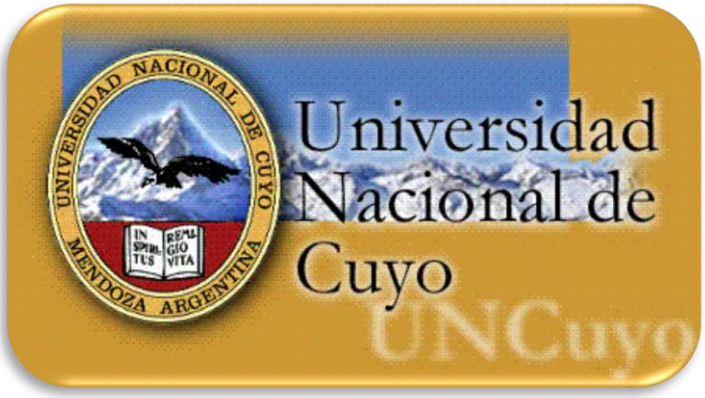
Doctorado en Ingeniería



Métodos Numéricos 2009

Estudio de Casos Prácticos

Master Ing. Pablo DE SIMONE

Ing. Roberto HAARTH

PREFACIO

Lo único que se tiene en mente al escribir,
es el primer párrafo, lo demás surge
por voluntad y dedicación.

Los autores.2010

El contenido de este trabajo surge de la necesidad de cumplir con un objetivo común de los autores, aprovechar y ocupar el espacio destinado a la enseñanza y aprendizaje de conocimientos en el área de las matemáticas aplicadas a ingeniería.
Es el deseo que el contenido de este texto sea de utilidad para aquellas personas que realizan una formación de grado o posgrado, pretendiendo ser éste, una guía introductoria fuertemente orientada en lo práctico a través del desarrollo de ejercicios propuestos por el Dr. Richard Branham en el curso de formación doctoral impartido en la Facultad de Ingeniería-U.N.Cuyo del ciclo 2009-2010.
El lector podrá observar, a medida que se introduce en los diferentes casos, un desarrollo que se inicia con el enunciado del ejercicio, le sigue un contenido teórico referido al tema y finalmente, el abordaje propiamente dicho de la resolución del caso.
Para dar consistencia al trabajo, los autores presentan los algoritmos utilizados para la realización de la programación secuencial, iteraciones y gráficos que resultan de aplicar el software específico en cada caso. La finalidad es puramente educativa, haciendo referencia cuando se necesita al software de uso con el único fin determinado que no sea el exclusivamente académico.
Los autores agradecen a aquellas personas que, sin saberlo, colaboraron respondiendo las consultas y aportando consejos útiles sobre la metodología adecuada de uso para la solución de los casos prácticos que se presentan.
Este aporte de material no hace más que depositar en el escritorio del lector un complemento adicional a la abundante bibliografía existente sobre el tema Métodos Numéricos.
Es el espíritu de los autores, que nuevas generaciones de estudiantes mejoren los métodos y algoritmos utilizados en la resolución de casos, incorporando nuevos ejemplos con la única finalidad de lograr la continua mejora de la enseñanza en el área de las matemáticas aplicadas a ingeniería.

AGRADECIMIENTOS

Los autores agradecen nuevamente a aquellas personas que, sin saberlo, aportaron con respuestas y consejos útiles sobre la resolución de los ejercicios desarrollados.

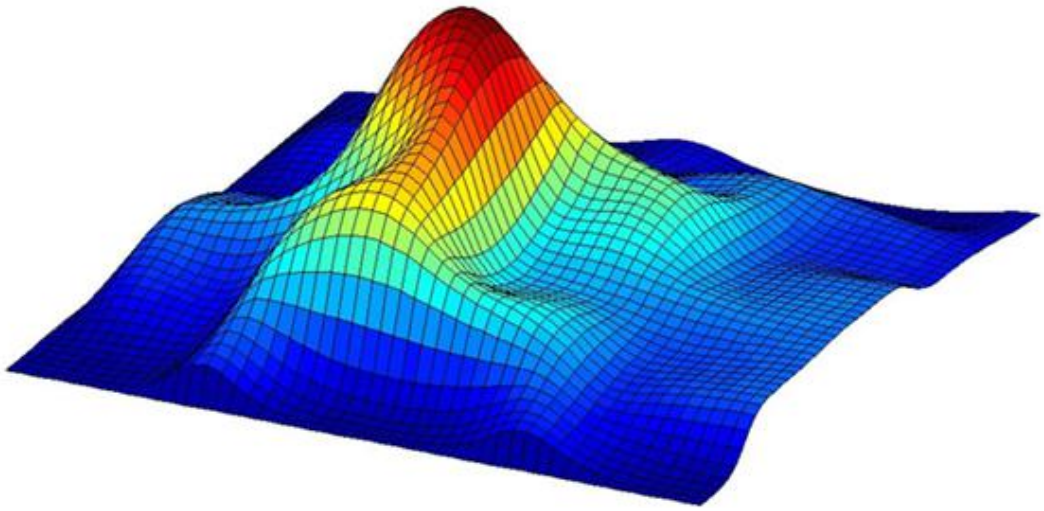
Finalmente, queda un profundo agradecimiento para la Facultad de Ingeniería-U.N.Cuyo, al Sr. Decano Ing. Marcelo Estrella Orrego, autoridades, y en particular, una mención especial al director y personal de la Biblioteca de la Facultad, por permitir que este material quede depositado para consulta, junto a otras obras bibliográficas de elevado valor cultural.

I N D I C E

La metodología utilizada para el estudio y resolución de los casos propuestos es la siguiente:

- ✓ **Enunciado**: Datos generales y propuestas de investigación.
- ✓ **Teoría**: Bibliografía y fundamentos en cuáles se sustenta el desarrollo del caso.
- ✓ **Resolución del caso práctico**: Comprende el estudio del caso propiamente dicho (desarrollo de fórmulas, gráficos, conclusiones, etc.).

Caso Práctico nº 1



Métodos Numéricos 2009

Master Ing. Pablo DE SIMONE

Ing. Roberto HAARTH

Caso Práctico nº 1

✓ Enunciado:

Encontrar el valor que converge la serie para un valor de x= 0.5
Calcular la convergencia con un error de e< 0.5 *10 $^{-8}$
Utilizar el valor de ∅ (x=1) = 1 para asegurar la convergencia.

$$\emptyset(x) = \sum_{k=1}^{\infty} \frac{1}{\left(k(k+x)\right)}$$

✓ Teoría:

Series Numéricas. Convergencia

Sea a_n una sucesión de números reales, se define la serie general a_n y se escribe

$$\sum_{n=1}^{\infty} a_n = \lim (a_1 + \cdots + a_n).$$

Si el límite de la n-ésima suma parcial a1 + · · · + an "es finito", se dice que la serie es "convergente"; si "es infinito o no existe", que es "divergente".

1. PROPIEDADES DE LAS SERIES

1_El carácter (convergente o divergente) de una serie no cambia si se modifica un número finito de sus términos.

2_Para que la serie $\sum a_n$ converja es necesario que lim a_n = 0.

3_Si las series $\sum a_n$ y $\sum b_n$ convergen, entonces: $\sum a_n + b_n$ y $\sum \lambda a_n$ con $\lambda \in \Re$ también, teniéndose

$$\sum (a_n + b_n) = \sum a_n + \sum b_n \quad \text{y} \quad \sum \lambda a_n = \lambda \sum a_n.$$

CONVERGENCIA DE LA SERIE

Para estudiar la convergencia de la serie se consideran las formas de operar matemáticamente

Opcion1: Ecuación original

$$\emptyset(x) = \sum_{k=1}^{\infty} \frac{1}{\left(k(k+x)\right)}$$

Opción 2: Ecuación aplicando consideración del error

$$\emptyset(x) = \sum_{k=1}^{\infty} \frac{1}{(k^2+kx)}$$

Por recomendación de teoría de errores, si se tiene:
a*(b+c) entonces es conveniente para disminuir el error de arrastre hacer: (a*b + a*c)

METODOS DE ACELERACION DE LA CONVERGENCIA

El estudio de las series de números reales no termina con el análisis de la convergencia la sumatoria de algunas series sencillas, como la geométrica o las telescópicas.

En la práctica, cuando aparece una serie se requiere la suma, ésta no siempre puede hacerse de forma exacta, por lo que hay que recurrir al procedimiento de aproximar el valor de la serie por el de su suma parcial para un índice suficientemente grande.

Lamentablemente, este procedimiento "ingenuo" no siempre da resultado porque, en muchos casos, la sucesión de las sumas parciales converge muy lentamente.

Para solventar este inconveniente existen métodos que transforman la serie en otra de la misma suma, pero cuyas sumas parciales convergen "más rápidamente"; estos métodos se conocen como métodos de aceleración de la convergencia.

Estos métodos de aceleración de la convergencia permiten encontrar el valor con un número menor de iteraciones.

Para fijar ideas consideremos el ejemplo de la serie armónica:

$$S = \sum_{n=1}^{\infty} \frac{1}{n^2}$$

Vamos a calcular aproximadamente su suma utilizando la suma parcial n-ésima S_n, cometiendo un error menor que 10^{-3}. El problema es determinar el valor de n, por lo que se acota el resto

$$R_n = \frac{1}{(n+1)^2} + \frac{1}{(n+2)^2} + \cdots.$$

Se tiene entonces:

$$\frac{1}{(n+k)^2} \leq \frac{1}{(n+k-1)(n+k)},$$

Por lo tanto:

$$R_n \leq \frac{1}{n(n+1)} + \frac{1}{(n+1)(n+2)} + \cdots = \frac{1}{n}.$$

EL METODO DE KUMMER

Una posible forma de "acelerar" la convergencia de esta serie es utilizar el siguiente truco: supongamos que queremos calcular la suma de la serie:

$$S = \sum_{n=1}^{\infty} a_n$$

y que conocemos la suma de otra serie

$$B = \sum_{n=1}^{\infty} b_n$$

cuyo término general está relacionado con el de la primera por la condición

$$\lim_{n\to\infty} \frac{a_n}{b_n} = l \neq 0.$$

Comprueba que:

$$S = \sum_{n=1}^{\infty} a_n = lB + \sum_{n=1}^{\infty} \left(1 - l\frac{b_n}{a_n}\right) a_n.$$

Como $1 - l\frac{b_n}{a_n} \to 0$ cuando $n \to \infty$, los términos de la nueva serie son menores que los de la original. Se ha transformado entonces la serie de partida en otra cuyas sumas parciales convergen más rápidamente.

$$\sum_{n=1}^{\infty} \frac{1}{n(n+1)}$$

Para aplicar este método a la serie armónica vamos considerar la serie cuya suma es 1.

Entonces, aplicando la fórmula anterior:

$$S = \sum_{n=1}^{\infty} \frac{1}{n^2} = 1 + \sum_{n=1}^{\infty} \frac{1}{n^2(n+1)}.$$

Sabiendo que $S=\Pi^2/6$, comprueba que sumando 8 términos en esta última fórmula se consigue la suma de la serie armónica con un error menor que 10^{-5}

EL METODO DE AITKEN

Se aplica el algoritmo de aceleración basado en el método de Aitken, tratando de producir una serie que converja más rápido para (1/(k*(k+x)):

El método de Aitken consiste en considerar que cada término de la serie conserva una proporcionalidad con el término anterior, de modo que:

$$x_{n+1} \approx C * x_n \quad \text{y} \quad x_{n+2} \approx C * x_{n+1}$$

Eliminando C en ambos términos y despejando

$$\bar{x}_{n+2} = x_{n+2} - \frac{(x_{n+2} - x_{n+1})^2}{x_{n+2} - 2x_{n+1} + x_n}$$

✓ Resolución del Caso Práctico nº 1:

$$\emptyset(x) = \sum_{k=1}^{\infty} \frac{1}{(k(k+x))}$$

Para nuestro caso, se tiene que k*(k+x) = (k^2 + k*x)

Dado el error e< 0.5 *10^{-8} Se considera en la serie que este valor se obtiene de considerar el valor de la f(k) – f(k-1) y con redondeo porque la acumulación de errores es menor.

Para el valor de x=0.5 se calcula la convergencia de la ecuación asignando valores a k.

El cálculo numérico se desarrolla utilizando Matlab.

Algoritmo:

```
% Serie Convergente opcion 2
% Ensayo con opcion1 sumatoria y= 1/(k^2+k*0.5)
% Metodos numericos 2009-2010
double presicion
k=0 ;
sum1=0 ;
valorsuma=0 ;
valora=0 ;
error1=1 ;
tini=clock;
while error1 > 0.5 * 10 ^-8
k=k+1
y1=k^2
y2=k*0.5
y=1/(y1+y2)
valorsuma=sum1+y
valorsuma1=valorsuma
valora=sum1
error1= abs(valorsuma-valora)
sum1=valorsuma
end
tend=clock;
format long
%PRESENTACION DE DATOS FINALES
sumatoria=valorsuma
iteraciones=k
error=error1
tiemposeg= etime(tend,tini)
Operaporseg=iteraciones/tiempo
```

RESULTADOS:

sumatoria = 1.227340570153880

iteraciones = 14142

error = 4.999919145376452e-009

tiemposeg = 10.809999999999999

Operaporseg = 1.308233117483811e+003

Se requieren 14142 iteraciones para lograr la convergencia según el error indicado.

Desarrollo de la convergencia por método kummer

Para nuestro caso particular:

$$\sum_{n=1}^{\infty} a_n = \sum_{k=1}^{\infty} \frac{1}{(k(k+x))}$$

$$B = \sum_{n=1}^{\infty} b_{n=} \sum_{n=1}^{\infty} \frac{1}{n^2}$$

$$\lim_{n\to\infty} \sum_{n=1}^{n} b(n) \to \frac{1}{6} \cdot \pi^2$$

$$S = \sum_{n=1}^{\infty} a_n = lB + \sum_{n=1}^{\infty} \left(1 - l\frac{b_n}{a_n}\right) a_n .$$

Para $\lim_{n\to\infty} \frac{a(n)}{b(n)} \to 1$ $L := 1$

El valor inicial se la serie es l*B = 1.645 (1.64494176)

$$S = 1.645 + \sum_{n=1}^{\infty} \left(1 - L\frac{b_n}{a_n}\right) a_n$$

$$S(n) = \left(1 - L\frac{b_n}{a_n}\right) a_n$$

$$S(n) = \frac{\left[1 - \frac{1}{k}(k+1)\right]}{[k(k+0.5)]}$$

Algoritmo:

```
% Ensayo con metodo acelerado kummer
double presicion
k=0 ;sum1=0 ; valorsuma=0 ; valora=0 ; error1=1 ;
s=1.645; s1=0; s2=0; s3=0; s4=0;
s5=1-2
tini=clock;
while error1 > 0.5 * 10 ^-8
k=k+1
s1=(k+0.5)
s2=1-(s1/k)
s3=k*(k+0.5)
s4=s2/s3
valorsuma=sum1+s4
%valorsuma1=valorsuma
%valora=sum1
vactual=valorsuma+s
valora=sum1+s
error1= abs(vactual-valora)
sum1=valorsuma
end
tend=clock;
%PRESENTACION DE DATOS FINALES
sumatoria=vactual
iteraciones=k
error=error1
tiemposeg= etime(tend,tini)
Operaporseg=iteraciones/tiempo
```

RESULTADOS (método Kummer)

sumatoria = 1.227478368773328
iteraciones = 464
error = 4.999750391476709e-009
tiemposeg = 0.109000000000002
Operaporseg = 4.256880733944885e+003

Por el método Kummer se requieren solo 464 iteraciones contra las 14142 iteraciones convencionales.

RESULTADO método Aitken:
El ensayo realizado con AITKEN resultó ser más lento que el Método de Kummer por lo que no se consideran sus resultados como concluyentes.
El método Aitken resulta adecuado para acelerar la convergencia sin pérdida de precisión en los cálculos.

Caso Práctico nº 2

Métodos Numéricos 2009

Master Ing. Pablo DE SIMONE

Ing. Roberto HAARTH

Caso Práctico nº 2

✓ Enunciado:

Calcular las raíces del polinomio:

$$f(x) = 1 * x^7 + 6 * x^6 + 6 * x^5 - 21 * x^4 - 36 * x^3 + 3 * x^2 + 13 * x + 3$$

Coeficientes:

a_7= 1 a_6= 6 a_5=6 a_4 = -21 a_3= -36 a_2=3 a_1=13 a_0= 3

✓ Teoría:

Conceptos sobre los Métodos

Método de la Bisección:

El **método de bisección** es un algoritmo de búsqueda de raíces que trabaja dividiendo el intervalo a la mitad y seleccionando el subintervalo que tiene la raíz.

