.

3.1 INTRODUCCIÓN

En este capítulo, se trata de minimizar una función de varias variables donde el problema no lineal lo igualamos a cero convirtiéndolo en un problema de minimización.

Revisaremos dos tipos de minimización de varias variables, el Método de Nelder-Mead y el Método del Gradiente o del descenso acelerado.

3.2 MINIMIZACIÓN DE FUNCIONES DE VARIAS VARIABLES

Examinando las funciones de una sola variable, ya hemos indicado que el estudio de diferentes fenómenos obliga a utilizar las funciones de dos y más variables independientes.

Para encontrar el mínimo de una función de varias variables utilizaremos un método llamado Nelder-Mead, que es un tipo de búsqueda del simplex.

En dos dimensiones, un simplex es una figura geométrica que consiste de tres puntos (vértices), junto con todos los segmentos de la línea que los conectan; en otras palabras un triángulo.

En tres dimensiones, un simplex es un tetraedro con cuatro puntos y las caras poligonales que los conectan.

En R^n, un simplex consiste de $n+1$ puntos y la conexión de ellos en segmentos del hiperplano. Para la simplicidad, nosotros limitamos nuestra descripción del método del simplex a dos dimensiones.

3.3 MÉTODO DE NELDER-MEAD

El método busca de modo aproximado una solución óptima a un problema con *n* variables cuando la función a minimizar varía suavemente.

Sea $f(x, y)$ la función que queremos minimizar, si partimos de un triángulo inicial cuyo vértice son $v_k = (x_k, y_k)$ $k = 1, 2, 3$.

Entonces evaluamos la función $f(x, y)$ en cada uno de los vértices y obtenemos los $z_k = f(x_k, y_k)$ para cada $k = 1, 2, 3$.

Luego por el ordenamiento de los subíndices $z_1 \leq z_2 \leq z_3$, introducimos la notación $O = (x_1, y_1)$ $B = (x_2, y_2)$ y $P = (x_3, y_3)$ donde O es el vértice óptimo, B es el vértice bueno, el siguiente del óptimo y P es el vértice peor.

El siguiente paso es calcular el punto medio del segmento que une al vértice O con el vértice B mediante las siguientes coordenadas:

$$M = \frac{O + B}{2} = \left(\frac{x_1 + x_2}{2}, \frac{y_1 + y_2}{2} \right)$$

Si la función decrece al momento de moverlos desde P hasta O a lo largo del triángulo inicial y también al moverlo desde P hasta B es de esperar que la función $f(x, y)$ tome valores menores en puntos alejados del peor vértice P situados al otro lado del segmento que une O con B.

Si tomamos el punto R como el punto simétrico de P respecto de M, entonces el nuevo triángulo es OBR y la fórmula vectorial para hallar R es $R = M + (M - P) = 2M - P$ como pueden ver en Figura 3.1

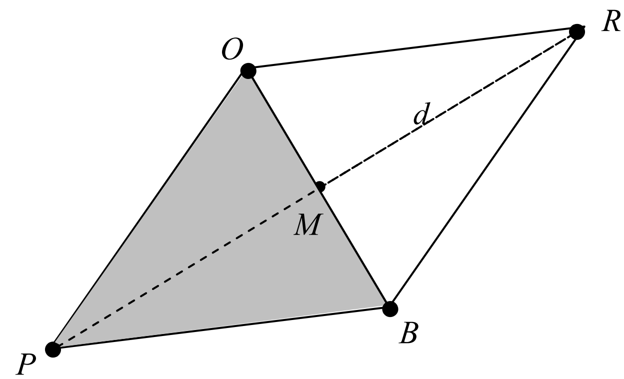


Figura 3.1

En la Figura 3.1, se representa el triángulo OBP, el punto medio M y el punto reflejado R.

Hay que considerar dos casos:

(i) Si $f(R) < f(P)$ entonces nos hemos movido en la dirección correcta hacia el mínimo. Extendiéndonos sobre el segmento que une M y R hasta un punto E, formamos un triángulo extendido OBE, como se indica en la Figura 3.2.

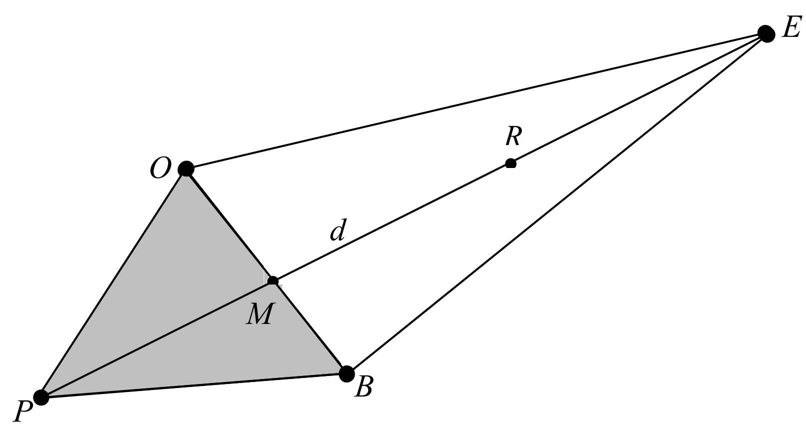


Figura 3.2

Si $f(E) < f(R)$, entonces hemos encontrado un vértice mejor que R y la fórmula vectorial para calcular E es: $E = R + (R - M) = 2R - M$.

(ii) Si $f(R) = f(P)$ ó $f(P) < f(R)$ entonces hay que probar otro punto. Consideremos los puntos medios C_1 y C_2 de los segmentos rectilíneos $\overline{PM}$ y $\overline{MR}$ respectivamente. (Véase Figura 3.3). Se calcula $f(C_1)$ y $f(C_2)$. El punto en que la función tome el valor menor lo llamamos C y el nuevo triángulo es ahora OBC.

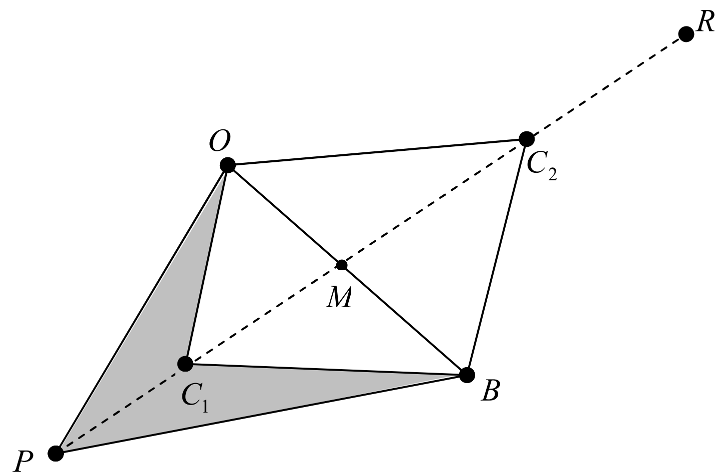


Figura 3.3

Si $f(P) < f(C)$ entonces tenemos que encoger el triángulo en la dirección de O.