Supóngase que queremos resolver la ecuación *f*(*x*) = 0 (donde f es continua. Dados dos puntos *a* y *b* tal que *f*(*a*) y *f*(*b*) tengan signos distintos, sabemos por el Teorema de Bolzano que *f* debe tener, al menos, una raíz en el intervalo [*a*, *b*]. El método de bisección divide el intervalo en dos, usando un tercer punto *c* = (*a*+*b*) / 2. En este momento, existen dos posibilidades: *f*(*a*) y *f*(*c*), ó *f*(*c*) y *f*(*b*) tienen distinto signo. El algoritmo de bisección se aplica al subintervalo donde el cambio de signo ocurre.

El método de bisección es menos eficiente que el método de Newton, pero es mucho más seguro asegurar la convergencia.

Si *f* es una función continua en el intervalo [*a*, *b*] y *f*(*a*)*f*(*b*) < 0, entonces este método converge a la raíz de *f*. De hecho, una cota del error absoluto es:

$$\frac{|b-a|}{2^n}$$

en la *n*-ésima iteración. La bisección converge linealmente, por lo cual es un poco lento. Sin embargo, se garantiza la convergencia si f(*a*) y f(*b*) tienen distinto signo.

Si existieran más de una raíz en el intervalo entonces el método sigue siendo convergente pero no resulta tan fácil caracterizar hacia qué raíz converge el método.

Método de Newton:

Está basado en el uso de una línea tangente como aproximación de f(x), cerca de los puntos donde el valor de la función es cero.

1.- Escoger un número inicial (x0)
2.- Calcular la siguiente aproximación de x1 utilizando la fórmula:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

3.- **Si** | xn-xn+1 | < ε **entonces** xn+1 es una raíz
De otra forma pasar al punto 2

En el método de Newton-Raphson la única manera de alcanzar la convergencia es seleccionar un valor inicial lo suficientemente cercano a la raíz buscada. Así, se ha de comenzar la iteración con un valor razonablemente cercano al cero (denominado punto de arranque o valor supuesto). La relativa cercanía del punto inicial a la raíz depende mucho de la naturaleza de la propia función; si ésta presenta múltiples puntos de inflexión o pendientes grandes en el entorno de la raíz, entonces las probabilidades de que el algoritmo diverja aumentan, lo cual exige seleccionar un valor supuesto cercano a la raíz. Una vez se ha hecho esto, el método linealiza la función por la recta tangente en ese valor supuesto. La abscisa en el origen de dicha recta será, según el método, una mejor aproximación de la raíz que el valor anterior. Se realizarán sucesivas iteraciones hasta que el método haya convergido lo suficiente.

Sea *f* : [*a*, *b*] -> **R** función derivable definida en el intervalo real [*a*, *b*]. Empezamos con un valor inicial x_0 y definimos para cada número natural *n*

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}.$$

Donde *f* ' denota la derivada de *f*.

Nótese que el método descrito es de aplicación exclusiva para funciones de una sola variable con forma analítica o implícita cognoscible. Existen variantes del método aplicables a sistemas discretos que permiten estimar las raíces de la tendencia, así como algoritmos que extienden el método de Newton a sistemas multivariables, sistemas de ecuaciones, etc.

Método de la secante

En análisis numérico el **método de la secante** es un método para encontrar los ceros de una función de forma iterativa.

Es una variación del método de Newton-Raphson donde en vez de calcular la derivada de la función en el punto de estudio, teniendo en mente la definición de derivada, se aproxima la pendiente a la recta que une la función evaluada en el punto de estudio y en el punto de la iteración anterior. Este método es de especial interés cuando el coste computacional de derivar la función de estudio y evaluarla es demasiado elevado, por lo que el método de Newton no resulta atractivo.

El método se define por la relación de recurrencia:

$$x_{n+1} = x_n - \frac{x_n - x_{n-1}}{f(x_n) - f(x_{n-1})} f(x_n).$$

El método se basa en obtener la ecuación de la recta que pasa por los puntos (x_{n-1}, $f(x_{n-1})$) y (x_n, $f(x_n)$). A dicha recta se le llama *secante* por cortar la gráfica de la función. Posteriormente se escoge como siguiente elemento de la relación de recurrencia, x_{n+1}, la intersección de la recta secante con el eje de abscisas obteniendo la fórmula.

Método de Brent

Richard Brent vislumbró una rutina que combinaba la confiabilidad del método de bisección con la velocidad del método de la secante, y añadió otro método que puede ser más rápido. La idea es que se comienza con un cambio de signo en el intervalo, del cual no se sale. Entonces se tiene tres alternativas para realizar el siguiente paso:

- El paso de bisección (lento);
- El paso de la secante (mediano);
- El paso "cuadrático inverso" (rápido);

Si la iteración resulta bastante bien, se toma el paso rápido. En el caso de que se esté en el paso de la secante o cuadrático inverso y no mejora, se toma el paso de bisección.

Como se ve, este método bien complicado para codificar. Este tipo de programa aplica el concepto de adaptatividad, lo que significa que se "adapta y ajusta según la experiencia previa" del problema.

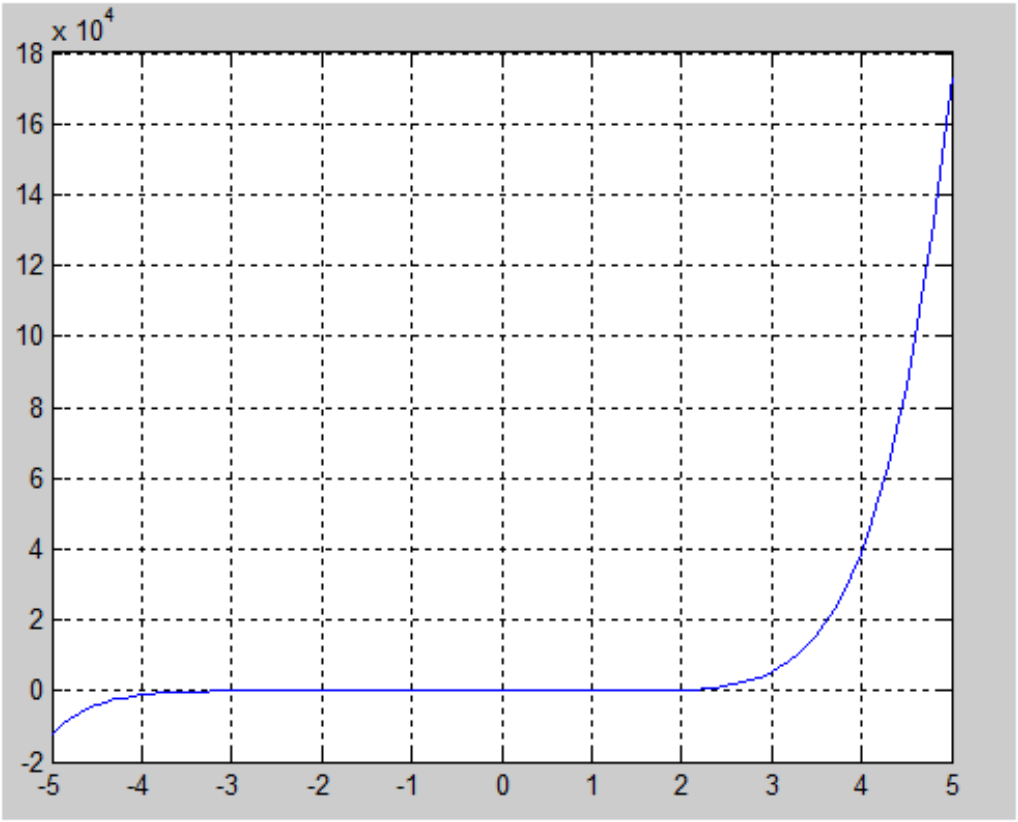
✓ Resolución del Caso Práctico nº 2:

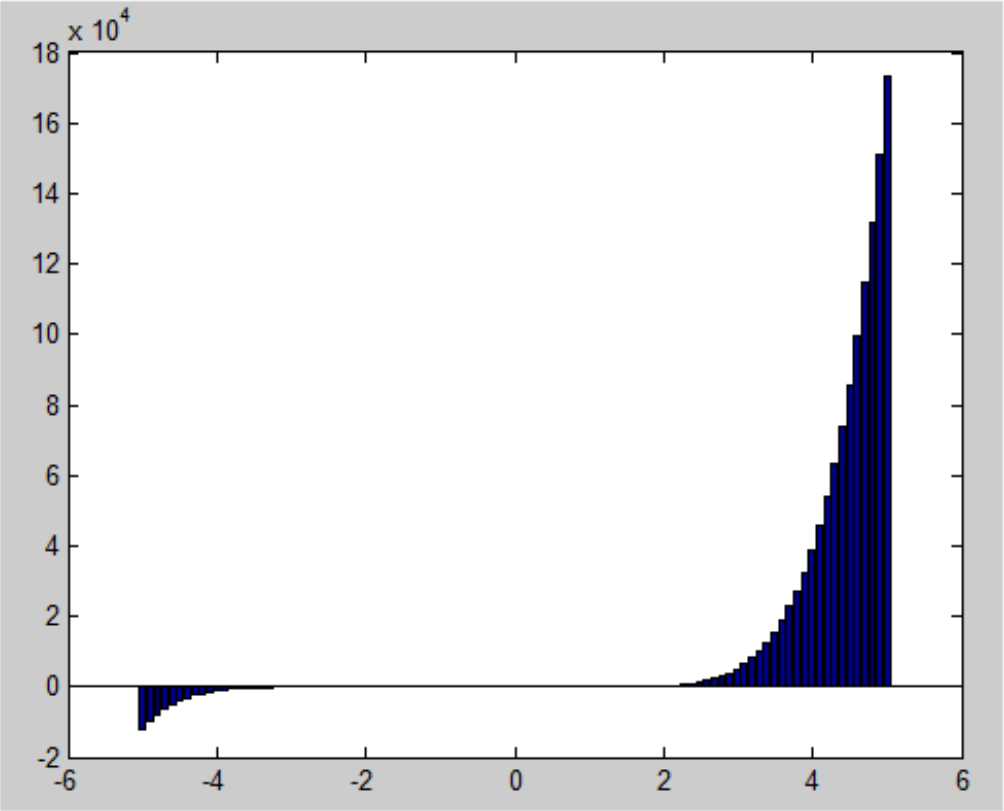
BÚSQUEDA DE LAS RAÍCES APLICANDO MÉTODOS NUMÉRICOS

El polinomio se puede graficar haciendo un barrido de un rango para x con el objeto de determinar los cambios de signos.

Se utiliza en esta ocasión MATLAB para graficar la función

```
%CALCULO DE RAICES-grafica
x=linspace(-5,+5,100);
%x=-4:.01:4;
y=1*x.^7+6*x.^6+6*x.^5-21*x.^4-36*x.^3-3*x.^2+13*x+3;
plot(x,y)
grid
```





Del gráfico se observa que el intervalo de exploración es -3< x < +3 o bien -4< x < +4
El gráfico muestra también que la función se comporta tangente al eje x cuando pasa por cero.

Surge la pregunta:
¿Cuáles son los valores de x que hacen que f(x)=0 en el intervalo considerado?

Se debe tener en consideración que los métodos de intervalo cerrado (Bisección, Regula Falsi, secante) no funcionan o son inaplicables (cuando la función es tangente al eje x)

Otro problema es que cerca de la solución, la derivada tiende a cero, lo cual provoca que el algoritmo de Newton-Raphson tenga problemas de convergencia al tener una división por cero.

Se requiere utilizar métodos de intervalo abierto (método que no dependa de la variable inicial para comenzar a iterar cerca del cero).

El polinomio, al hacerse tangente en f(x)=0 presenta raíces dobles.

Raíces del Polinomio:
Utilizando MATLAB y su función “roots” se pueden calcular las raíces del polinomio

```
%CALCULO DE RAICES
p=[1 6 6 -21 -36 3 13 3]
r=roots(p)
```

Los resultados muestran que la ecuación tiene 7 raíces (con Matlab)

```
p =
     1     6     6   -21   -36     3    13     3
r =
 -2.618033988749875 + 0.000000059704635i
 -2.618033988749875 - 0.000000059704635i
 -2.490863615361067
  1.834243184313920
  0.656620431047111
 -0.381966011250105 + 0.000000003982789i
 -0.381966011250105 - 0.000000003982789i
```

METODO BISECCION:

```
% METODO BISECCION
clc
format long
A=0; C=0; I=0; G=0; Num=0; Cp=0; B=0;
fprintf('\t\tMétodo de Bisección\n\n');
Fx=input('\nDigite la Función: ','s');
A=input('\nDigite el Valor de A: ');
C=input('\nDigite el Valor de C: ');
Num=input('\nDigite el Número de Iteraciones: ');
Cp=input('\nDigite el Criterio de Parada: ');
Er=Cp+1;
x=A;
fA=eval(Fx);
x=C;
fC=eval(Fx);
fprintf('\n\n');
if (fA*fC > 0)
fprintf('\n\nNo existen Raíces en esta Ecuación');
else
while (Er>Cp & I<Num)
I=I+1;
Ant=B;
G=A+C;
B=G/2;
x=B;
fB=eval(Fx);
if (fA*fB<=0) C=B; fC=fB;
else     A=B; fA=fB;
end
Er=abs(((B-Ant) /B)*100);
fprintf('A=%.3f    B=%.3f    C= %.3f    fA=%.3f    fB=%.3f    fC=%.3f    Er= %.3f',A,B,C,fA,fB,fC,Er);
fprintf('\n');
end
fprintf('\n\nLa Raíz es: %.3f',B);
x=-10:.1:10;
ezplot(Fx);
end
```

Se realizan algunas iteraciones

Digite la Función: 1*x.^7+6*x.^6+6*x.^5-21*x.^4-36*x.^3+3*x.^2+13*x+3
Digite el Valor de A: -2
Digite el Valor de C: 1.5
Digite el Número de Iteraciones: 10000
Digite el Criterio de Parada: 0.000001

A=-0.250 B=-0.250 C= 1.500 fA=0.414 fB=0.414 fC=-67.570 Er= 100.000
A=0.625 B=0.625 C= 1.500 fA=1.271 fB=1.271 fC=-67.570 Er= 140.000
A=0.625 B=1.063 C= 1.063 fA=1.271 fB=-31.459 fC=-31.459 Er= 41.176
A=0.625 B=0.844 C= 0.844 fA=1.271 fB=-11.128 fC=-11.128 Er= 25.926
A=0.625 B=0.734 C= 0.734 fA=1.271 fB=-3.863 fC=-3.863 Er= 14.894
A=0.625 B=0.680 C= 0.680 fA=1.271 fB=-1.035 fC=-1.035 Er= 8.046
A=0.652 B=0.652 C= 0.680 fA=0.182 fB=0.182 fC=-1.035 Er= 4.192
A=0.652 B=0.666 C= 0.666 fA=0.182 fB=-0.410 fC=-0.410 Er= 2.053
A=0.652 B=0.659 C= 0.659 fA=0.182 fB=-0.110 fC=-0.110 Er= 1.037
A=0.656 B=0.656 C= 0.659 fA=0.037 fB=0.037 fC=-0.110 Er= 0.521
A=0.656 B=0.657 C= 0.657 fA=0.037 fB=-0.037 fC=-0.037 Er= 0.260
A=0.657 B=0.657 C= 0.657 fA=0.000 fB=0.000 fC=-0.037 Er= 0.130
A=0.657 B=0.657 C= 0.657 fA=0.000 fB=-0.018 fC=-0.018 Er= 0.065
A=0.657 B=0.657 C= 0.657 fA=0.000 fB=-0.009 fC=-0.009 Er= 0.033
A=0.657 B=0.657 C= 0.657 fA=0.000 fB=-0.004 fC=-0.004 Er= 0.016
A=0.657 B=0.657 C= 0.657 fA=0.000 fB=-0.002 fC=-0.002 Er= 0.008
A=0.657 B=0.657 C= 0.657 fA=0.000 fB=-0.001 fC=-0.001 Er= 0.004
A=0.657 B=0.657 C= 0.657 fA=0.000 fB=-0.000 fC=-0.000 Er= 0.002
A=0.657 B=0.657 C= 0.657 fA=0.000 fB=-0.000 fC=-0.000 Er= 0.001
A=0.657 B=0.657 C= 0.657 fA=0.000 fB=0.000 fC=-0.000 Er= 0.001
A=0.657 B=0.657 C= 0.657 fA=0.000 fB=-0.000 fC=-0.000 Er= 0.000
A=0.657 B=0.657 C= 0.657 fA=0.000 fB=0.000 fC=-0.000 Er= 0.000
A=0.657 B=0.657 C= 0.657 fA=0.000 fB=-0.000 fC=-0.000 Er= 0.000
A=0.657 B=0.657 C= 0.657 fA=0.000 fB=-0.000 fC=-0.000 Er= 0.000
A=0.657 B=0.657 C= 0.657 fA=0.000 fB=-0.000 fC=-0.000 Er= 0.000
A=0.657 B=0.657 C= 0.657 fA=0.000 fB=-0.000 fC=-0.000 Er= 0.000
A=0.657 B=0.657 C= 0.657 fA=0.000 fB=0.000 fC=-0.000 Er= 0.000
A=0.657 B=0.657 C= 0.657 fA=0.000 fB=0.000 fC=-0.000 Er= 0.000
A=0.657 B=0.657 C= 0.657 fA=0.000 fB=0.000 fC=-0.000 Er= 0.000