El punto B se reemplaza por M y el punto P se reemplaza por S, que es el punto medio del segmento que une O con P. (Véase Figura 3.4)

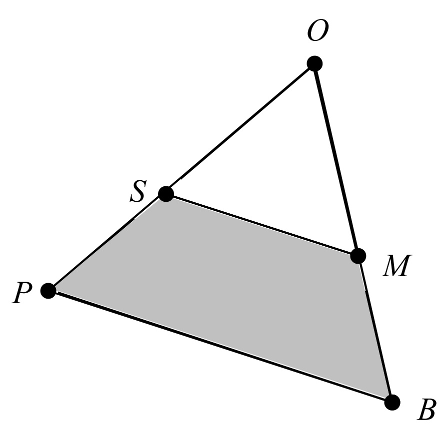


Figura 3.4

Cuando una función es muy plana cerca del mínimo los valores $f(0), f(B)\, y\, f(P)$ son iguales, así que el algoritmo no puede seguir.

Ejemplo:

Vamos a usar el algoritmo de Nelder–Mead para hallar el mínimo de $f(x,y) = x^2 + y^2 + x - 2y - xy + 1$ con los siguientes vértices de partida $v_1(0,0)$, $v_2(2,0)$ y $v_3(2,1)$.

Solución:

La función $f(x,y)$ toma los valores determinados $f(0,0)=1$, $f(2,0)=7$ y $f(2,1)=4$. Comparando los valores, determinamos $O,\ B,\ y\ P$.

$O = (0,0)$ $\qquad\qquad$ $B = (2,1)$ $\qquad\qquad$ $P = (2,0)$

Ahora hay que reemplazar el vértice $P = (2,0)$

$$M = \frac{O+B}{2} = \left(\frac{0+2}{2}, \frac{0+1}{2}\right) = (1, 0.5)$$

$$R = 2M - P = 2(1,0.5) - (2,0) = (0,1)$$

El valor de la función en R es: $f(R) = f(0,1) = 0$ como $f(R) < f(P)$, se extiende el triángulo inicial OBP.

Dado que $f(O) > f(R)$ construímos el vértice E

$$E = 2R - M = 2(0,1) - (1,0.5) = (-1,1.5)$$

El valor $f(E) = f(-1,1.5) = 1.75$ y como $f(E) > f(O)$ entonces se reemplaza P por R, los vértices del nuevo triángulo son:

$V_1 = (0,1)$ $\qquad\qquad$ $V_2 = (0,0)$ $\qquad\qquad$ y $\qquad\qquad$ $V_3 = (2,1)$, o sea los nuevos $O, B, P.$

Repitiendo el proceso con estos vértices, se tiene que:

$$M = \frac{O+B}{2} = \left(\frac{0+0}{2}, \frac{1+0}{2}\right) = (0{,}0.5)$$

$$R = 2M - P = 2(0, 0.5) - (2{,}1) = (-2, 0)$$

$$f(R) = f(-2{,}0) = 3$$

Como $f(R) < f(P)$ se extiende nuevamente el triángulo y dado que $f(O) < f(R)$, reemplazamos P por R y los vértices nuevos son $O = (0{,}1)$, $B = (0, 0)$ y $P = (-2, 0)$.

El proceso continúa y genera una sucesión de triángulos que converge al punto solución $(0{,}1)$.

En la tabla siguiente se muestran los valores de la función en los vértices del triángulo en algunos pasos de la iteración.

k	Óptimo		Punto bueno		Punto peor	
	p_i	$f(p_i)$	p_i	$f(p_i)$	p_i	$f(p_i)$
1	(0,0)	1	(2,1)	4	(2,0)	7
2	(0,1)	0	(0,0)	1	(2,1)	4
3	(0,1)	0	(0,0)	1	(-2,0)	3
4	(0,1)	0	(-1,0.25)	0.81	(0,0)	1
5	(0,1)	0	(-0.25,0.31)	0.36	(-1,0.25)	0.81
6	(0,1)	0	(-0.56,0.45)	0.30	(-0.25,0.31)	0.36
7	(0,1)	0	(-0.31,1.13)	0.15	(-056,0.45)	0.30
8	(0,1)	0	(-0.20,0.75)	0.05	(-0.31,1.13)	0.15
9	(0,1)	0	(-0.20,1)	0.04	(-0.20,0.75)	0.05
10	(0,1)	0	(-0.15,0.87)	0.01	(-0.20,1)	0.04

En la Figura 3.5 se muestra la sucesión de triángulos $\{t_k\}$que converge al punto $(0,1)$ en el método de Nelder –Mead.

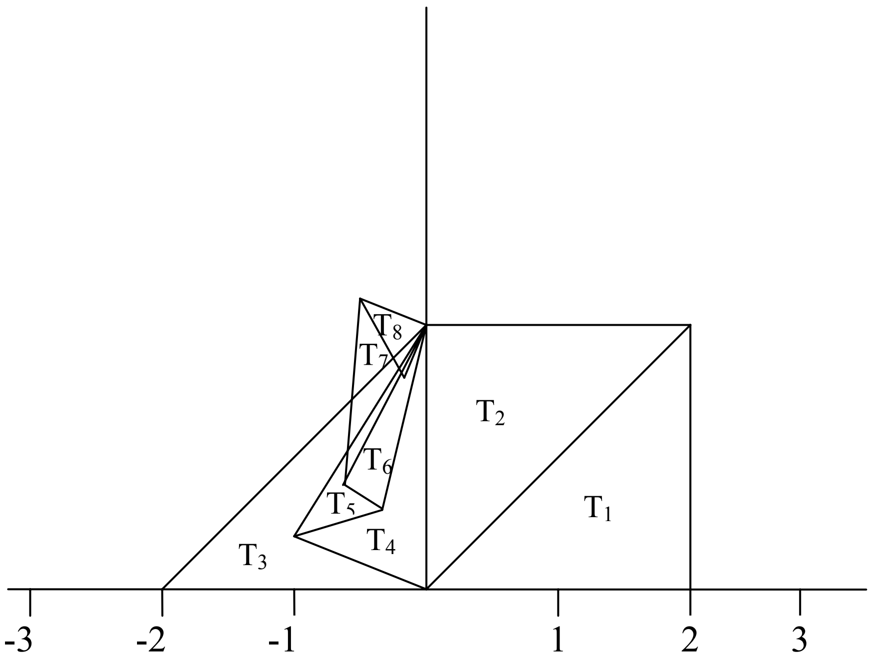


Figura 3.5

ALGORITMO DE NELDER-MEAD

Para encontrar un mínimo local de $f(x_1, x_2, x_3 ...., x_n)$ siendo f una función continua de *n* variable reales y $v_k = \left(v_{k,1}, v_{k,2}, ..., v_{k,n}\right)$ para $k = 0,1,....,n$ los $n+1$ vectores que forman el simplex inicial.