La Raíz es: 0.657>>

Resultado
Bisección acusa una raíz en x= 0.657 en el intervalo -2 < x < 1.5

Digite la Función: 1*x.^7+6*x.^6+6*x.^5-21*x.^4-36*x.^3+3*x.^2+13*x+3

Digite el Valor de A: -3

Digite el Valor de C: 0

Digite el Número de Iteraciones: 10000

Digite el Criterio de Parada: .0001

A=-3.000 B=-1.500 C= -1.500 fA=-9.000 fB=11.133 fC=11.133 Er= 100.000
A=-3.000 B=-2.250 C= -2.250 fA=-9.000 fB=1.352 fC=1.352 Er= 33.333
A=-2.625 B=-2.625 C= -2.250 fA=-0.000 fB=-0.000 fC=1.352 Er= 14.286
A=-2.625 B=-2.438 C= -2.438 fA=-0.000 fB=0.097 fC=0.097 Er= 7.692
A=-2.531 B=-2.531 C= -2.438 fA=-0.020 fB=-0.020 fC=0.097 Er= 3.704
A=-2.531 B=-2.484 C= -2.484 fA=-0.020 fB=0.007 fC=0.007 Er= 1.887
A=-2.508 B=-2.508 C= -2.484 fA=-0.013 fB=-0.013 fC=0.007 Er= 0.935
A=-2.496 B=-2.496 C= -2.484 fA=-0.005 fB=-0.005 fC=0.007 Er= 0.469
A=-2.496 B=-2.490 C= -2.490 fA=-0.005 fB=0.001 fC=0.001 Er= 0.235
A=-2.493 B=-2.493 C= -2.490 fA=-0.002 fB=-0.002 fC=0.001 Er= 0.118
A=-2.492 B=-2.492 C= -2.490 fA=-0.001 fB=-0.001 fC=0.001 Er= 0.059
A=-2.491 B=-2.491 C= -2.490 fA=-0.000 fB=-0.000 fC=0.001 Er= 0.029
A=-2.491 B=-2.491 C= -2.491 fA=-0.000 fB=0.000 fC=0.000 Er= 0.015
A=-2.491 B=-2.491 C= -2.491 fA=-0.000 fB=0.000 fC=0.000 Er= 0.007
A=-2.491 B=-2.491 C= -2.491 fA=-0.000 fB=-0.000 fC=0.000 Er= 0.004
A=-2.491 B=-2.491 C= -2.491 fA=-0.000 fB=0.000 fC=0.000 Er= 0.002
A=-2.491 B=-2.491 C= -2.491 fA=-0.000 fB=0.000 fC=0.000 Er= 0.001
A=-2.491 B=-2.491 C= -2.491 fA=-0.000 fB=-0.000 fC=0.000 Er= 0.000
A=-2.491 B=-2.491 C= -2.491 fA=-0.000 fB=0.000 fC=0.000 Er= 0.000
A=-2.491 B=-2.491 C= -2.491 fA=-0.000 fB=0.000 fC=0.000 Er= 0.000
A=-2.491 B=-2.491 C= -2.491 fA=-0.000 fB=0.000 fC=0.000 Er= 0.000

La Raíz es: -2.491>>

Resultado
Bisección acusa una raíz en x= -2.491 en el intervalo -3 < x < 0

Digite la Función: 1*x.^7+6*x.^6+6*x.^5-21*x.^4-36*x.^3+3*x.^2+13*x+3

Digite el Valor de A: 1

Digite el Valor de C: 3

Digite el Número de Iteraciones: 1000

Digite el Criterio de Parada: .0001

A=1.000 B=2.000 C= 2.000 fA=-25.000 fB=121.000 fC=121.000 Er= 100.000
A=1.500 B=1.500 C= 2.000 fA=-67.570 fB=-67.570 fC=121.000 Er= 33.333
A=1.750 B=1.750 C= 2.000 fA=-33.876 fB=-33.876 fC=121.000 Er= 14.286
A=1.750 B=1.875 C= 1.875 fA=-33.876 fB=22.294 fC=22.294 Er= 6.667
A=1.813 B=1.813 C= 1.875 fA=-10.224 fB=-10.224 fC=22.294 Er= 3.448
A=1.813 B=1.844 C= 1.844 fA=-10.224 fB=4.824 fC=4.824 Er= 1.695
A=1.828 B=1.828 C= 1.844 fA=-2.989 fB=-2.989 fC=4.824 Er= 0.855
A=1.828 B=1.836 C= 1.836 fA=-2.989 fB=0.844 fC=0.844 Er= 0.426
A=1.832 B=1.832 C= 1.836 fA=-1.091 fB=-1.091 fC=0.844 Er= 0.213
A=1.834 B=1.834 C= 1.836 fA=-0.128 fB=-0.128 fC=0.844 Er= 0.106
A=1.834 B=1.835 C= 1.835 fA=-0.128 fB=0.357 fC=0.357 Er= 0.053
A=1.834 B=1.834 C= 1.834 fA=-0.128 fB=0.114 fC=0.114 Er= 0.027
A=1.834 B=1.834 C= 1.834 fA=-0.007 fB=-0.007 fC=0.114 Er= 0.013
A=1.834 B=1.834 C= 1.834 fA=-0.007 fB=0.053 fC=0.053 Er= 0.007
A=1.834 B=1.834 C= 1.834 fA=-0.007 fB=0.023 fC=0.023 Er= 0.003
A=1.834 B=1.834 C= 1.834 fA=-0.007 fB=0.008 fC=0.008 Er= 0.002
A=1.834 B=1.834 C= 1.834 fA=-0.007 fB=0.000 fC=0.000 Er= 0.001
A=1.834 B=1.834 C= 1.834 fA=-0.003 fB=-0.003 fC=0.000 Er= 0.000
A=1.834 B=1.834 C= 1.834 fA=-0.002 fB=-0.002 fC=0.000 Er= 0.000

```
A=1.834   B=1.834   C= 1.834   fA=-0.001   fB=-0.001   fC=0.000   Er= 0.000
A=1.834   B=1.834   C= 1.834   fA=-0.000   fB=-0.000   fC=0.000   Er= 0.000

La Raíz es: 1.834>>
```

Resultado
Bisección acusa una raíz en x= 1.834 en el intervalo 1< x < 3

METODO DE LA SECANTE

```
% METODO DE LA SECANTE
%
clc
syms x
f=inline(input('ingrese la funcion :','s'));
x0=input('ingrese intervalo inferior:');
x1=input('ingrese intervalo superior:');
tol=input('ingrese la tolerancia:');
fx0=feval(f,x0);
fx1=feval(f,x1);
n=1;
if(abs(fx0))<(abs(fx1)) fx2=900; disp('n x0 x1 x2 f(x0) f(x1) f(x2) ')
while (abs(fx2))>tol
x2=x1-((x1-x0)/(fx1-fx0))*fx1;
fx2=feval(f,x2);
fprintf('\n%d %0.4f %0.4f %0.4f %0.4f %0.4f %0.4f',n,x0,x1,x2,fx0,fx1,fx2)
x0=x1;
x1=x2;
fx0=feval(f,x0);
fx1=feval(f,x1);
n=n+1;
end
else
disp('NO SE PUEDE REALIZAR POR ESTE METODO');
disp('|F(X0)| TIENE QUE SER MENOR A |F(X1)|');
end
```

Se realizan algunas iteraciones

```
ingrese la función :1*x.^7+6*x.^6+6*x.^5-21*x.^4-36*x.^3+3*x.^2+13*x+3
ingrese intervalo inferior:-3
ingrese intervalo superior:3
ingrese la tolerancia:.00001
n x0 x1 x2 f(x0) f(x1) f(x2)

1 -3.0000 3.0000 -2.9900 -9.0000 5415.0000 -8.2668
2 3.0000 -2.9900 -2.9809 5415.0000 -8.2668 -7.6344
3 -2.9900 -2.9809 -2.8707 -8.2668 -7.6344 -2.4919
4 -2.9809 -2.8707 -2.8173 -7.6344 -2.4919 -1.2415
5 -2.8707 -2.8173 -2.7642 -2.4919 -1.2415 -0.5215
6 -2.8173 -2.7642 -2.7258 -1.2415 -0.5215 -0.2312
7 -2.7642 -2.7258 -2.6952 -0.5215 -0.2312 -0.0989
8 -2.7258 -2.6952 -2.6723 -0.2312 -0.0989 -0.0421
```

9 -2.6952 -2.6723 -2.6554 -0.0989 -0.0421 -0.0176
10 -2.6723 -2.6554 -2.6432 -0.0421 -0.0176 -0.0073
11 -2.6554 -2.6432 -2.6346 -0.0176 -0.0073 -0.0029
12 -2.6432 -2.6346 -2.6288 -0.0073 -0.0029 -0.0012
13 -2.6346 -2.6288 -2.6249 -0.0029 -0.0012 -0.0005
14 -2.6288 -2.6249 -2.6224 -0.0012 -0.0005 -0.0002
15 -2.6249 -2.6224 -2.6207 -0.0005 -0.0002 -0.0001
16 -2.6224 -2.6207 -2.6197 -0.0002 -0.0001 -0.0000
17 -2.6207 -2.6197 -2.6191 -0.0001 -0.0000 -0.0000
18 -2.6197 -2.6191 -2.6187 -0.0000 -0.0000 -0.0000
19 -2.6191 -2.6187 -2.6184 -0.0000 -0.0000 -0.0000
20 -2.6187 -2.6184 -2.6183 -0.0000 -0.0000 -0.0000

Resultado
Secante acusa una raíz encontrada en x: -2.6187 en el intervalo -3<x<3

ingrese la funcion :1*x.^7+6*x.^6+6*x.^5-21*x.^4-36*x.^3+3*x.^2+13*x+3
ingrese intervalo inferior:-2
ingrese intervalo superior:1
ingrese la tolerancia:.00001
n x0 x1 x2 f(x0) f(x1) f(x2)

1 -2.0000 1.0000 -1.5000 5.0000 -25.0000 11.1328
2 1.0000 -1.5000 -0.7297 -25.0000 11.1328 2.6995
3 -1.5000 -0.7297 -0.4832 11.1328 2.6995 0.2475
4 -0.7297 -0.4832 -0.4583 2.6995 0.2475 0.1411
5 -0.4832 -0.4583 -0.4253 0.2475 0.1411 0.0455
6 -0.4583 -0.4253 -0.4095 0.1411 0.0455 0.0185
7 -0.4253 -0.4095 -0.3988 0.0455 0.0185 0.0069
8 -0.4095 -0.3988 -0.3924 0.0185 0.0069 0.0027
9 -0.3988 -0.3924 -0.3884 0.0069 0.0027 0.0010
10 -0.3924 -0.3884 -0.3860 0.0027 0.0010 0.0004
11 -0.3884 -0.3860 -0.3844 0.0010 0.0004 0.0001
12 -0.3860 -0.3844 -0.3835 0.0004 0.0001 0.0001
13 -0.3844 -0.3835 -0.3829 0.0001 0.0001 0.0000
14 -0.3835 -0.3829 -0.3825 0.0001 0.0000 0.0000
15 -0.3829 -0.3825 -0.3823 0.0000 0.0000 0.0000
16 -0.3825 -0.3823 -0.3822 0.0000 0.0000 0.0000
17 -0.3823 -0.3822 -0.3821 0.0000 0.0000 0.0000

Resultado
Secante acusa una raíz encontrada en x: -0.3823 en el intervalo -2<x<1

ingrese la funcion :1*x.^7+6*x.^6+6*x.^5-21*x.^4-36*x.^3+3*x.^2+13*x+3
ingrese intervalo inferior:-3
ingrese intervalo superior:1
ingrese la tolerancia:.00001
n x0 x1 x2 f(x0) f(x1) f(x2)

1 -3.0000 1.0000 -5.2500 -9.0000 -25.0000 -18952.8030
2 1.0000 -5.2500 1.0083 -25.0000 -18952.8030 -25.8295

3 -5.2500 1.0083 1.0168 -18952.8030 -25.8295 -26.6960
4 1.0083 1.0168 0.7537 -25.8295 -26.6960 -4.9883
5 1.0168 0.7537 0.6932 -26.6960 -4.9883 -1.6844
6 0.7537 0.6932 0.6624 -4.9883 -1.6844 -0.2495
7 0.6932 0.6624 0.6570 -1.6844 -0.2495 -0.0169
8 0.6624 0.6570 0.6566 -0.2495 -0.0169 -0.0002
9 0.6570 0.6566 0.6566 -0.0169 -0.0002 -0.0000>>

Resultado
Secante acusa una raíz encontrada en x: 0.6570 en el intervalo -3<x<1

ingrese la funcion :1*x.^7+6*x.^6+6*x.^5-21*x.^4-36*x.^3+3*x.^2+13*x+3
ingrese intervalo inferior:1.5
ingrese intervalo superior:3
ingrese la tolerancia:.00001
n x0 x1 x2 f(x0) f(x1) f(x2)

1 1.5000 3.0000 1.5185 -67.5703 5415.0000 -67.4296
2 3.0000 1.5185 1.5367 5415.0000 -67.4296 -67.0190
3 1.5185 1.5367 4.5107 -67.4296 -67.0190 87862.8936
4 1.5367 4.5107 1.5390 -67.0190 87862.8936 -66.9480
5 4.5107 1.5390 1.5412 87862.8936 -66.9480 -66.8725
6 1.5390 1.5412 3.5469 -66.9480 -66.8725 17533.8374
7 1.5412 3.5469 1.5489 -66.8725 17533.8374 -66.5845
8 3.5469 1.5489 1.5564 17533.8374 -66.5845 -66.2459
9 1.5489 1.5564 3.0351 -66.5845 -66.2459 5889.2314
10 1.5564 3.0351 1.5729 -66.2459 5889.2314 -65.3188
11 3.0351 1.5729 1.5889 5889.2314 -65.3188 -64.1502
12 1.5729 1.5889 2.4694 -65.3188 -64.1502 1201.8635
13 1.5889 2.4694 1.6335 -64.1502 1201.8635 -59.3828
14 2.4694 1.6335 1.6729 1201.8635 -59.3828 -53.0831
15 1.6335 1.6729 2.0045 -59.3828 -53.0831 125.5432
16 1.6729 2.0045 1.7714 -53.0831 125.5432 -26.6676
17 2.0045 1.7714 1.8123 125.5432 -26.6676 -10.3298
18 1.7714 1.8123 1.8381 -26.6676 -10.3298 1.9206
19 1.8123 1.8381 1.8340 -10.3298 1.9206 -0.1044
20 1.8381 1.8340 1.8342 1.9206 -0.1044 -0.0010
21 1.8340 1.8342 1.8342 -0.1044 -0.0010 0.0000>>

Resultado
Secante acusa una raíz encontrada en x: 1.8340 en el intervalo 1.5<x<3

METODO NEWTON

```
% METODO NEWTON
 % FUNCION POLINOMICA Y SU DERIVADA ( f1 y f2)
 x=0
     f1=1*x.^7+6*x.^6+6*x.^5-21*x.^4-36*x.^3+3*x.^2+13*x+3
     f2=7*x.^6+36*x.^5+30*x.^4-84*x.^3-108*x.^2+6*x+13
     n=0
x = input('enter the starting value x0  ')
    format compact
   disp('  iterate       x           f(x)          est. error  ')
   error1=1;
   while error1 > 0.00002
    n=n+1;
  f1=1*x.^7+6*x.^6+6*x.^5-21*x.^4-36*x.^3+3*x.^2+13*x+3;
 f2=7*x.^6+36*x.^5+30*x.^4-84*x.^3-108*x.^2+6*x+13;
       s=f1/f2;
x1=x-s;
x=x1;
f1=1*x.^7+6*x.^6+6*x.^5-21*x.^4-36*x.^3+3*x.^2+13*x+3;
error1=abs(f1);
sprintf('   %2d    %2.10f    %2.10f    % 2.10f    %2.10f \n', n,x,f1,error1)
x=x+0.0001;
end
```

Se realizan algunas iteraciones

```
x =
     0
f1 =
     3
f2 =
    13
n =
     0
enter the starting value x0  -3
x =
    -3
  iterate       x           f(x)          est. error
ans =
    1    -2.8815789474    -2.8291444678     2.8291444678
ans =
    2    -2.7934631109    -0.8646502419     0.8646502419
ans =
    3    -2.7304183782    -0.2579936750     0.2579936750
ans =
    4    -2.6868908611    -0.0746405343     0.0746405343
ans =
    5    -2.6581051975    -0.0207170446     0.0207170446
ans =
    6    -2.6400425123    -0.0054345648     0.0054345648
ans =
    7    -2.6293676960    -0.0013180147     0.0013180147
ans =
    8    -2.6234228974    -0.0002827872     0.0002827872
ans =
    9    -2.6202710516    -0.0000473605     0.0000473605
ans =
   10    -2.6186560243    -0.0000036078     0.0000036078
```

Resultado
Newton acusa una raíz encontrada en x: -2.6186560243 para valor inicial x= -3

```
x =
     0
f1 =
     3
f2 =
    13
n =
     0
enter the starting value x0  -2
x =
    -2
  iterate       x           f(x)          est. error
ans =
    1   -2.2941176471    0.9194424017    0.9194424017
ans =
    2   -2.3974512957    0.2386161460    0.2386161460
ans =
    3   -2.4518189166    0.0615719212    0.0615719212
ans =
    4   -2.4790588549    0.0135615149    0.0135615149
ans =
    5   -2.4890624796    0.0018089799    0.0018089799
ans =
    6   -2.4907587182    0.0001028620    0.0001028620
ans =
    7   -2.4908466738    0.0000165921    0.0000165921
```

Resultado
Newton acusa una raíz encontrada en x: es: -2.4908466738 para valor inicial x= -2

```
x =
     0
f1 =
     3
f2 =
    13
n =
     0
enter the starting value x0  -1
x =
    -1
  iterate       x           f(x)          est. error
ans =
    1   -0.5625000000    0.7759331726    0.7759331726
ans =
    2   -0.4696513823    0.1861209571    0.1861209571
ans =
    3   -0.4251160453    0.0452054773    0.0452054773
ans =
    4   -0.4030288652    0.0107736032    0.0107736032
ans =
```

```
    5    -0.3919984959     0.0024437440     0.0024437440
ans =
    6    -0.3864828584     0.0004952575     0.0004952575
ans =
    7    -0.3837245563     0.0000750619     0.0000750619
ans =
    8    -0.3823452901     0.0000034914     0.0000034914
```

Resultado
Newton acusa una raíz encontrada en x:: -0.3823452901 para valor inicial x= -1

```
x =
     0
f1 =
     3
f2 =
    13
n =
     0
enter the starting value x0  1.5
x =
   1.500000000000000
  iterate       x           f(x)          est. error
ans =
    1    141.0000000000    1155459036129975.0000000000    1155459036129975.0000000000
ans =
    2    120.7375007090    392756589308197.6900000000    392756589308197.6900000000
ans =
    3    103.3700992795    133502900401110.9700000000    133502900401110.9700000000
ans =
    4    88.4842867339    45379063181938.8830000000    45379063181938.8830000000
ans =
    5    75.7256397582    15424709738033.1910000000    15424709738033.1910000000
ans =
    6    64.7903821317    5242926942809.7969000000    5242926942809.7969000000
ans =
    7    55.4181520707    1782067401509.4050000000    1782067401509.4050000000
ans =
    8    47.3858032922    605710990400.7984600000    605710990400.7984600000
ans =
    9    40.5020922211    205870602487.4356700000    205870602487.4356700000
ans =
   10    34.6031248709    69969054865.7384800000    69969054865.7384800000
ans =
   11    29.5484550275    23779016952.7465320000    23779016952.7465320000
ans =
   12    25.2177408873    8080698797.8600140000    8080698797.8600140000
ans =
   13    21.5078806268    2745734240.1837430000    2745734240.1837430000
ans =
   14    18.3305588218    932835746.0438144200    932835746.0438144200
ans =
   15    15.6101454826    316857968.1728097200    316857968.1728097200
ans =
   16    13.2818979715    107597650.7767946700    107597650.7767946700
ans =
```

```
   17    11.2904234556    36523434.1579994780    36523434.1579994780
ans =
   18    9.5883660454    12390895.7884333660    12390895.7884333660
ans =
   19    8.1352886404    4200466.5392728280    4200466.5392728280
ans =
   20    6.8967250532    1422366.6672068723    1422366.6672068723
ans =
   21    5.8433837281    480870.2135210487    480870.2135210487
ans =
   22    4.9504911141    162184.8291356558    162184.8291356558
ans =
   23    4.1972720730    54502.2563019857    54502.2563019857
ans =
   24    3.5665794618    18209.9110973619    18209.9110973619
ans =
   25    3.0447102039    6025.1280801085    6025.1280801085
ans =
   26    2.6214851992    1958.3280332052    1958.3280332052
ans =
   27    2.2907001506    613.8630369877    613.8630369877
ans =
   28    2.0508242070    176.9272075218    176.9272075218
ans =
   29    1.9039278322    40.8145280175    40.8145280175
ans =
   30    1.8440620178    4.9864661670    4.9864661670
ans =
   31    1.8344756332    0.1153351864    0.1153351864
ans =
   32    1.8342434528    0.0001331620    0.0001331620
ans =
   33    1.8342432088    0.0000121215    0.0000121215
```

Resultado

Newton acusa una raíz encontrada en x: 1.8342432088 para valor inicial x= 1.5

```
x =
     0
f1 =
     3
f2 =
    13
n =
     0
enter the starting value x0  0.5
x =
   0.500000000000000
  iterate       x           f(x)          est. error
ans =
    1    0.7570093458    -5.1901766864    5.1901766864
ans =
    2    0.6714871786    -0.6564633099    0.6564633099
ans =
    3    0.6570474776    -0.0183245332    0.0183245332
ans =
```

```
   4    0.6566209870    -0.0000238348     0.0000238348
ans =
   5    0.6566204513    -0.0000008689     0.0000008689
```

Resultado
Newton acusa una raíz encontrada en x:: 0.6566204513 para valor inicial x= 0.5

TABLA DE RESULTADOS COMPARADA

RESULTADOS

Función roots MATLAB	Método Newton	Bisección	Secante
-2.618033988749875 + 0.000000059704635i	-2.6186560243		-2.6187
-2.618033988749875 - 0.000000059704635i			
-2.490863615361067	-2.4908466738	-2.491	
1.834243184313920	1.8342432088	1.834	1.834
0.656620431047111	0.6566204513	0.657	0.657
-0.381966011250105 + 0.000000003982789i	-0.3823452901		-0.3823
-0.381966011250105 - 0.000000003982789i			

RESULTADOS:
Los algoritmos utilizados arrojan resultados similares de los valores de las raíces del polinomio.
En relación al número de iteraciones, el método de Newton se aproxima más rápido en las primeras iteraciones. Converge más rápido hacia las raíces negativas pero tiende a oscilar en los valores de raíces positivas.
El método de la bisección sirve cuando la curva corta el eje una sola vez. Para su uso debe buscarse el modo para que cruce por el cero una sola vez. Este método es robusto y no requiere del cálculo de la derivada, pero es poco eficiente porque aumenta en número de iteraciones para encontrar la raíz.
El método de la secante también es un robusto pero no eficiente aunque presenta menos iteraciones que el método de la Bisección.
El método de newton es más eficiente que el método de la secante pero no es recomendable a nivel general porque tiene algunos problemas de inestabilidad.
Un método no ensayado en este práctico es el método de Brent.
Combina la confiabilidad del método de bisección con la velocidad del método de la secante.
Presenta tres alternativas: el paso de bisección (lento); el paso de la secante (mediano); el paso interpolación parabólica inversa o "cuadrático inverso" (rápido).
Si la iteración resulta bastante bien, se toma el paso rápido. En el caso de que se esté en el paso de la secante y no mejora, se toma el paso de bisección.
Un inconveniente que presenta el método de Brent es no se aplica si el polinomio presenta coeficientes complejos. En tal caso la solución pasa por formar un sistema de 2 ecuaciones reales y no complejas. Para resolver ecuaciones complejas se utiliza Muller.
Como se ve, el método de Brent es complicado para codificar en un algoritmo eficiente. Este tipo de programa aplica el concepto de adaptatividad lo que significa que se "adapta y ajusta según la experiencia previa" del problema.

Caso Práctico nº 3

Métodos Numéricos 2009

Master Ing. Pablo DE SIMONE

Ing. Roberto HAARTH

Caso Práctico nº 3

✓ Enunciado:

Datos de Población en años y millones de habitantes

Año	Población
1900	76
1910	92
1920	106
1930	123
1940	132
1950	151
1960	179
1970	203
1980	227
1990	250

Calcular la población para el año 1890. Usar Splines como método de interpolación.
Comparar interpolando con un polinomio de grado 9

✓ Teoría:

Interpolación

El procedimiento de interpolación consiste en generar a partir de un conjunto N de puntos, un polinomio P que represente el comportamiento de los puntos

$$\begin{array}{cccccc} X & x_0 & x_1 & \cdots & x_n \\ Y & y_0 & y_1 & \cdots & y_n \end{array}$$

$$\downarrow$$

$$P(x) = b_0 + b_1(x - x_0) + b_2(x - x_0)(x - x_1) + \cdots b_n(x - x_0)\cdots(x - x_{n-1})$$

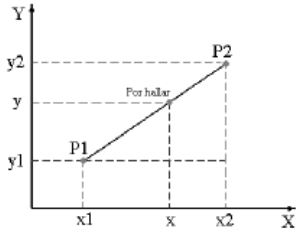
Condiciones de Interpolación

$$P(x_i) = y_i, \quad \forall i = 0,\ldots,n$$

$$\begin{array}{cccccc} X & x_0 & x_1 & \cdots & x_n \\ Y & y_0 & y_1 & \cdots & y_n \end{array} \qquad \begin{array}{l} P(x_0) = y_0 \\ P(x_1) = y_1 \\ \vdots \\ P(x_n) = y_n \end{array}$$

Interpolación Lineal:

Cantidad de puntos necesarios: 2 puntos



$$\begin{array}{cccc} X & x_0 & x_1 \\ Y & y_0 & y_1 \end{array}$$

Estructura General Polinomio Lineal:

$$P(x) = b_0 + b_1(x - x_0)$$

Dado que $P(x_0) = y_0$, entonces:

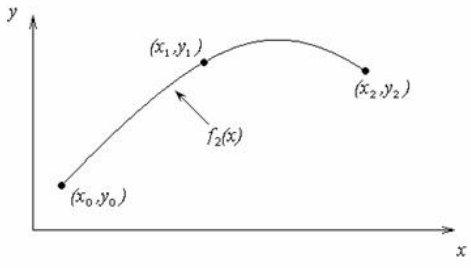
$$P(x_0) = b_0 + b_1(x_0 - x_0) = b_0 = y_0$$

$$P(x_1) = y_0 + b_1(x_1 - x_0) = y_1$$

$$\therefore \quad b_1 = \frac{y_1 - y_0}{x_1 - x_0}$$

Interpolación Cuadrática:

Cantidad de puntos necesarios: 3 puntos



$$\begin{array}{cccc} X & x_0 & x_1 & x_2 \\ Y & y_0 & y_1 & y_2 \end{array}$$

$$P(x) = b_0 + b_1(x - x_0) + b_2(x - x_0)(x - x_1)$$

$$P(x_2) = b_0 + b_1(x_2 - x_0) + b_2(x_2 - x_0)(x_2 - x_1) = y_2$$

$$b_0 = y_0$$

$$b_1 = \frac{y_1 - y_0}{x_1 - x_0} \qquad \therefore b_2 = \frac{\frac{y_2 - y_1}{x_2 - x_1} - \frac{y_1 - y_0}{x_1 - x_0}}{x_2 - x_0}$$

Diferencias divididas de Newton:

El objetivo es simplificar las operaciones

$$f[x_i, x_j] = \frac{f[x_i] - f[x_j]}{x_i - x_j}$$

Caso lineal:

$$\begin{array}{l} x_0 \to f[x_o] = y_o \\ x_1 \to f[x_1] = y_1 \end{array} \quad \overset{b_0}{\searrow} \quad \overset{b_1}{f[x_1, x_0] = \frac{f[x_1] - f[x_0]}{x_1 - x_0}}$$

$$f(x) = b_0 + b_1(x - x_0) = f[x_0] + f[x_1, x_0](x - x_0)$$

Caso cuadrático:

$$\begin{array}{l} x_0 \to f[x_o] = y_o \\ x_1 \to f[x_1] = y_1 \\ x_2 \to f[x_2] = y_2 \end{array} \quad \begin{array}{l} \overset{b_0}{\searrow} \\ \overset{b_0}{\searrow} \end{array} \quad \begin{array}{l} \overset{b_1}{f[x_1, x_0] = \frac{f[x_1] - f[x_0]}{x_1 - x_0}} \\ f[x_2, x_1] = \frac{f[x_2] - f[x_1]}{x_2 - x_1} \end{array} \quad \overset{b_2}{\searrow} \quad f[x_2, x_1, x_0] = \frac{f[x_2, x_1] - f[x_1, x_0]}{x_2 - x_0}$$

$$\begin{aligned} f(x) &= b_0 + b_1(x - x_0) + b_2(x - x_0)(x - x_1) \\ &= f[x_0] + f[x_1, x_0](x - x_0) + f[x_2, x_1, x_0](x - x_0)(x - x_1) \end{aligned}$$

Polinomio de Interpolación de Lagrange:

Es un método alternativo

$$\begin{array}{cccccc} X & x_0 & x_1 & \cdots & x_n \\ Y & y_0 & y_1 & \cdots & y_n \end{array}$$

$$P(x) = y_0 l_0(x) + y_1 l_1(x) + \cdots + y_n l_n(x)$$

$$\begin{aligned} &P(x_0) = y_0 \Rightarrow l_0(x_0) = 1, \quad l_0(x_i) = 0 \ \forall i \neq 0 \\ &P(x_1) = y_1 \Rightarrow l_1(x_1) = 1, \quad l_1(x_i) = 0 \ \forall i \neq 1 \\ &\vdots \\ &P(x_n) = y_n \Rightarrow l_n(x_n) = 1, \quad l_n(x_i) = 0 \ \forall i \neq n \end{aligned}$$

Despejando $l_0(x_0) = 1$ y $l_0(x_i) = 0$, $i \neq 0$, entonces se propone:

$$l_0(x) = c(x - x_1)(x - x_2)\cdots(x - x_n)$$

Como $l_0(x_0) = 1$, entonces:

$$1 = l_0(x_0) = c(x_0 - x_1)(x_0 - x_2)\cdots(x_0 - x_n)$$

$$\Rightarrow c = \frac{1}{(x_0 - x_1)(x_0 - x_1)\cdots(x_0 - x_n)}$$

$$\therefore \quad l_0(x) = \frac{(x - x_1)(x - x_2)\cdots(x - x_n)}{(x_0 - x_1)(x_0 - x_2)\cdots(x_0 - x_n)}$$

Generalización de Lagrange:

$$l_j(x) = \frac{\prod_{\forall i \neq j, i=0,\ldots,n} (x - x_i)}{\prod_{\forall i \neq j, i=0,\ldots,n} (x_j - x_i)}, \quad j = 0,\ldots,n$$

Interpolación por Splines:

En este método en lugar de representar todos los puntos por un polinomio de alto grado, se divide en segmentos a los puntos, cada segmento es un polinomio de menor grado.
Una función spline s(x) está formada por varios polinomios, cada uno definido en un intervalo, cada uno de los cuales se une bajo ciertas condiciones de continuidad:

$x_0 < x_1 < \ldots < x_n$

X	x_0	x_1	$\cdots$	x_n
Y	y_0	y_1	$\cdots$	y_n

a) $s(x_i) = y_i$, $\forall\ i = 0, \ldots, n$
b) $s(x)$ es un polinomio de grado $\leq k$ en cada subintervalo $[x_{i-1}, x_i]$
c) $s(x)$ tiene derivada continua hasta de orden k-1

Splines Lineales: Los puntos se unen con segmentos de rectas.

Splines cuadráticos: Los puntos se unen con segmentos de curvas cuadráticas.

Splines cúbicos: Los puntos de unen con segmentos de curvas cúbicas.

Splines cúbico:

Este procedimiento consiste en representar el fenómeno estudiado a través de polinomios de tercer grado. Ello se realiza tomando pares de puntos, e imponiendo condiciones de **suavidad**, esto es, restricciones a las derivadas de primer y segundo grado.