Entrada: Vértices iniciales; $v_1(x_1, y_1), v_2(x_2, y_2)$ y $v_3(x_3, y_3)$, máximo números de iteraciones *m* y la tolerancia tol.

Salida: El vértice óptimo, minimo local $f(0)$ y el mensaje de fracaso.

Paso 1: Tomar $k = 1$

Paso 2: Mientras $k \leq m$, seguir pasos 3-17

Paso 3: Ordenar los vértices iniciales, comparando los valores $f(x_1, y_1), f(x_2, y_2)$ y $f(x_3, y_3)$. Los vértices ordenados son O, B, P.

Paso 4: Tomar $M = \dfrac{O+B}{2}$ y $R = 2M - P$

Paso 5: Si $f(R) < f(P)$

Entonces extender (siguiendo pasos 6-8)

Pasos 6: Si $f(O) < f(R)$ entonces
Se reemplaza P por R

Sino (seguir pasos 7-8)

Paso 7: Calcular $E = 2R - M$ y $f(E)$

Paso 8: Si $f(E) < f(O)$ entonces
Se reemplaza P por E.

Si no
Se reemplaza P por R

Sino encoger (siguiendo pasos 9-13)

Paso 9: Si $f(R) < f(P)$ entonces
Se reemplaza P por R.

Sino (seguir pasos 10-11)

Paso 10: Calcular $C = \frac{P+M}{2}$ y $f(C)$

Paso 11: Si $f(C) < f(P)$ entonces
Se reemplaza P por C

Sino (seguir pasos 12-13)

Paso 12: Calcular $S = \frac{O+P}{2}$ y $f(S)$

Paso 13: reemplazar P por S y
B por M.

Paso 14: Calcular $f(O), f(B), f(P)$

Paso 15: Comparando los valores anteriores yh = mayor, yl = menor.

Paso 16: Si $((yh) > (yl + tol))$ entonces
Salida $(0, f(O))$
Parar

Paso 17: Sea $k = k + 1$

Paso 18: Salida ("El método fracaso después de m iteraciones $m =$", m).
Parar

En el Anexo 4, se presenta el programa en lenguaje C correspondiente al algoritmo de Nelder Mead para encontrar el mínimo de una función $f(x_1, x_2)$ y tres vectores iniciales.

3.4 MÉTODO DEL GRADIENTE

Para aproximar un mínimo local de una función diferenciable $f(x)$ de n variables reales $x = (x_1, x_2, ..., x_n)$ usando el método del gradiente, partimos de un punto inicial p_0 y buscamos en este punto una mejora en la semirrecta que parte de p_0 en la dirección señalada por el vector $s = -G$, siendo $G = grad\, f(p_0)$.

Tomando $p_1 = p_0 + s$. Calculamos $G = grad\, f(p_1)$ y usamos una nueva dirección de búsqueda señalada por el vector $s_1 = -G$; y así obtenemos p_2.

El proceso iterativo produce una sucesión $\{p_k\}$ de puntos que tiene la propiedad $f(p_0) > f(p_1) > \Lambda > f(p_k)\Lambda$

Si $\lim_{k \to \infty} p_k = p$ entonces $f(p)$ es un mínimo local de $f(x)$.

Ejemplo:

Encontrar el mínimo local de la función $f(x_1, x_2) = (x_1^2 + x_2^2 - 1)^2 + (x_1^2 - x_2)^2$ con punto inicial $p_0 = (0.5, 0.5)$ usando el método del gradiente con una tolerancia de 0.00001.

Solución:

Como $gradf\,(x_1, x_2) = f_{x_1}(x_1, x_2)\, i + f_{x_2}(x_1, x_2)\, j$ calculemos f_{x_1} y f_{x_2}.

$$f_{x_1}(x_1, x_2) = 2(x_1^2 + x_2^2 - 1)(2x_1) + 2(x_1^2 - x_2)(2x_1)$$

$$f_{x_2}(x_1, x_2) = 2(x_1^2 + x_2^2 - 1) + 2(x_1^2 - x_2)(-1)$$

Dado que $p_0 = (0.5, 0.5)$ se tiene que:

$$f_{x_1}(0.5,0.5) = 2((0.5)^2 + (0.5)^2 - 1)(2(0.5)) + 2((0.5)^2 - (0.5))(2(0.5)) = -1.5$$

$$f_{x_2}(0.5,0.5) = 2((0.5)^2 + (0.5)^2 - 1)(2(0.5)) + 2((0.5)^2 - (0.5))(-1) = -0.5$$

La dirección de búsqueda es:

$$s = -grad\, f(x_1, x_2) = -(-1.5, -0.5) = (1.5, 0.5)$$

Luego $p = p_0 + s = (0.5, 0.5) + (1.5, 0.5) = (2,1)$

El valor de $f(p_0) = f(0.5, 0.5) = 0.3125$

El valor de $f(p) = f(2,1) = 25$

$$z = f(p) - f(p_0) = 25 - 0.3125 = 24.6875$$

Ahora dividiremos el gradiente s por 2, hasta que $z < 0$ para obtener el nuevo punto p.

$$s = \frac{s}{2} = (0.75, 0.25)$$

$$\|s\| = \sqrt{(0.75)^2 + (0.25)^2} = 0.790569 \quad > 0.00001$$

$$p = p_0 + \frac{s}{2} = (0.5, 0.5) + (0.75, 0.25) = (1.25, 0.75)$$

El valor de *f (p)= f (1.25, 0.75)=1.9258*

$$z = f(p) - f(p_0) = 1.9258 - 0.3125 = 1.6133 > 0$$

$$s = \frac{s}{2} = \left(\frac{0.75}{2}, \frac{0.25}{2}\right) = (0.375, 0.125)$$

$$p = p_0 + s = (0.5, 0.5) + (0.375, 0.125) = (0.875, 0.625)$$

El valor de $f(p) = f(0.875, 0.625) = 0.044189$

$$z = f(p) - f(p_0) = 0.044189 - 0.3125 = -0.268311 < 0$$

Ahora $p_0 = (0.875, 0.625)$ se tiene que:

$$f_{x_1}(0.875,0.625)=2\,((0.875)^2+(0.625)^2-1)(2(0.875))+2((0.875)^2-(0.625))(2(0.875))=1.039062$$

$$f_{x_2}(0.875,0.625)=2\,((0.875)^2+(0.625)^2-1)(2(0.625))+2((0.875)^2-(0.625))(-1)=0.109375$$

La dirección de búsqueda

$$s=-grad\ f(x_1,x_2)=-(1.039062,0.109375)=(-1.039062,-0.109375)$$

Luego $p=p_0+s=(0.875,0.625)+(-1.039062,-0.109375)=(-0.164062,-0.515625)$

El valor de $f(p_0)=f(0.875,0.625)=0.044189$

El valor de $f(p)=f(-0.164062,-0.515625)=0.79018$

$$z=f(p)-f(p_0)=0.79018-0.044189=0.74599>0$$

Ahora dividiremos el gradiente s por 2, hasta que $z<0$ para obtener el nuevo punto p.

$$s=\frac{s}{2}=\left(-0.51955,-0.05469\right)$$

$$\|s\|=\sqrt{(-0.51955)^2+(-0.05469)^2}=0.522422>0.00001$$

$$p=p_0+\frac{s}{2}=\left(0.875,0.625\right)+(-0.51955,-0.05469)=(0.3554,0.5703)$$

$$z=f(p)-f(p_0)=0.49792-0.04419=0.45373>0$$

$$s=\frac{s}{2}=\left(-\frac{0.51955}{2},-\frac{0.05469}{2}\right)=\left(-0.259775,-0.27345\right)$$

$$p=p_0+s=\left(0.875,0.625\right)+\left(-0.259775,-0.27345\right)=\left(0.6152,0.5977\right)$$

El valor de $f(p)=f\left(0.615225,0.597655\right)=0.11791$

$z = f(p) - f(p_0) = 0.11791 - 0.04419 = 0.07371 > 0$

$s = \frac{s}{2} = (-0.1299, -0.0137)$

$p = p_0 + \frac{s}{2} = (0.875, 0.625) + (-0.1299, -0.0137) = (0.74512, 0.61133)$

El valor de $f(p) = f(0.74512, 0.61133) = 0.0082014$
$z = f(p) - f(p_0) = 0.0082014 - 0.04419 = -0.035989 < 0$

El proceso iterativo genera una sucesión de puntos que los mostraremos en la siguiente tabla, hasta encontrar el mínimo p = 1.832422.

k	p_i	$f(p_i)$
1	(0.5, 0.5)	
2	(0.875, 0.625)	0.26831
3	(0.74512, 0.61133)	0.035987

ALGORITMO DEL GRADIENTE

Para aproximar un mínimo local de una función diferenciable $f(x)$ de n variables reales $x = (x_1, x_2, ..., x_n)$ usando el método del gradiente a partir de un punto inicial p_0.

Entrada: Punto inicial p_0, máximo número de iteraciones m, tolerancia
df para $f(p)$.

Salida: Mínimo aproximado p y $f(p)$ ó mensaje de fracaso.

Paso 1: Tomar $k = 1$.

Paso 2: Mientras $k \le m$ seguir los pasos 3-7.

Paso 3: Evaluar el gradiente de f en p_0.

Paso 4: Calcular el negativo del gradiente, que da la dirección de búsqueda.

Paso 5: Tomar $p = p_0 + s$ donde s es una distancia especificada en la dirección del gradiente negativo.

Paso 6: Calcular $f(p_0)$ y $f(p)$.

Paso 7: Si $|f(p) - f(p_0)| < tol$ entonces
La solución es p
Parar.
Sino
El tamaño de la distancia es reducida a la mitad
hasta que la misma es menor que una tolerancia dada.

Paso 8: Tomar $p_0 = p$.

Paso 9: Tomar $k = k + 1$;

Paso 10: (“El método fracaso después de m iteraciones $m =$”, m).
Parar.

En el Anexo 5, se presenta el programa en lenguaje C correspondiente al algoritmo del Gradiente para encontrar el mínimo de una función $f(x_1, x_2)$ dado un punto inicial.

Comparando los resultados obtenidos para la función $f(x, y) = x^2 - 4x + y^2 - y - xy$ con los métodos de Nelder-Mead y del gradiente se encuentra que para el primer método el punto mínimo es $p = (3, 1.8)$ con 9 iteraciones y con vértices $v_1(0,0), v_2(1.2,0)\, y\, v_3(0,0.8)$.

Para el segundo método el punto mínimo es $p = (3.011563, 1.98438)$ con 4 iteraciones y con punto inicial $(0.5, 0.5)$, en ambos casos con una precisión de *0.1*.

Con el método analítico de la sección 1.3 el punto mínimo de la función es $p = (3, 2)$ para este caso el gradiente da un valor más cercano al punto antes mencionado.

Una aplicación del método del Gradiente que puede resultar muy útil se da en la resolución de sistemas no lineales de ecuaciones, específicamente, cuando el problema de resolver un sistema no lineal de ecuaciones se convierte en un problema de Minimización de una función de varias variables. A continuación se detalla el problema:

Sea el sistema general no lineal de ecuaciones

$$f_1(x_1, x_2, x_3, \Lambda\; x_n) = 0$$

$$f_2(x_1, x_2, x_3, \Lambda\; x_n) = 0$$

…………..

$$f_n(x_1, x_2, x_3, \Lambda\; x_n) = 0$$

que en forma abreviada se puede escribir como $F(x) = 0$.

El problema de resolver el sistema no lineal $F(x) = 0$ se transforma en un problema de minimización definiendo $h(x) = (f_1(x))^2 + (f_2(x))^2 + \Lambda\; (f_n(x))^2$ y se determina el mínimo de $h(x)$ mediante el método del gradiente. El mínimo encontrado es la solución del sistema $F(x) = 0$.

Ejemplo:

Resolver el sistema no lineal $f_1(x_1, x_2, x_3)=x_1^2+x_2^2+x_3^2+10x_1-4=0$

$$f_2(x_1, x_2, x_3)=x_1^2-x_2^2+x_3^2+10x_2-5=0$$
$$f_3(x_1, x_2, x_3)=x_1^2+x_2^2-x_3^2+10x_3-6=0$$

encontrando el mínimo de la función $(f_1)^2 + (f_2)^2+(f_3)^2$

Solución:

La función a minimizar es:

$$h(x_1, x_2, x_3)=(x_1^2+x_2^2+x_3^2+10x_1-4)^2 + (x_1^2-x_2^2+x_3^2+10x_2-5)^2 + (x_1^2+x_2^2x_3^2+10x_3-6)^2$$

En el Anexo 6, se presenta el programa en lenguaje C correspondiente al algoritmo del Gradiente para resolver sistemas no lineales de ecuaciones mediante la minimización de una función de tres variables dado un punto inicial.