Consideremos dos puntos consecutivos: (x_i, y_i) y (x_{i+1}, y_{i+1}), y el polinomio cúbico:

$$p_i(x) = a_i \cdot (x - x_i)^3 + b_i \cdot (x - x_i)^2 + c_i \cdot (x - x_i) + d_i$$

.

A continuación debemos imponer las condiciones antes señaladas:

1. Ambos puntos $((x_i, y_i)$ y $(x_{i+1}, y_{i+1}))$ pertenecen a $p_i(x)$.

   Para (x_i, y_i):

$$p_i(x_i) = a_i \cdot (x_i - x_i)^3 + b_i \cdot (x_i - x_i)^2 + c_i \cdot (x_i - x_i) + d_i = d_i = y_i \tag{1}$$

   Para (x_{i+1}, y_{i+1}):

$$p_i(x_{i+1}) = a_i h_i^3 + b_i h_i^2 + c_i h_i + d_i = y_{i+1} \tag{2}$$

   donde $h_i = x_{i+1} - x_i$.

2. La segunda derivada de dos aproxiamciones distintas debe ser iguales en el extremo común.

   La primera derivada:

$$p_i'(x) = 3a_i(x - x_i)^2 + 2b_i(x - x_i) + c_i \tag{3}$$

   La segunda derivada:

$$p_i''(x) = 6a_i(x - x_i) + 2b_i \tag{4}$$

   Definiendo $p''(x_i) = S_i$ y $p''(x_{i+1}) = S_{i+1}$ y reemplazando en (4) en x_i y x_{i+1}, entonces

   Si $x = x_i$:

$$S_i = 6a_i(x_i - x_i) + 2b_i = 2b_i \tag{5}$$

   Si $x = x_{i+1}$:

$$S_{i+1} = 6a_i(x_{i+1} - x_i) + 2b_i = 6a_i h_i + 2b_i \tag{6}$$

   Reordenando las ecuaciones (5) y (6) se obtiene:

$$b_i = \frac{S_i}{2} \tag{7}$$

$$a_i = \frac{S_{i+1} - 2b_i}{6h_i} = \frac{S_{i+1} - S_i}{6h_i} \tag{8}$$

   Si reemplazamos las ecuaciones (1), (7) y (8) en (2) se llega a:

$$y_{i+1} = \frac{S_{i+1} - S_i}{6h_i} h_i^3 + \frac{S_i}{2} h_i^2 + c_i h_i + y_i$$

con lo cual:

$$\begin{aligned} c_i &= \frac{y_{i+1}-y_i}{h_i} - \frac{S_{i+1}-S_i}{6}h_i - \frac{S_i h_i}{2} \\ &= \frac{y_{i+1}-y_i}{h_i} - \frac{S_{i+1}+2S_i}{6}h_i \end{aligned} \tag{9}$$

3. Ahora impondremos continuidad para la primera derivada:

$$p'(x_{i-1}) = p'(x_i) \tag{10}$$

Evaluando (3) en x_{i-1} se obtiene:

$$\begin{aligned} p'(x_{i-1}) &= 3a_{i-1}(x_i - x_{i-1})^2 + 2b_{i-1}(x_i - x_{i-1}) + c_{i-1} \\ &= 3a_{i-1}h_{i-1}^2 + 2b_{i-1}h_{i-1} + c_{i-1} \end{aligned} \tag{11}$$

y, además, la evalúo en x_i

$$\begin{aligned} p'(x_i) &= 3a_i(x_i - x_i)^2 + 2b_i(x_i - x_i) + c_i \\ &= c_i \end{aligned} \tag{12}$$

De la ecuación (10), con reemplazos de (11) y (12) ...

$$3a_{i-1}h_{i-1}^2 + 2b_{i-1}h_{i-1} + c_{i-1} = c_i \tag{13}$$

A continuación reemplazamos (7), (8) y (9) en la expresión (13) (para i y $i-1$)

A continuación reemplazamos (7), (8) y (9) en la expresión (13) (para i y $i-1$)

$$3\frac{S_{i+1}-S_i}{6h_i}h_{i-1}^2 + 2\frac{S_i}{2}h_{i-1} + \left(\frac{y_i - y_{i-1}}{h_{i-1}} - \frac{S_i + 2S_{i-1}}{6}h_{i-1}\right) = \frac{y_{i+1}-y_i}{h_i} - \frac{S_{i+1}+2S_i}{6}h_i$$

Reordenando la última expresión se concluye:

$$h_{i-1}S_{i-1} + 2(h_{i-1}+h_i)S_i + h_i S_{i+1} = 6\left(\Delta_i - \Delta_{i-1}\right) \tag{14}$$

con $i = 1, \ldots, n-1$. Se ha definido: $\Delta_i = \frac{y_{i+1}-y_i}{h_i}$.

Así se define un sistema de $n-1$ ecuaciones para $n+1$ incógnitas (los S_i). En general, para resolver el sistema se deben imponer condiciónes externas. Existen dos posibilidades:

- Condiciones sobre la segunda derivada: $S_0 = A$ y $S_n = B$.
  Existe un caso particular que consiste en considerar $S_0 = S_n = 0$ y se le denomina como **Spline Natural**.
- Condiciones sobre la primera derivada: $p_0'(x_0) = A$ y $p_n'(x_n) = B$. Con lo cual se agregan dos ecuaciones:

$$\begin{aligned} 2h_0 S_0 + h_1 S_1 &= 6(\Delta_0 - A) \\ h_{n-2}S_{n-1} + 2h_{n-1}S_n &= 6(B - \Delta_{n-1}) \end{aligned}$$

y llegamos a tener $n+1$ ecuaciones para $n+1$ incógnitas.

✓ Resolución del Caso Práctico nº 3:

- **INTERPOLACION CON SPLINES**

Datos de Población en años y millones de habitantes:

Año	**Población**
1900	76
1910	92
1920	106
1930	123
1940	132
1950	151
1960	179
1970	203
1980	227
1990	250

N= 10 X es el vector correspondiente a los años
Y es el vector correspondiente a la población

x=[1900,1910,1920,1930,1940,1950,1960,1970,1980,1990]
y=[76,92,106,123,132,151,179,203,227,250]

El grafico que resulta de representar los datos es:

```
% Matlab grafica  Población vs años
x=[1900,1910,1920,1930,1940,1950,1960,1970,1980,1990]
y=[76,92,106,123,132,151,179,203,227,250]
plot (x,y,'b')
title('Datos de Población años 1900-1990');
ylabel('Millones de habitantes');
grid on
```

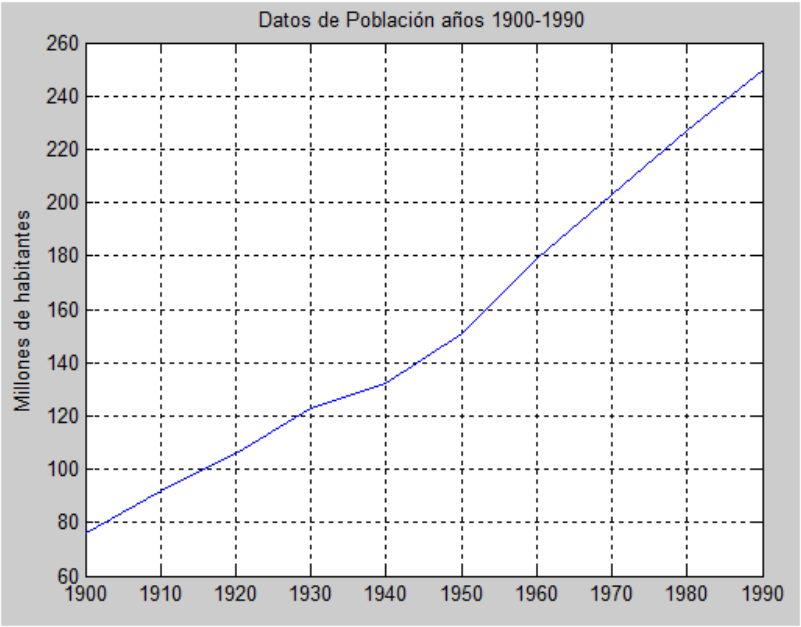


Gráfico en función de los datos
x=[1900,1910,1920,1930,1940,1950,1960,1970,1980,1990]
y=[76,92,106,123,132,151,179,203,227,250]

Se llama $g_i(x)$ a los polinomios de grado 3 que interpola en cada intervalo entre 2 puntos consecutivos.

Para resolver se asume que las derivadas segundas del polinomio interpolador son nulas en los extremos del intervalo, entonces se determinan las constantes del mismo según:

$$g_i(x) = a_i\,(x - x_i)^3 + b_i\,(x - x_i)^2 + c_i\,(x - x_i) + d_i$$

Cálculo de los vectores auxiliares:

i= 0…..(Nmax -2)

$$h_i = x_{i+1} - x_i$$

$$\Delta_i = \frac{y_{i+1} - y_i}{h_i}$$

Matriz H para i= 0….(Nmax -3):

$$\begin{pmatrix} 2h_0 + 2h_1 & h_1 & 0 & 0 & 0 & 0 & 0 & 0 \\ h_1 & 2h_1 + 2h_2 & h_2 & 0 & 0 & 0 & 0 & 0 \\ 0 & h_2 & 2h_2 + h_3 & h_3 & 0 & 0 & 0 & 0 \\ 0 & 0 & h_3 & 2h_3 + h_4 & h_4 & 0 & 0 & 0 \\ 0 & 0 & 0 & h_4 & 2h_4 + h_5 & h_5 & 0 & 0 \\ 0 & 0 & 0 & 0 & h_5 & 2h_5 + h_6 & h_7 & 0 \\ 0 & 0 & 0 & 0 & 0 & h_6 & 2h_6 + h_7 & h_7 \\ 0 & 0 & 0 & 0 & 0 & 0 & h_7 & 2h_7 + h_8 \end{pmatrix}$$

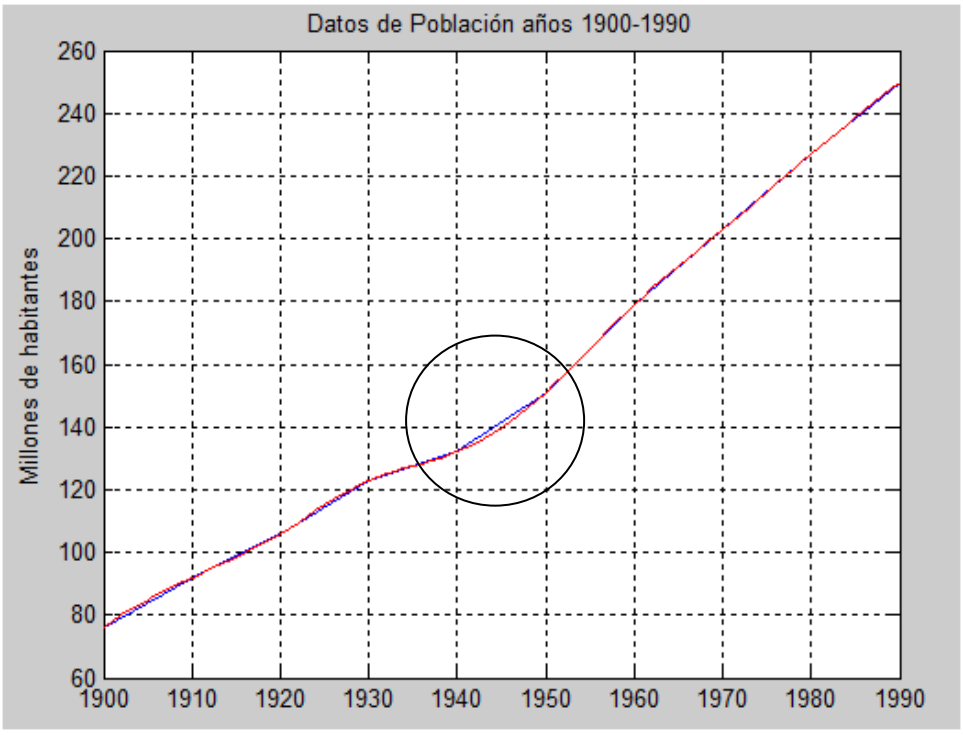
Se determinan los coeficientes a_i b_i c_i d_i ecuaciones (7) (8) (9)

Resolviendo se forman las polinomios para cada intervalo entre puntos adyacentes
Resultando para el caso 8 ecuaciones porque los extremos inicial y final se consideran 0

$$g_i(x) = a_i\,(x - x_i)^3 + b_i\,(x - x_i)^2 + c_i\,(x - x_i) + d_i$$

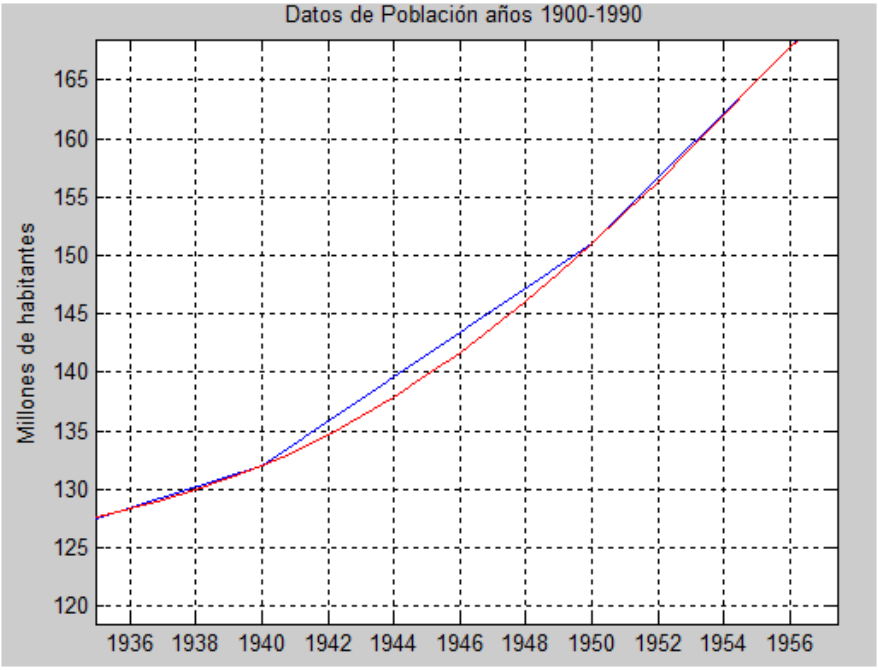
Gráfico comparado entre Splines y la gráfica original

```
% Matlab grafica  Población vs años
x=[1900,1910,1920,1930,1940,1950,1960,1970,1980,1990]
y=[76,92,106,123,132,151,179,203,227,250]
plot (x,y,'b')
title('Datos de Población años 1900-1990');
ylabel('Millones de habitantes');
grid on
axis ([1890 2000 50 250])
n=10;
t = 1:n;
ts = 1:1/10:n;
xs = spline(t,x,ts);
ys = spline(t,y,ts);
hold on
plot(xs,ys,'r');
% plot(xs,ys,'*r');
hold off
%axis ([1890 2000 50 300])
```

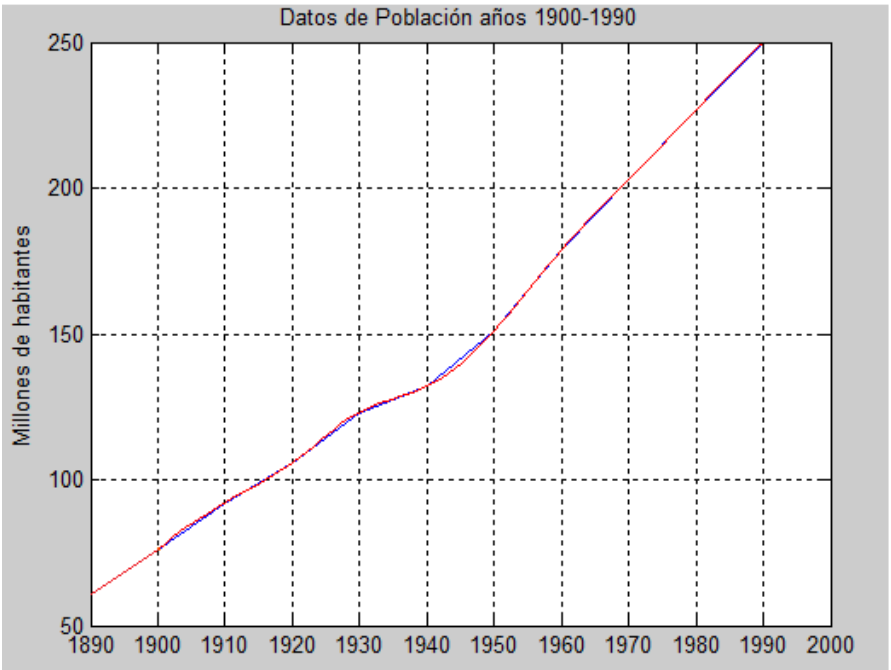


Gráfica obtenida con el polinomio interpolador splines y la gráfica original.