Al correr este programa el punto mínimo es (0.339567, 0.444938, 0.621555) luego la solución del sistema es $x_1 = 0.339567$, $x_2 = 0.444938$ y $x_3 = 0.621555$.

La aplicación presentada anteriormente es solo una muestra de la utilidad de los métodos de optimización numérica en la solución de problemas que requerirían un mayor esfuerzo y tiempo si fueran resueltos por un método analítico, cuando esto fuera posible. Aunque muchas veces nos podemos encontrar con problemas donde la única vía de solución es la numérica porque la solución analítica no es factible.

En el presente trabajo abordamos cuatro métodos de minimización, dos para el caso de funciones de una variable y dos para el caso de varias variables, aunque en este último caso limitado a dos y tres variables.

Existen otros métodos que tratan el problema de minimización usando diversos requerimientos. Como nuestro objetivo era dar a conocer algunos métodos de minimización, el tema no se agota con este trabajo, por el contrario se puede ampliar el estudio de otros métodos usando como referencia este trabajo.

ANEXOS

ANEXO 1
ALGORITMO DE LA SECANTE

Para encontrar una solución para f(x)=o dada las aproximaciones iniciales p_0 y p_1:

ENTRADA: Aproximaciones iniciales p_0, p_1; tolerancia tol; número máximo de iteraciones N_0.

SALIDA: Solución aproximada p o mensaje de fracaso.

Paso 1: Tomar i=2;
$q_0=f(p_0)$;
$q_1=f(p_1)$.

Paso 2: Mientras $i \leq N_0$ haga paso 3-6

Paso 3: Tomar $p = p_1- q_1 (p_1-p_0)/ (q_1-q_0)$. (calcule p_i)

Paso 4: Si $/p-p_1/<tol$ entonces

SALIDA (p); (procedimiento terminado satisfactoriamente.)
PARAR.

Paso 5: Tomar i=i+1.

Paso 6: Tomar $p_0 = p_1$; (redefina p_0, q_0, p_1, q_1)
$q_0=q_1$;
$p_1=p$;
$q_1=f(p)$.

Paso 7: SALIDA (El método falló después de N_0 =', N_0);
(Procedimiento terminado sin éxito.)

PARAR

PROGRAMA DE LA SECANTE

```
/*Programa para encontrar una solución de la ecuación  f(x)=0  dada dos*/
/* aproximaciones iniciales p0 y p1 mediante el método de la secante */

#include <stdio.h>
#include <math.h>
main( )
{
int i,n;
float p,p0,p1,t,tol,q0,q1;
float f(float x);
system("cls");
printf("\n Introduzca la aproximación inicial p0:  ");
scanf("%f",&p0);
printf("\n Introduzca la aproximación inicial p1:  ");
scanf("%f",&p1);
printf("\n Introduzca la tolerancia:  ");
scanf("%f",&tol);
printf("\n Introduzca el número máximo de  iteraciones ");
scanf("%d",&n);
i=2;
q0=f(p0);
q1=f(p1);
printf("\n i     pi        /pi-pi-1/ ");
while(i<=n)
  {
        p=p1 - q1*(p1-p0)/(q1-q0);
        t=fabs(p-p1);
        printf("\n %d   %f    %f ",i,p,t);
        if (t<tol)
                {
                 printf("\n ");
                 printf("\n La solución aproximada es p=%f ",p);
                 exit(0);
                }

  i=i+1;
  p0=p1;
```

```
  p1=p;
  q0=q1;
  q1=f(p);
  }

printf("\n El método fracaso después de n iteraciones n = %d ",n);

}



float f(float x)

 {

 float y;
 y = cos(x)-x;
 return(y);

 }
```

Salida

Introduzca la aproximación inicial p0: 0

Introduzca la aproximación inicial p1: 1.570796

Introduzca la tolerancia: 0.01

Introduzca el número máximo de iteraciones 30

i	pi	/pi-pi-1/
2	0.611015	0.959781
3	0.723270	0.112254
4	0.739567	0.016298
5	0.739083	0.000484

La solución aproximada es p=0.739083
Elapsed time = 00:00:15.65. Program returned (0). Press any key.

ANEXO 2
PROGRAMA DE LA SECCIÓN ÁUREA

```
*Programa que utiliza el método de sección áurea para minimizar*/
/*una función unimodal de una variable f(x) en [a, b] */

#include<stdio.h>
#include<math.h>
main( )
{
 int k=0,m;
 float a,b,c,d,r,fa,fb,fc,fd,p,fp,ta,to,dp,dy;
 float f(float x);
 system("cls");
 r=(sqrt(5)-1)/2;
 printf("\n Introducir los extremos del intervalo:");
 printf("\n a=");
 scanf("%f",&a);
 printf("\n b=");
 scanf("%f",&b);
 printf("\n Introducir las tolerancias para las abcisas y las ordenadas:");
 printf("\n ta=");
 scanf("%f",&ta);
 printf("\n to=");
 scanf("%f",&to);
 printf("\n Introducir el número máximo de iteraciones:");
 printf("\n m=");
 scanf("%d",&m);
 fa=f(a);
 fb=f(b);
 c=a+(1-r)*(b-a);
 d=a+r*(b-a);
 fc=f(c);
 fd=f(d);
 printf("\nk      a        c        d        b         fc        fd");
 printf("\n%d  %1.7f  %1.7f  %1.7f  %1.7f  %1.7f  %1.7f", k, a, c, d, b, fc,
fd);
```

```
k=1;
while(k<=m)

{

   if(fc<fd)

        {
          b=d;
          fb=fd;
          d=c;
          fd=fc;
          c=a+(1-r)*(b-a);
          fc=f(c);
          dp=fabs(b-a);
          dy=fabs(fb-fa);

        }

   else

        {
          a=c;
          fa=fc;
          c=d;
          fc=fd;
        d=a+r*(b-a);
        fd=f(d);
        dp=fabs(b-a);
        dy=fabs(fb-fa);

        }

   printf("\n%d  %1.7f  %1.7f  %1.7f  %1.7f  %1.7f  %1.7f", k, a, c, d, b,
   fc, fd);

   if((dy<to)||(dp<ta))
```

```
          {

            p=a;
            fp=fa;
            if(fb<fa)

              {
               p=b;
               fp=fb;

              }

      printf("\n");
      printf("\n El valor mínimo  se alcanza en p =%1.7f y es fp =%1.7f ",p,
       fp);
      exit(0);

          }
      k=k+1;

  }

 printf("\n El método fracasó después de m = %d iteraciones ", m);

}

float f(float x)

{
  float y;

  y= 4*pow(x,3)- 8*pow(x,2)- 11*x + 5;
  return(y);
}
```

Salida

Introducir los extremos del intervalo:
a=0
b=2
Introducir las tolerancias para las abcisas y las ordenadas:
ta=0.0001
to=0.0001
Introducir el número máximo de iteraciones:
m=30

k	a	c	d	b	fc	fd
0	0.0000000	0.7639320	1.2360680	2.0000000	-6.2886901	-13.2654848
1	0.7639320	1.2360680	1.5278641	2.0000000	-13.2654848	-16.2150631
2	1.2360680	1.5278641	1.7082039	2.0000000	-16.2150631	-17.1960411
3	1.5278641	1.7082039	1.8196602	2.0000000	-17.1960411	-17.4048004
4	1.7082039	1.8196602	1.8885438	2.0000000	-17.4048004	-17.3640594
5	1.7082039	1.7770877	1.8196602	1.8885438	-17.3638287	-17.4048004
6	1.7770877	1.8196602	1.8459713	1.8885438	-17.4048004	-17.4051628
7	1.8196602	1.8459713	1.8622327	1.8885438	-17.4051628	-17.3956184
8	1.8196602	1.8359214	1.8459713	1.8622327	-17.4073143	-17.4051628
9	1.8196602	1.8297101	1.8359214	1.8459713	-17.4072247	-17.4073143
10	1.8297101	1.8359214	1.8397601	1.8459713	-17.4073143	-17.4068279
11	1.8297101	1.8335489	1.8359214	1.8397601	-17.4074059	-17.4073143
12	1.8297101	1.8320826	1.8335489	1.8359214	-17.4073849	-17.4074059

El valor mínimo se alcanza en p=1.8359214 y fp=-17.4073143
Elapsed time = 00:00:12.85. Program returned (0). Press any key.

ANEXO 3
PROGRAMA DE INTERPOLACIÓN CUADRÁTICA

Programa que utiliza el método de interpolación cuadrática para/
/*minimizar una función unimodal de una variable, *f(x)*, en [a, b] */

```
#include <stdio.h>
#include <math.h>
main( )
{
int k=1,m,resp;
float a,b,ta,p,h,p0,p1,p2,z0,t0,t1,t2,tol,y0,y1,y2,e0,e1,e2,hmin,ymin,pmin;
float f(float x);
float df(float x);
system("cls");
printf("\n Introduzca los extremos del intervalo: ");
printf("\n a=");
scanf("%f",&a);
printf("\n b=");
scanf ("%f",&b);
printf("\n Introduzca las tolerancias para las ordenadas: ");
printf("\n ta= ");
scanf("%f",&ta);
printf("\n t0= ");
scanf("%f", &t0);
printf("\n t1= ");
scanf("%f",&t1);
printf("\n t2= ");
scanf("%f",&t2);
printf("\n Introduzca el número máximo de  iteraciones ");
printf("\n m= ");
scanf("%d",&m);
p0=a;
printf("\n k   h     p0    hmin    pmin    e0     e1      e2 ");
while(k<=m)
  {
   z0=df(p0);
   if (z0<0)
          h=1;
```

```
 if (z0>0)
           h=-1;

repetir:p1=p0+h;
  p2 =p0 + 2*h;

  y0= f(p0);
  y1= f(p1);
  y2= f(p2);
  if((y0>y1)&&(y1<y2))
                        goto calcula;

  if((y0>y1)&&(y1>y2))
              {
               h = 2*h;
               goto repetir;
              }
  if(y0<=y1)
              {
               h =h/2;
               goto repetir;
              }

 calcula: hmin=(h*(4*y1-3*y0-y1))/(4*y1-2*y0-2*y2);
  pmin=p0+hmin;
  ymin=f(pmin);
  e0 =fabs(y0-ymin);
  e1 =fabs(y1-ymin);
  e2 =fabs(y2- ymin);
  printf("\n %d  %g  %f   %f %f  %f %f %f",k, h, p0, hmin, pmin, e0, e1,
  e2);
  if (((e1<t0) && (e1<t1)) && (e2<t2))

                     {
                      printf("\n ");

                      printf("\n La solución aproximada es pmin =%f ", pmin);

                      exit(0);
```

```
                    }
    p0=pmin;
    k=k+1;
   }

printf("\n El método fracaso después de m iteraciones m =%d ",m);

}

float f(float x)
  {
  float y;

  y=4*pow(x,3)-8*pow(x,2)-11*x+5;
  return(y);

  }

  float df(float x)
  {

  float y;
  y=12*pow(x,2)-16*x-11;
  return(y);
  }
```

Salida

Introduzca los extremos del intervalo:
a=0
b=2

Introduzca las tolerancias para las ordenadas:
ta = 0.001

t0= 0.001

t1= 0.001

t2= 0.001

Introduzca el número máximo de iteraciones m= 30

k	h	p0	hmin	pmin	e0	e1	e2
1	2	0.000000	0.515625	0.515625	7.250473	14.749527	91.250473
2	1	0.515625	1.021257	1.536882	14.030777	0.158707	6.661636
3	0.5	1.536882	0.046742	1.583624	0.315474	0.196903	7.512011
4	0.25	1.583624	0.173674	1.757298	0.731504	0.079178	0.860575
5	0.125	1.757298	0.018568	1.775866	0.033705	0.011438	0.399279
6	0.0625	1.775866	0.038508	1.814374	0.040470	0.004650	0.060078
7	0.03125	1.814374	0.004889	1.819263	0.002245	0.000637	0.024113
8	0.015625	1.819263	0.009335	1.828598	0.002445	0.000280	0.003838
9	0.0078125	1.828598	0.001240	1.829838	0.000143	0.000038	0.001493
10	0.00390625	1.829838	0.002318	1.832155	0.000153	0.000017	0.000242

La solución aproximada es pmin=1.832155
Elapsed time = 00:00:17.08. Program returned (0). Press any key.