Se observa una zona interesante (ver zona del círculo) correspondiente a los años 1940 y 1950. Graficando en esta zona se aprecia la diferencia del polinomio interpolador.



Resultado:
La línea superior es la recta correspondiente a los datos entre 1940 y 1950.
La línea inferior curva corresponde a la interpolación splines cúbico entre los años 1940 y 1950.
Se aprecia que splines cúbico “suaviza” la curva por las restricciones impuestas a las derivadas de primer y segundo orden.



Resultado: Año 1890 Población: 60 millones

- **INTERPOLACION CON UN POLINIOMIO DE GRADO n=9**

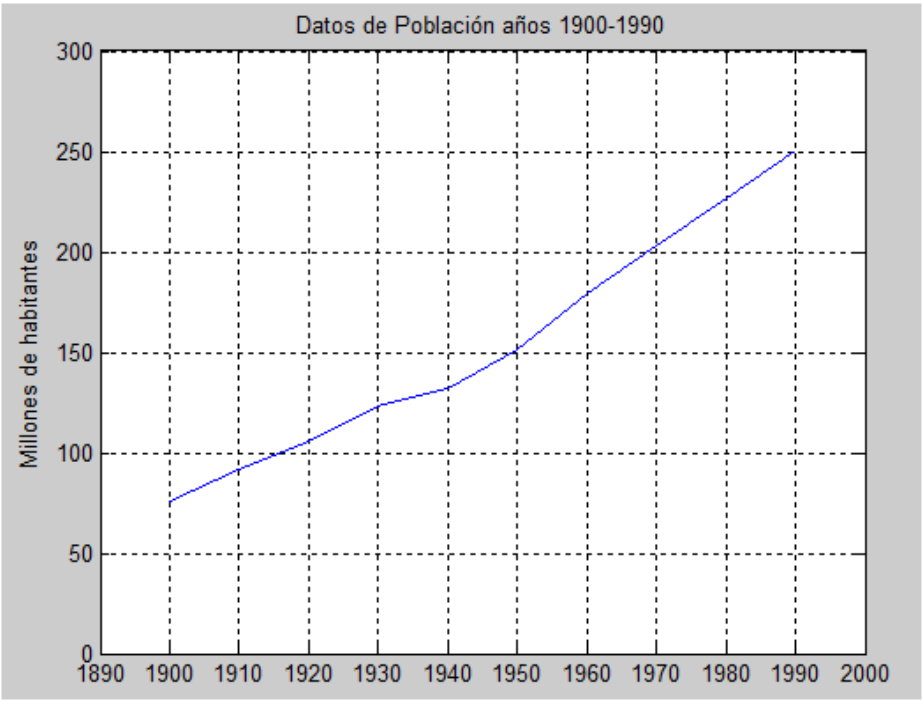
Datos de Población en años y millones de habitantes:

Año	**Población**
1900	76
1910	92
1920	106
1930	123
1940	132
1950	151
1960	179
1970	203
1980	227
1990	250

N= 10 X es el vector correspondiente a los años
Y es el vector correspondiente a la población
x=[1900,1910,1920,1930,1940,1950,1960,1970,1980,1990]
y=[76,92,106,123,132,151,179,203,227,250]

Gráfica de la función dato:

```
% Matlab grafica  Población vs años
x=[1900,1910,1920,1930,1940,1950,1960,1970,1980,1990]
y=[76,92,106,123,132,151,179,203,227,250]
plot (x,y,'b')
title('Datos de Población años 1900-1990');
ylabel('Millones de habitantes');
grid on
axis ([1890 2000 0 300])
```



Para interpolar con un polinomio de grado n, se considera el caso general:

Interpolación y Aproximación de Funciones

Problema básico de Interpolación: Dados los datos (x_i,y_i) , 1<=i<=n, queremos hallar una función g(x) tal que:

$$g(x_i)=y_i \text{ , } 1<=i<=n.$$

Problema de Interpolación Polinomial: Dados los datos (x_i,y_i) , 1<=i<=n, queremos hallar un polinomio $p_{n-1}(x)$ **de grado a lo más n-1**, tal que:

$$p_{n-1}(x_i)=y_i \text{ , } 1<=i<=n.$$

Esto generaliza al caso general como sigue. Escribimos $p_{n-1}(x)=a_1+a_2x+...+a_nx^{n-1}$.
Ahora:

$$y_i=p_{n-1}(x_i)=a_1+a_2 x_i+...+a_nx_i^{n-1} \text{ , } 1<=i<=n.$$

Esto es equivalente al sistema:

$$\begin{pmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^{n-1} \\ 1 & x_2 & x_2^2 & \cdots & x_2^{n-1} \\ 1 & x_3 & x_3^2 & \cdots & x_3^{n-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^{n-1} \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \\ a_3 \\ \vdots \\ a_n \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_n \end{pmatrix}$$

La matriz de coeficientes de este sistema se conoce como la *matriz de Vandermonde* y se puede demostrar que es no singular si los x_i son todos distintos.
Esta condición se asume de aquí en adelante.
De modo que el polinomio de interpolación $p_{n-1}(x)$ existe por construcción. La unicidad de $p_{n-1}(x)$ se verifica usando el Teorema Fundamental del Algebra. De hecho, si q(x) es otro polinomio de grado n-1 que interpola a los datos, entonces $p_{n-1}(x)$-q(x) es polinomio de grado n-1 con n raices, los x_i . Esto es imposible a menos que q(x)= $p_{n-1}(x)$.

Para calcular la matriz de Vandermonde V, primero observamos que si j>1, la columna j de V se obtiene multiplicando (componente a componente) el vector columna $(x_1,\ldots,x_n)$ con la columna j-1 de V:

$$\begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} .* \begin{bmatrix} v_{1,j-1} \\ \vdots \\ v_{n,j-1} \end{bmatrix} = \begin{bmatrix} v_{1,j} \\ \vdots \\ v_{n,j} \end{bmatrix}$$

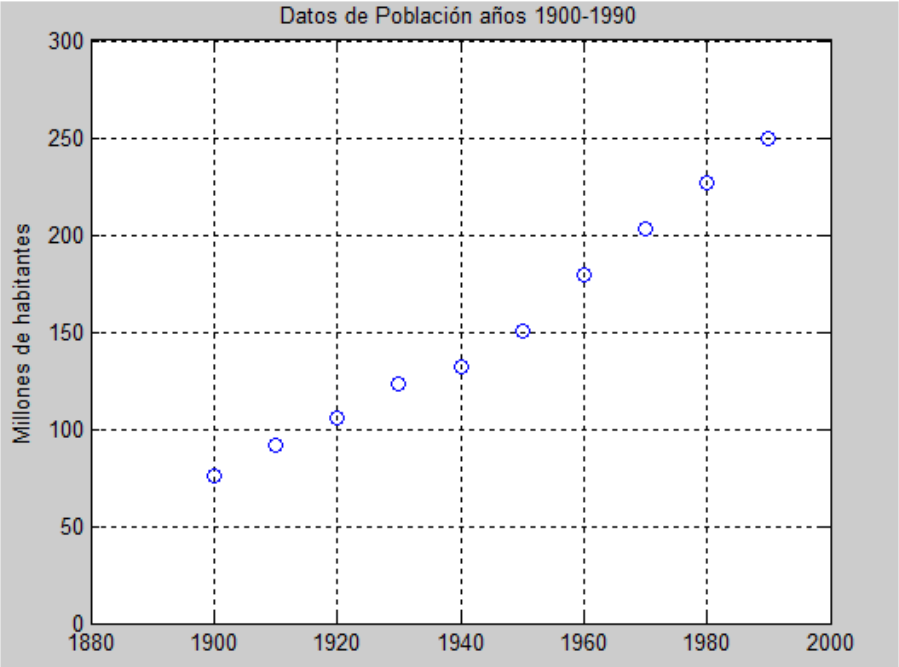
lo cual se puede escribir (en código MATLAB) como V(:,j)=x.*V(:,j-1), donde $x=[x_1\ x_2\ \ldots\ x_n]$.

El polinomio $p_{n-1}(x)$ se puede evaluar ahora en su forma anidada: $a_1+x(a_2+...(a_{n-2}+x(a_{n-1}+a_nx))\ldots)$

Supongamos un valor “z”. Dado un valor de z para evaluar $p_{n-1}(z)$.
Se generaliza al caso en que z es un vector columna de valores en el que tenemos que evaluar el polinomio.

Desarrollo

```
%Modelo de Matlab  para los años 1890 -1900
% Matlab grafica  Población vs años
% Intervalo de Tiempo en años
t = (1900:10:1990)'
% Población en millones
p=[76.0,92.0,106.0,123.0,132.0,151.0,179.0,203.0,227.0,250.0,282.0];
plot(t,p,'bo');
axis([1880 2000 0 300]);
title('Datos de Población años 1900-1990');
ylabel('Millones de habitantes');
grid on
```



Cálculo de la matriz Vandermonde. Determinación de los coeficientes.
Vamos a ajustar los datos con un polinomio en t que se utilizará para extrapolar a t = 1890.
Los coeficientes del polinomio se obtienen al resolver un sistema lineal de ecuaciones con la matriz Vandermonde A (i, j) = s (i) ^ (n-j);

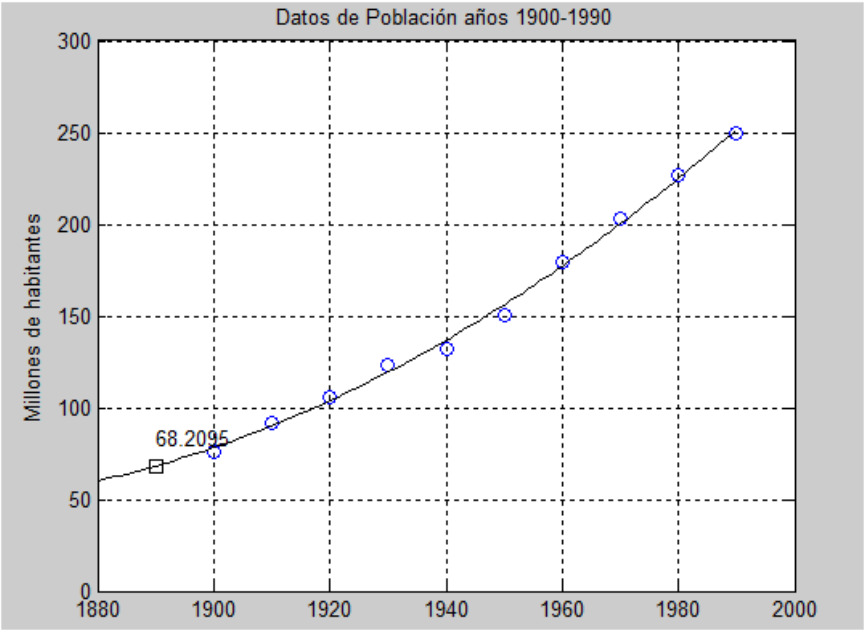
```
% Matriz de Vandermonde y coeficientes  c
n = length(t)
s = (t-1950)/50
A = zeros(n);
A(:,end) = 1;
for j = n-1:-1:1, A(:,j) = s .* A(:,j+1); end

c = A(:,n-2:n)\p
```

Valores obtenidos

```
s =
  -1.000000000000000
  -0.800000000000000
  -0.600000000000000
  -0.400000000000000
  -0.200000000000000
                   0
   0.200000000000000
   0.400000000000000
   0.600000000000000
   0.800000000000000
c =
  1.0e+002 *
   0.231777462121212
   1.009799734848485
   1.560095636363637
```

Con estos datos evaluamos el polinomio para cada año considerando el valor solicitado:

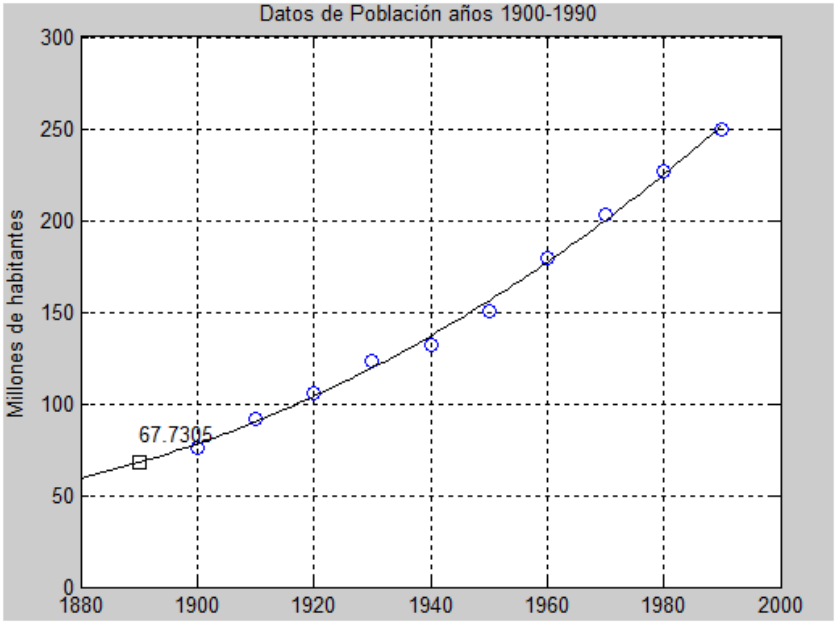
```
%Modelo de Matlab  para los años 1890 -1900
% Polinomio resultante y valor extrapolado para el año 1890
v = (1880:1990)';
x = (v-1950)/50;
w = (1890-1950)/50;
y = polyval(c,x);
z = polyval(c,w);
%
hold on
plot(v,y,'k-');
plot(1890,z,'ks');
text(1890,z+15,num2str(z));
hold off
```



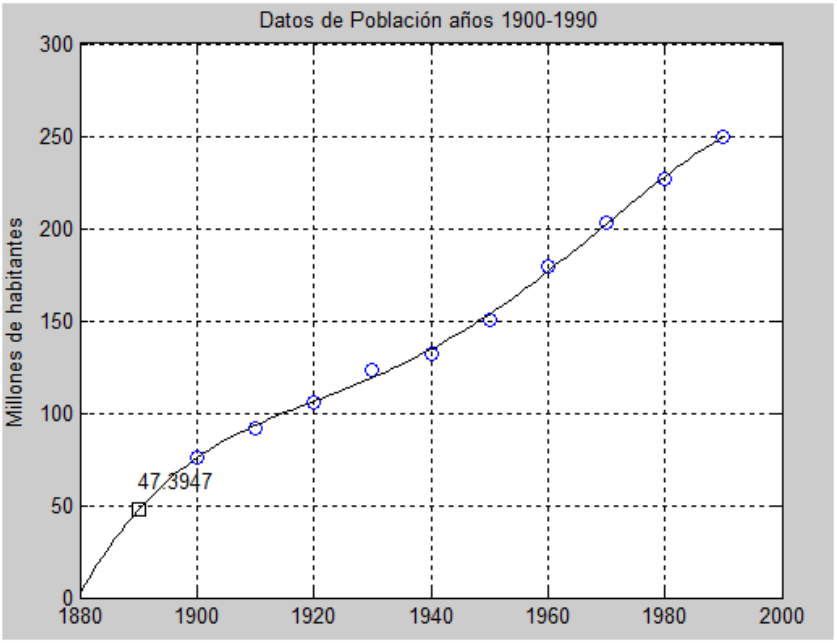
Resultado: Año 1890 Población: 68.2095 millones

Ajuste cúbico y cuadrático:
Se pueden determinar los coeficientes para un ajuste cúbico y cuadrático.
Para esto hacemos:

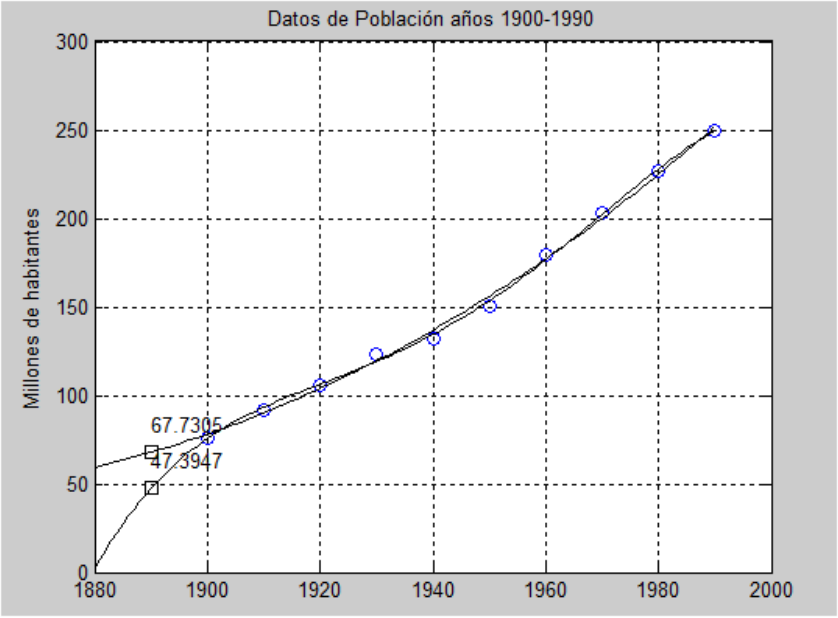
```
% ajuste cúbico
c = A(:,n-3:n)\p
v = (1880:1990)';
x = (v-1950)/50;
w = (1890-1950)/50;
y = polyval(c,x);
z = polyval(c,w);
%
hold on
plot(v,y,'k-');
plot(1890,z,'ks');
text(1890,z+15,num2str(z));
hold off
```




```
% ajuste cuadrático
c = A(:,n-5:n)\p
v = (1880:1990)';
x = (v-1950)/50;
w = (1890-1950)/50;
y = polyval(c,x);
z = polyval(c,w);
%
hold on
plot(v,y,'k-');
plot(1890,z,'ks');
text(1890,z+15,num2str(z));
hold off
```



Resultado comparativo de las gráficas

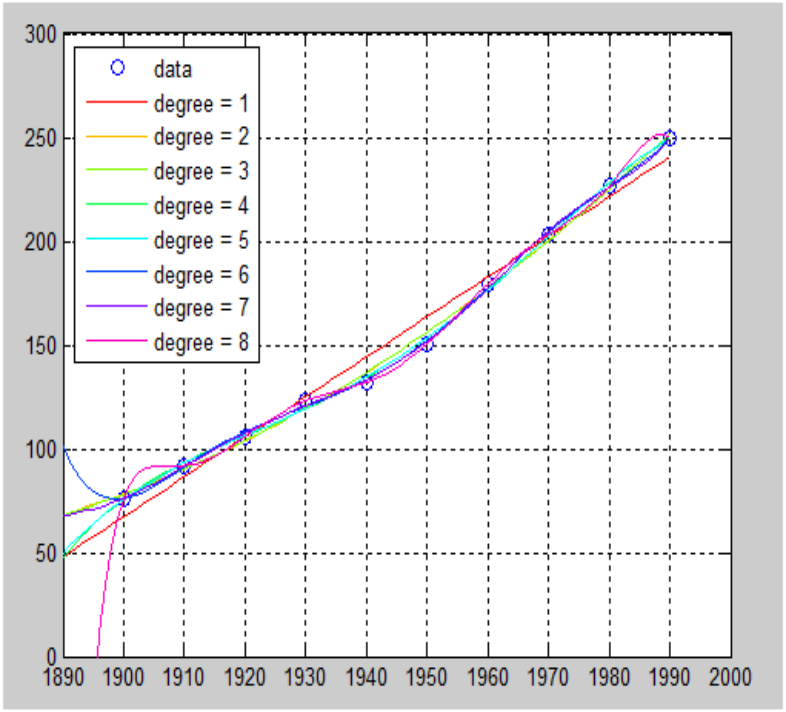


Resultados año 1890:
67.7305 millones
47.3947 millones

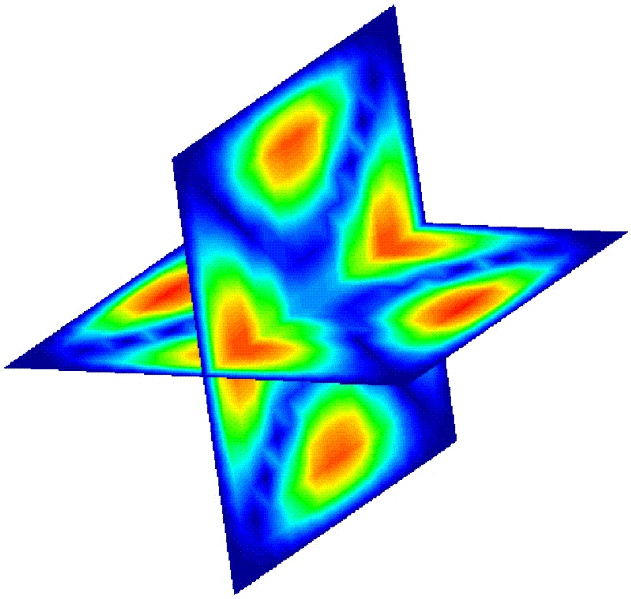
Observación:

A medida que se modifica el grado se observa un comportamiento más errático.

```
%Modelo de Matlab
%Modelo de Matlab  para los años 1890
-1900
cla
plot(t,p,'bo'); hold on;
axis([1890 2000 0 300]);
colors = hsv(8); labels = {'data'};
for d = 1:8
  [Q,R] = qr(A(:,n-d:n));
  R = R(1:d+1,:); Q = Q(:,1:d+1);
  c = R\(Q'*p);
% Same as c = A(:,n-d:n)\p;
  y = polyval(c,x);
  z = polyval(c,11);
  plot(v,y,'color',colors(d,:));
  labels{end+1} = ['degree = ' int2str(d)];
end
legend(labels,2)
```



Caso Práctico nº 4



Métodos Numéricos 2009

Master Ing. Pablo DE SIMONE

Ing. Roberto HAARTH

Caso Práctico nº 4

✓ Enunciado:

Evaluar la integral mediante métodos de cuadratura numérica.

$$\int_0^{\infty} \frac{logx}{x^2 + 1}\, dx$$

✓ Teoría:

Integración Numérica

Cuadratura Numérica

El principio de los métodos de integración numérica, basados en las fórmulas de Newton-Cotes, consiste en ajustar una un polinomio a un conjunto de puntos y luego integrarlo. Al realizar dichas integrales obtenemos, entre otras, las reglas de trapecio y de Simpson 1/3 las cuales dan lugar a reglas de integración compuestas que buscan que el error sea cada vez menor.

Newton - Cotes: Estimación de la integral, se basa sobre valores de la función uniformemente espaciados. La localización de estos puntos fue fijo.

Regla del Trapecio

La regla del trapecio está dada por:

$$\int_a^b f(x)\, dx \simeq \frac{b-a}{2}\left[f(a) + f(b)\right]$$

si queremos programar esta regla sólo debemos tener en cuenta que los datos de entrada son a, b, f y el dato de salida es la aproximación

Regla del trapecio compuesta

Este es un ejemplo de regla compuesta.

$$\int_a^b f(x)dx \sim \frac{b-a}{n}\left(\frac{f(a)+f(b)}{2}+\sum_{k=1}^{n-1} f\left(a+k\frac{b-a}{n}\right)\right)$$

Donde $a+kh$ son los subintervalos

siendo $h=\frac{b-a}{n}$

y $k=0,1,2,\ldots,n-1$.

Regla Trapezoidal: Debe pasar a través de los puntos extremos. Existen casos donde la fórmula resulta en un error grande.
Supóngase que la restricción de los puntos fijos es eliminada y se tiene la libertad de evaluar el área bajo una recta que conecta dos puntos cualesquiera sobre la curva. Al ubicar estos puntos en forma "inteligente", se puede definir una línea recta que equilibre los errores negativos y positivos. El objetivo de la cuadratura de Gauss es determinar los coeficientes de una ecuación de la forma:

$$I \approx c_0 f(x_0)+c_1 f(x_1)$$

En la cual las *c* son coeficientes desconocidos. En contraste con la regla trapezoidal que usa los puntos extremos fijos *a* y *b*, los argumentos de la función x_0 y x_1 no están fijos en los puntos extremos.

Se tienen cuatro incógnitas que deben ser evaluadas y se requieren cuatro condiciones para determinarlas con exactitud. Se pueden obtener dos de esas condiciones al suponer que la ecuación ajusta la integral de una constante y de una función lineal con exactitud. Para tener las otras dos condiciones, se extiende este razonamiento al suponer que también ajusta la integral de una función parabólica ($y = x_2$) y de una cúbica ($y = x_3$). Las cuatro funciones a resolverse son:

$$c_0 f(x_0)+c_1 f(x_1)=\int_{-1}^{1} 1\,dx = 2$$

$$c_0 f(x_0)+c_1 f(x_1)=\int_{-1}^{1} x\,dx = 0$$

$$c_0 f(x_0)+c_1 f(x_1)=\int_{-1}^{1} x^2\,dx = \frac{2}{3}$$

$$c_0 f(x_0)+c_1 f(x_1)=\int_{-1}^{1} x^3\,dx = 0$$

Resolviendo simultáneamente:

$$c_0 = c_1 = 1$$
$$x_0 = -\frac{1}{\sqrt{3}} = -0.5773503...$$
$$x_0 = \frac{1}{\sqrt{3}} = 0.5773503...$$

Sustituyendo en la ecuación de coeficientes para obtener la ecuación de *Gauss–Legendre* de dos puntos:

$$I \simeq f\left(-\frac{1}{\sqrt{3}}\right) + f\left(\frac{1}{\sqrt{3}}\right)$$

Se llega a un resultado interesante en el que la simple suma de los valores de la función en $x = 1/\sqrt{3}$ y $-x = 1/\sqrt{3}$ dan una estimación de la integral que tiene una exactitud de tercer orden. Obsérvese que los límites de integración de las ecuaciones son de -1 a 1. Esto se hizo para simplificar las matemáticas y para hacer la formulación tan general como sea posible. Es posible usar un cambio de variable para trasladar cualquier límite a esta forma. Suponiendo que una nueva variable x_d se relaciona con la variable original x en una forma lineal, como en:

$$x = a_0 + a_1 x_d$$

Si el límite inferior $x = a$, corresponde a $x_d = -1$, estos valores podrán sustituirse en la ecuación anterior para dar:

$$a = a_0 + a_1(-1)$$

De manera similar, el límite superior $x = b$, corresponde a $x_d = 1$, para dar:

$$b = a_0 + a_1(1)$$

Resolviendo estas ecuaciones simultáneamente:

$$a_0 = \frac{b+a}{2} \qquad a_1 = \frac{b-a}{2}$$

Las cuales se pueden sustituir en la ecuación de relación para obtener:

$$x = \frac{(b+a)+(b-a)x_d}{2}$$

Esta ecuación puede diferenciarse para dar:

$$dx = \frac{b-a}{2} dx_d$$

Las dos ecuaciones anteriores podrán sustituirse para x y dx, respectivamente, en la evaluación que se habrá de integrar. Esas sustituciones transforman el intervalo de integración sin cambiar el valor de la integral.

Regla de Simpson 1/3 simple y compuesta

Tenemos que las reglas de Simpson 1/3 simple y compuesta están dadas por:

$$\int_a^b f(x)\,dx \simeq \frac{b-a}{6}\left[f(a)+4f\left(\frac{a+b}{2}\right)+f(b)\right]$$

$$\int_a^b f(x)\,dx \simeq \frac{h}{3}\left[f(a)+4\sum_{i=1}^{\frac{n}{2}} f(x_{2i-1})+2\sum_{i=1}^{\frac{n}{2}-1} f(x_{2i})+f(b)\right]$$

donde h = (b – a)/n , para n un entero par mayor igual a 2 y xi = a + ih. Los límites de las sumatorias y los subindices 2i − 1 y 2i indican que xj con j par se evalua en f y se multiplica por dos, y si j es impar se evalua en f y se multiplica por 4. (esto se obtiene al aplicar regla de Simpson 1/3 simple sobre los subintervalos [x0, x2] , [x2, x4] , ..., [xn−2, xn]).

Cuadratura de Gauss

Las reglas de Newton–Cotes garantizan la integración exacta de polinomios hasta un determinado grado: el error en las reglas impares (trapecio, tres octavos,...) depende de la derivada de un orden superior (segunda, cuarta,...) y por tanto la regla integra exactamente polinomios hasta el mismo grado que indica la regla (primero, tercero,...); el error en las reglas pares (punto medio, Simpson, Boole,...) depende de la derivada de orden dos unidades por encima de la regla (segunda, cuarta, sexta,...) y por tanto integra exactamente polinomios polinomios hasta un grado por encima del orden de la regla (primero, tercero, quinto,...).
A la vista de este comportamiento vamos a buscar reglas de integración numérica de la forma $\sum w_i f(x_i)$ que integren todos los polinomios hasta un determinado grado, el mayor posible para el número de sumandos elegidos para la regla. Obtendremos así las llamadas reglas Gaussianas o cuadraturas de Gauss.
Si utilizamos un sólo nodo la regla resultante integrará exactamente polinomios de grado menor o igual a uno: (normalizamos el cálculo al intervalo [−1, 1]). Sea x0 el nodo y w0 el peso correspondiente,

$$\int_{-1}^{1} dx = w_0, \qquad \int_{-1}^{1} x\,dx = w_0 x_0.$$

De aquí resulta x0 = 0, w0 = 2.

Si utilizamos dos nodos podemos integrar exactamente todos los polinomios de grado menor o igual que tres:

$$I_2 f = w_1 f(x_1) + w_2 f(x_2)$$

$$\begin{aligned} I_2(1) &= w_1 + w_2 &&= 2 \\ I_2(x) &= w_1 x_1 + w_2 x_2 &&= 0 \\ I_2(x^2) &= w_1 x_1^2 + w_2 x_2^2 &&= \tfrac{2}{3} \\ I_2(x^3) &= w_1 x_1^3 + w_2 x_2^3 &&= 0 \end{aligned}$$

Este sistema no lineal resulta tener solución única. Ésta verifica:

$$w_1 = w_2 = 1 \text{ y } x_1 = -x_2 = -\sqrt{3}/3.$$

Así:

$$I_2(f) = f(-\frac{\sqrt{3}}{3}) + f(\frac{\sqrt{3}}{3}).$$

En general y por el mismo procedimiento, utilizando n nodos podemos integrar exactamente todos los polinomios de grado _ 2n − 1. Para obtener los nodos y los pesos de In debemos resolver el sistema no lineal

$$\begin{aligned} 2 &= w_1 + \cdots + w_n \\ 0 &= w_1 x_1 + \cdots + w_n x_n \\ \tfrac{2}{3} &= w_1 x_1^2 + \cdots + w_n x_n^2 \\ \cdots &\quad \cdots \quad \cdots \quad \cdots \\ \tfrac{2}{2n-1} &= w_1 x_1^{2n-2} + \cdots + w_n x_n^{2n-2} \\ 0 &= w_1 x_1^{2n-1} + \cdots + w_n x_n^{2n-1} \end{aligned}$$

La resolución de este sistema no lineal no es en general sencilla, de hecho no está claro a priori que el sistema tenga solución o que, en caso de existir, ésta sea única.
Abordamos el problema de su resolución por otros medios. Supongamos primeramente que existen los xi y que son distintos y están en el intervalo [0, 1]. El polinomio Ln(x) = (x − x1)(x − x2) · · · (x − xn) debe tener integral exacta por medio de la regla de Gauss y por tanto esta debe ser cero, ya que Ln(xi) = 0.
Si Ln(x) = xn + a1xn−1 + · · · + an−1x + an podemos determinar los coeficientes a_i por medio del sistema lineal de ecuaciones:

$$0 = \int_{-1}^{1} x^k L_n(x) dx \qquad k = 1, \ldots, n.$$

De hecho se obtienen dos sistemas lineales separados, uno que contiene solamente a los coeficientes de índice par, y el otro solamente a los de índice impar. Uno de ellos será homogéneo con solución trivial, por tanto el polinomio obtenido tendrá solamente términos cuyo grado tenga la misma paridad que n. Los ceros de este polinomio son los nodos xi buscados. En general se obtendrán por algún método de resolución de ecuaciones no lineales (Newton,...). Los polinomios Ln reciben el nombre de polinomios mónicos de Legendre y han sido estudiados extensivamente.