ANEXO 4
PROGRAMA DE NELDER-MEAD

```
/* Programa que utiliza el método de Nelder- Mead para minimizar*/
/* una función de varias variables */

#include <stdio.h>
#include <math.h>
main( )
{
int j,k,n;

float x[4],y[4],z[4],tol,h1,h2,h3,o1,o2,b1,b2,p1,p2,m1,m2,r1,r2,yo,yb,yp;
float c1,c2,e1,e2,s1,s2,ymax,ymin,aux1,aux2,aux3,aux4;
float f(float v,float w);

system("cls");
printf("\n Introduzca el número máximo de  iteraciones ");
printf("\n n= ");
scanf("%d",&n);
printf("\n Introduzca la tolerancia ");
printf("\n tol= ");
scanf("%f",&tol);
printf("\n Introduzca los vértices ");
for(j =1; j<=3; ++j)

   {
         printf("\n v(x[%d],y[%d])= ", j, j);
         scanf("%f %f",&x[j],&y[j]);

   }
for(j =1;j<=3; ++j)

printf("\n f(x[%d],y[%d])=%g ", j, j, f(x[j],y[j]));
printf("\n ");
printf("\n El punto óptimo es o(o1,o2)= ");
scanf("%f %f",&o1,&o2);
printf("\n El punto bueno es b(b1,b2)= ");
scanf("%f %f",&b1,&b2);
```

```
printf("\n El punto peor es p(p1,p2)= ");
scanf("%f %f",&p1,&p2);
printf("\n k   punto óptimo          punto bueno         punto peor ");

k=1;
while (k<= n)
{
 printf("\n %d  %g %g              %g %g               %g %g
",k,o1,o2,b1,b2,p1,p2);
  m1=(o1+b1)/2;
  m2 = (o2+b2)/2;
  r1=(2*m1)-p1;
  r2=(2*m2)-p2;

  if (f(r1,r2) < f(p1,p2))

          {
             if(f(o1,o2)< f(r1,r2))
                    {

                      p1=r1;
                      p2=r2;
                     }

                  else

                    {

                      e1=2*r1-m1;
                      e2=2*r2-m2;

                      if(f(e1,e2)< f(o1,o2))

                              {

                              p1=e1;
                              p2=e2;

                               }
```

```
                else
                 {
                     p1=r1;
                     p2=r2;
                 }

              }

          }

else
  {
        if( f(r1,r2)< f(p1,p2) )
                  {
                   p1=r1;
                   p2=r2;
                  }

         else
             {
               c1= (p1+m1)/2;
               c2=(p2+m2)/2;

               if( f(c1,c2) < f(p1,p2))
                  {
                   p1=c1;
                   p2=c2;
                  }

               else
```

```
                {

                  s1=(o1+p1)/2;
                  s2=(o2+p2)/2;
                  p1=s1;
                  p2=s2;
                  b1=m1;
                  b2=m2;

                 }

                 }
   }

z[1]=f(o1,o2);
z[2]=f(b1,b2);
z[3]=f(p1,p2);

ymax=z[1];
ymin=z[1];
if(ymax<z[2])
            ymax=z[2];
if(ymax<z[3])
            ymax=z[3];

if(ymin>z[2])
            ymin=z[2];

if(ymin>z[3])
            ymin=z[3];

if( ymax < (ymin+tol) )
       {

         printf("\n ");
         printf("\n ");

         printf("\n El vértice ptimo es o=(%g,%g)",o1,o2);
```

```
             printf("\n El mínimo local es f(o)=%g",f(o1,o2));
             exit(0);
            }
   if(f(p1,p2) < f(o1,o2))
                  {
                   aux1=o1;
                   o1=p1;
                   aux2=o2;
                   o2=p2;
                   aux3=b1;
                   aux4=b2;
                   b1=aux1;
                   b2=aux2;
                   p1=aux3;
                   p2=aux4;

                  }
         else
                  {
                   aux1=b1;
                   b1=p1;
                   aux2=b2;
                   b2=p2;
                   p1=aux1;
                   p2=aux2;

                  }
  k=k+1;

 }

}

float f(float v,float w)
{
float z;
z=pow(v,2)-4*v+pow(w,2)-w-v*w;
return (z);

}
```

Salida

Introduzca el número máximo de iteraciones
n= 30

Introduzca la tolerancia
tol= 0.1

Introduzca los vértices
v(x[1],y[1])= 0 0
v(x[2],y[2])= 1.2 0
v(x[3],y[3])= 0 0.8

f(x[1],y[1])=0
f(x[2],y[2])=-3.36
f(x[3],y[3])=-0.16

El punto óptimo es o(o1,o2)= 1.2 0

El punto bueno es b(b1,b2)= 0 0.8

El punto peor es p(p1,p2)= 0 0

k	punto óptimo		punto bueno		punto peor	
1	1.2	0	0	0.8	0	0
2	1.8	1.2	1.2	0	0	0.8
3	1.8	1.2	3	0.4	1.2	0
4	3.6	1.6	1.8	1.2	3	0.4
5	2.4	2.4	3.6	1.6	1.8	1.2
6	2.4	1.6	2.4	2.4	3.6	1.6
7	3	1.8	2.4	1.6	2.4	2.4
8	3	1.8	2.55	2.05	2.4	1.6
9	3	1.8	3.15	2.25	2.55	2.05

El vértice óptimo es o=(3,1.8)
El mínimo local es f(o)=-6.96
Elapsed time = 00:00:48.06. Program returned (0). Press any key.

ANEXO 5
PROGRAMA DEL GRADIENTE

```
/*Programa que utiliza el método del gradiente para minimizar*/
/*una función de dos variables reales a partir de un punto inicial po */

#include<stdio.h>
#include<math.h>

main( )
{
 int k,m;

 float x1,x2,nx1,nx2,df,normS,S1,S2,z,y0,y1;
 float f(float x, float y);
 float g1(float x,float y);
 float g2(float x,float y);
 system ("cls");
 printf ("\n Introduzca el punto inicial:");
 printf("\n x1=");
 scanf("%f",&x1);
 printf("\n x2=");
 scanf ("%f",&x2);
 printf ("\n Introducir la tolerancia para  f(p):");
 printf ("\n df=");
 scanf ("%f",&df);
 printf("\n Introducir el número máximo de iteraciones");
 printf("\n m=");
 scanf("%d",&m);
 printf("\nk     punto Pk      /f(Pk)-f(Pk-1)/");
 k=1;

 while (k<=m)

 {

  S1=g1(x1,x2);
  S2=g2(x1,x2);
```

```
 nx1=x1+S1;
 nx2=x2+S2;
 y0=f(x1,x2);
 y1= f(nx1,nx2);
 z = y1-y0;
while(z>0)
{
 S1=S1/2;
 S2=S2/2;
 normS=sqrt(S1*S1+S2*S2);
 if(normS<0.00001)
         break;
 else
    {
          nx1=x1+S1/2;
          nx2= x2 + S2/2;
          y1= f(nx1,nx2);
          z = y1-y0;

          }

 }

printf("\n %d    %g %g       %g ",k, nx1, nx2,fabs(z));
if(fabs(z)<df )
    {
        printf("\n Las iteraciones terminan");
        break;

   }

 x1 =nx1;
 x2 =nx2;
 k =k + 1;

}
printf("\n El punto mínimo es (%g, %g)",nx1,nx2);
printf("\n El valor mínimo es %g",f(nx1,nx2));
```

```
}

 float f(float x, float y)
  {
  float z;
  z= pow(x*x + y*y -1,2)+pow(x*x -y,2);
  return(z);
  }


 float g1(float x,float y)
  {
  float z;
  z =-(2*(x*x + y*y-1)*2*x + 2*(x*x -y)*2*x);
  return (z);
  }

 float g2(float x,float y)
  {
  float z;
  z=-(2*(x*x + y*y -1)*2*y + 2*(x*x -y)*(-1));
  return (z);
  }
```

Salida

Introduzca el punto inicial:
x1=0.5

x2=0.5

Introducir la tolerancia para f(p):
df=0.00001

Introducir el número máximo de iteraciones
m=30

k	punto Pk		/f(Pk)-f(Pk-1)/
1	0.875	0.625	0.268311
2	0.745117	0.611328	0.0359869
3	0.792509	0.619022	0.00799388
4	0.784462	0.617799	0.000194137
5	0.786565	0.618104	1.36311e-005
6	0.786047	0.618021	8.25193e-007

Las iteraciones terminan
El punto mínimo es (0.786047,0.618021)
El valor mínimo es 5.54618e-008
Elapsed time = 00:00:12.63. Program returned (33). Press any key.

ANEXO 6
PROGRAMA DEL GRADIENTE PARA RESOLVER SISTEMAS NO LINEALES

```
/*Programa que utiliza el método del gradiente para resolver */
/*un sistema de tres ecuaciones no lineales mediante la minimización de*/
/*una función de tres variables reales a partir de un punto inicial po */

#include<stdio.h>
#include<math.h>

main( )
{
  int k,m;

  float x1,x2,x3,nx1,nx2,nx3,df,normS,S1,S2,S3,z,y0,y1;
  float f(float x, float y, float w);
  float g1(float x,float y, float w);
  float g2(float x,float y, float w);
  float g3(float x,float y, float w);
  system("cls");
  printf("\n Introduzca el punto inicial:");
  printf("\n x1=");
  scanf("%f",&x1);
  printf("\n x2=");
  scanf("%f",&x2);
  printf("\n x3=");
  scanf("%f",&x3);
  printf("\n Introducir la tolerancia para  f(p):");
  printf("\n df =");
  scanf("%f",&df);
  printf("\n Introducir el número máximo de iteraciones");
  printf("\n m=");
  scanf("%d",&m);
  printf("\nk     punto Pk      /f(Pk)-f(Pk-1)/");
  k=1;

  while (k<=m)
```

```
{

  S1=g1(x1,x2,x3);
  S2=g2(x1,x2,x3);
  S3=g3(x1,x2,x3);
  nx1=x1+S1;
  nx2=x2+S2;
  nx3=x3+S3;
  y0=f(x1,x2,x3);
  y1= f(nx1,nx2,nx3);

  z = y1-y0;
 while(z>0)

 {

  S1=S1/2;
  S2=S2/2;
  S3=S3/2;
  normS=sqrt(S1*S1+S2*S2+S3*S3);
  if(normS<0.00001)
        break;
  else

     {
        nx1=x1+S1/2;
        nx2=x2+S2/2;
        nx3=x3+S3/2;
        y1=f(nx1,nx2,nx3);
        z=y1-y0;

        }
  }

printf("\n %d    %g %g %g      %g ", k, nx1, nx2, nx3, fabs(z));

 if( fabs(z)< df)

 {
```

```
printf("\n Las iteraciones terminan");
break;

      }
   x1 =nx1;
   x2 =nx2;
   x3 =nx3;
   k = k + 1;

 }
 printf("\n El punto mínimo es (%g, %g, %g)",nx1,nx2,nx3);
 printf("\n El valor mínimo es %g",f(nx1,nx2,nx3));


}

  float f(float x, float y, float w)
   {

   float z;
   z= pow(x*x + y*y + w*w +10*x-4,2)+ pow(x*x - y*y + w*w + 10*y-
5,2)+pow(x*x + y*y - w*w + 10*w-6,2);
   return(z);

   }

  float g1(float x,float y, float w)

   {

   float z;
   z=-(2*(x*x + y*y + w*w +10*x-4)*(2*x + 10)+2*(x*x - y*y + w*w +
10*y-5)*(2*x)+2*(x*x + y*y-w*w + 10*w-6)*(2*x));
   return (z);

   }

float g2(float x,float y, float w)
   {
```

```
float z;
z=-(2*(x*x + y*y + w*w + 10*x-4)*(2*y)+2*(x*x - y*y + w*w + 10*y-
5)*(-2*y + 10)+2*(x*x + y*y -  w*w + 10-6)*(2*y));
  return (z);

  }

 float g3(float x,float y, float w)

  {

  float z;
  z=-(2*(x*x + y*y + w*w +10*x-4)*(2*w)+ 2*(x*x -y*y + w*w + 10*y-
5)*(2*w)+2*(x*x + y*y - w*w  +10*w-6)*(-2*w + 10));
  return (z);

  }
```

Salida

Introduzca el punto inicial:
x1=0

x2=0

x3=0

Introducir la tolerancia para f(p):
df=0.00001

Introducir el número máximo de iteraciones
m=30

k	punto Pk			/f(Pk)-f(Pk-1)/
1	0.625	0.78125	0.9375	35.662
2	0.193968	0.451672	0.604167	39.1371
3	0.432136	0.457931	0.648212	0.695187
4	0.332611	0.423003	0.605755	1.28283
5	0.348617	0.454587	0.629703	0.105711
6	0.339571	0.444943	0.621559	0.0267713
7	0.339567	0.444938	0.621555	5.23031e-006

Las iteraciones terminan
El punto mínimo es (0.339567,0.444938, 0.621555)
El valor mínimo es 0.0903689
Elapsed time = 00:00:08.74. Program returned (30). Press any key.

BIBLIOGRAFIA

- Bundnick, F. Matemáticas Aplicadas para Administración, Economía y Ciencias Sociales. Cuarta Edición. México D.F. 1992.

- Burden, R. y Faires J. Análisis Numérico. Segunda Edición. Editorial Iberoamérica S.A de C.V. México D.F. 1996.

- Fausett V. Laurene. Applied Numerical Análisis using Matlab. Tercera Edición. Editorial Prentice Hall. Madrid. 1999.

- Galkin, V. Aplicación de la Computación en la Planificación y Optimización. Segunda Edición Editorial Mexicana. México. 1989.

- Mathews, J. y Fink, K. Métodos Numéricos con Matlab. Tercera Edición. Editorial Prentice Hall. Madrid . 2000.

- Mathur, Kamlesh y Solow, Daniel. Investigación de Operaciones. Primera Edición. Editorial McGraw-Hill. México. 1972.

- Piura L., Julio. Metodología de la Investigación científica: Un enfoque integrador. Editorial PAVSA. Managua. 2006.

- Zill, D. Cálculo con Geometría Analítica. Primera Edición. Editorial Ibero América S.A de C.V. México. 1987.