Una vez calculados (aproximados) los nodos, podemos volver al sistema 21.1, que resultará un sistema lineal en las wi con un exceso de ecuaciones. También podemos recurrir al siguiente método para calcular los pesos. Para calcular el peso wi consideramos el polinomio:

$$p_i(x) = \frac{L_n(x)}{x - x_i}$$

es decir, le quitamos al polinomio Ln el factor x − xi. La integral de este polinomio debe ser exacta por la regla de Gauss (su grado es menor que 2n − 1) y por tanto:

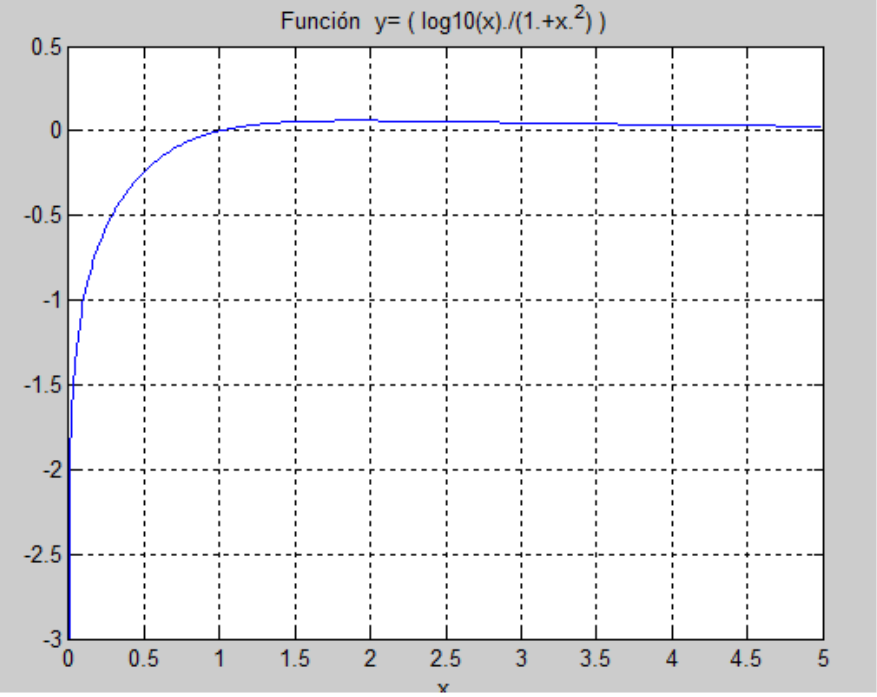
$$w_i p_i(x_i) = \int_{-1}^{1} p_i(x)dx.$$

Los dos cálculos anteriores resultan más sencillos si se observa que los nodos son simétricos respecto de cero y por tanto los pesos para dos nodos simétricos son iguales.

✓ Resolución del Caso Práctico nº 4:

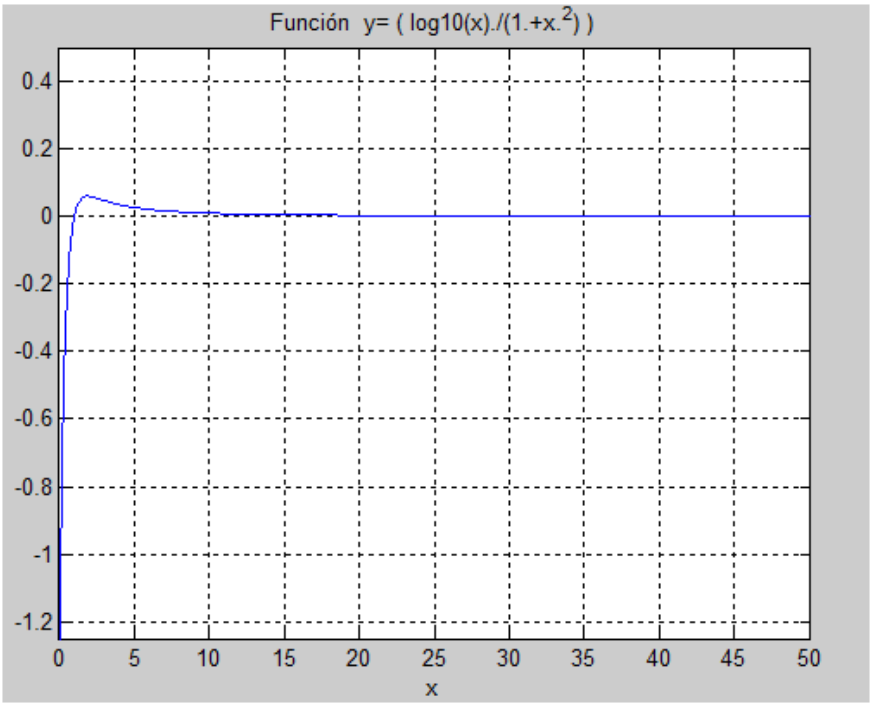
$$\int_0^{\infty} \frac{\log x}{x^2 + 1}\, dx$$

```
%Matlab Gráfico ecuación integral
cla
x = (0.001:0.01:5)'
%formula  2
y= ( log10(x)./(1.+x.^2) ) ;
plot (x,y);
title(' Función  y= ( log10(x)./(1.+x.^2) ) ');
xlabel('  x  ');
grid on
```



Esta integral presenta una singularidad cuando x=0 y se observa que converge lentamente haciéndose asintótica el eje x. Esto se observa con claridad si se grafica en un intervalo de x más amplio 0.001< x < 50.

```
%Matlab Gráfico ecuación integral intervalo 0.001< x < 50
cla
x = (0.001:0.01:50)'
%formula  2
y= ( log10(x)./(1.+x.^2) ) ;
plot (x,y);
title(' Función  y= ( log10(x)./(1.+x.^2) ) ');
xlabel('  x  ');
grid on
```



Observación: La función en el intervalo 0.001 < x < 50 se hace asintótica al eje x.
Esto complica la convergencia por lo que se requieren métodos de integración numérica

Métodos de cuadratura numérica

```
% Matlab Integración cuadratura trapecio
%function s=trapecio(f,a,b,n)
F=@(x) (log10(x)./(1.+x.^2));
a=0;
b=10;
n=1000;
for i=1:1:1000;
h=(b-a)/n;
x=linspace(a,b,n+1); % construimos malla
y=feval(f,x);      % evaluamos f en la malla
s=h*(0.5*y(1)+sum(y(2:end-1))+0.5*y(end));        % aplicamos la regla
end
```

```
Integracion cuadratura Newton Cotes
%function s=NewCotCerrada5ptos(f,a,b,n)
F=@(x) (log10(x)./(1.+x.^2));
a=0;
b=10;
n=1000;
n=4*ceil(n/4);                  % n es ahora divisible por 4
h=(b-a)/n;                      % calculamos h
x=linspace(a,b,n+1);                % construimos malla
y=feval(f,x);                   % evaluamos f en la malla
c1=14/45;
c2=64/45;
```

```
c3=8/15;
c4=64/45;
c5=28/45;
s=h*(c1*y(1)+...
   c2*sum(y(2:4:n))+c3*sum(y(3:4:n))...
   +c4*sum(y(4:4:n))+c5*sum(y(5:4:n))...
   +c1*y(n+1));       % aplicamos la regla
 return
```

Integración cuadratura Simpson

```
%function s=simpson(f,a,b,n)
F=@(x) (log10(x)./(1.+x.^2));
a=0;
b=10;
n=1000;
n=2*ceil(n/2);                  % n es ahora par
h=(b-a)/n;                      % calculamos h
x=linspace(a,b,n+1);                % construimos malla
y=feval(f,x);                   % evaluamos f en la malla
s=h/3*(y(1)+...
   4*sum(y(2:2:n))+2*sum(y(3:2:n))+...
   y(n+1));                     % aplicamos la regla
return
```

Cuadratura numérica de MATLAB: Utiliza cuadratura quad quadl y quadv.

Cuadratura Matlab quadl : q = quadl(fun,a,b) approximates the integral of function fun from a to b, to within an error of 10-6 using recursive adaptive Lobatto quadrature. fun is a function handle.

Cuadratura Matlab quad: q = quad(fun,a,b) tries to approximate the integral of function fun from a to b to within an error of 1e-6 using recursive adaptive Simpson quadrature. fun is a function handle.

Cuadratura Matlab quadv: q= quadv(fun,a,b) approximates the integral of the complex array-valued function fun from a to b to within an error of 1.e-6 using recursive adaptive Simpson quadrature. fun is a function handle.

```
%Matlab cuadratura numérica-2009
%Cuadratura Lobatto (quadl)
F=@(x) (log10(x)./(1.+x.^2));
Q = quadl (F,0,1000);
```

```
%Matlab cuadratura numérica-2009
%Cuadratura Simpson (quad)
F=@(x) (log10(x)./(1.+x.^2));
Q = quad (F,0,1000);
```



```
%Matlab cuadratura numérica-2009
%Cuadratura Simpson recursiva adaptativa (quadv)
F=@(x) (log10(x)./(1.+x.^2));
Q = quadv (F,0,1000);
```

Cuadro comparativo de resultados

(Se considera que, en la "práctica" cuando x= 1000000 es una situación que hace x = infinito, extremo superior de la integral).

Intervalo	Lobatto	Simpson	Simpson adaptativo/recursivo
0< x < 1000	-0.00343496816763196	-0.00344225236139347	-0.00344225236139347
0 < x 10000	-0.000443923288955026	-0.000451239875358477	-0.000451239875358477
0 < x 100000	-5.46788679091427e-005	-6.20272285275978e-005	-6.20272285275978e-005
0 < x < 1000000	-6.63641901760877e-006	-1.43131108326681e-005	-1.43131108326681e-005

RESULTADOS

Se prueban varios intervalos hasta llegar al extremo superior x=1000000 considerado prácticamente como el extremo superior infinito. El extremo inferior no puede ser x=0 ya que es una singularidad, se toma como inicio x=0.0001 o un valor inferior sin tocar el extremo cero.

Observando los resultados, al aumentar el intervalo de integración la función es asintótica al eje x, tiende a un valor sobre el eje de y cercano al cero. En el intervalo donde x>10000 los valores de la integral son similares no importando tanto el método numérico que se utilice.

Un análisis no realizado es el tiempo empleado en procesar los datos. Experiencias de otros ejemplos muestran que el tiempo de procesamiento matemático aumenta de forma no lineal en función del intervalo de integración y la longitud de las subdivisiones de cálculo numérico.

Caso Práctico nº 5

Métodos Numéricos 2009

Master Ing. Pablo DE SIMONE

Ing. Roberto HAARTH

Caso Práctico nº 5

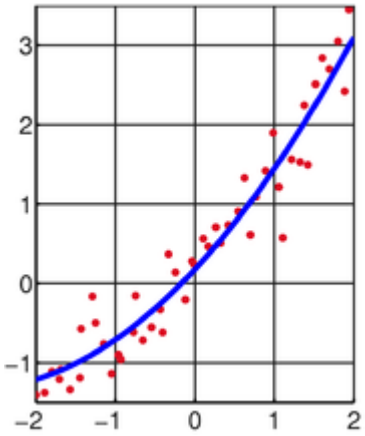
✓ Enunciado:

Calcular la solución por mínimos cuadrados usando Cholesky y después con una transformación ortogonal. Calcular las matrices de covarianza y correlación, σ_1, los errores de la solución. ¿Hay residuos discordantes?

$$A := \begin{pmatrix} 1 & 1 & 1 \\ 1 & 2 & 4 \\ 1 & 3 & 9 \\ 1 & 4 & 16 \\ 1 & 5 & 25 \\ 1 & 6 & 36 \\ 1 & 7 & 49 \\ 1 & 8 & 64 \\ 1 & 9 & 81 \\ 1 & 10 & 100 \end{pmatrix} \qquad d := \begin{pmatrix} 5 \\ 4 \\ 3 \\ 2 \\ 1 \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{pmatrix}$$

✓ Teoría:

Mínimos cuadrados



El resultado del ajuste de un conjunto de datos a una función cuadrática.

Mínimos cuadrados es una técnica de análisis numérico encuadrada dentro de la optimización matemática, en la que, dados un conjunto de pares (o ternas, etc.), se intenta encontrar la función

que mejor se aproxime a los datos (un "mejor ajuste"), de acuerdo con el criterio de mínimo error cuadrático.

En su forma más simple, intenta minimizar la suma de cuadrados de las diferencias ordenadas (llamadas *residuos*) entre los puntos generados por la función y los correspondientes en los datos.

Específicamente, se llama *mínimos cuadrados promedio* (LMS) cuando el número de datos medidos es 1 y se usa el método de descenso por gradiente para minimizar el residuo cuadrado. Se puede demostrar que LMS minimiza el residuo cuadrado esperado, con el mínimo de operaciones (por iteración), pero requiere un gran número de iteraciones para converger.

Desde un punto de vista estadístico, un requisito implícito para que funcione el método de mínimos cuadrados es que los errores de cada medida estén distribuidos de forma aleatoria.

El teorema de Gauss-Márkov prueba que los estimadores mínimos cuadráticos carecen de sesgo y que el muestreo de datos no tiene que ajustarse, por ejemplo, a una distribución normal. También es importante que los datos recogidos estén bien escogidos, para que permitan visibilidad en las variables que han de ser resueltas (para dar más peso a un dato en particular, véase mínimos cuadrados ponderados).

La técnica de mínimos cuadrados se usa comúnmente en el ajuste de curvas. Muchos otros problemas de optimización pueden expresarse también en forma de mínimos cuadrados, minimizando la energía o maximizando la entropía.

Historia

Carl Friedrich Gauss

El día de Año Nuevo de 1801, el astrónomo italiano Giuseppe Piazzi descubrió el asteroide Ceres. Fue capaz de seguir su órbita durante 40 días.

Durante el curso de ese año, muchos científicos intentaron estimar su trayectoria con base en las observaciones de Piazzi (resolver las ecuaciones no lineales de Kepler de movimiento es muy difícil).

La mayoría de evaluaciones fueron inútiles; el único cálculo suficientemente preciso para permitir a Zach, astrónomo alemán, reencontrar a Ceres al final del año fue el de un Carl Friedrich Gauss de 24 años (los fundamentos de su enfoque ya los había planteado en 1795, cuando aún tenía 18 años).

Pero su método de mínimos cuadrados no se publicó hasta 1809, apareciendo en el segundo volumen de su trabajo sobre mecánica celeste, *Theoria Motus Corporum Coelestium in sctionibus conicis solem ambientium*. El francés Adrien-Marie Legendre desarrolló el mismo método de forma independiente en 1805.

En 1829 Gauss fue capaz de establecer la razón del éxito maravilloso de este procedimiento: simplemente, el método de mínimos cuadrados es óptimo en muchos aspectos. El argumento concreto se conoce como teorema de Gauss-Márkov.

Formulación formal del problema bidimensional

Supóngase el conjunto de puntos (x_k,y_k), siendo $k = 1, 2, \ldots, n$. Sea $f_j(x)$, con $j = 1, 2, \ldots, m$una base de m funciones linealmente independientes. Queremos encontrar una función $f$combinación lineal de las funciones base tal que $f(x_k) \approx y_k$, esto es:

$$f(x) = \sum_{j=1}^{m} c_j f_j(x)$$

Se trata de hallar los m coeficientes c_j que hagan que la función aproximante f(x) sea la mejor aproximación a los puntos (x_k,y_k). El criterio de mejor aproximación puede variar, pero en general se basa en aquél que dé un menor error en la aproximación. El error en un punto (x_k,y_k) se podría definir como:

$$e_k = y_k - f(x_k)$$

En este caso se trata de medir y minimizar el error en el conjunto de la aproximación. En matemáticas, existen diversas formas de definir el error, sobre todo cuando éste se aplica a un conjunto de puntos (y no sólo a uno), a una función, etc. Dicho error podrá ser:

Error Máximo: $E_\infty(f) = max(|e_k|)$

Error Medio: $E_m(f) = \frac{\sum_{k=1}^{n} |e_k|}{n}$

Error Cuadrático Medio: $E_{cm}(f) = \sqrt{\frac{\sum_{k=1}^{n} (e_k)^2}{n}}$

La aproximación mínimo cuadrada se basa en la minimización del error cuadrático medio, o, equivalentemente, en la minimización del radicando de dicho error, el llamado error cuadrático, definido como:

$$E_c(f) = \frac{\sum_{k=1}^{n}(e_k)^2}{n}$$

Para alcanzar este objetivo, suponemos que la función *f* es de una forma particular que contenga algunos parámetros que necesitamos determinar. Por ejemplo, supongamos que es cuadrática, lo que quiere decir que $f(x) = ax^2 + bx + c$, donde no conocemos aún a, b y c. Ahora buscamos los valores de a, b y c que minimicen la suma de los cuadrados de los residuos (*S*):

$$S = \sum_{i=1}^{n}(y_i - f(x_i))^2$$

Esto explica el nombre de *mínimos cuadrados*. A las funciones que multiplican a los coeficientes buscados, esto es, a x^2, x y 1, se les conoce con el nombre de funciones base de la aproximación.

Dichas funciones base pueden ser cualesquiera funciones, y para ese caso se deduce a continuación la fórmula general en el caso de que la aproximación sea discreta y lineal.

La aproximación de mínimos cuadrados es la mejor aproximación al conjunto de puntos (x_k,y_k), según el criterio del error mínimo cuadrático.

Es posible generar otro tipo de aproximaciones si se toman los errores máximos o medio, pero la dificultad que entraña operar con ellos debido al valor absoluto de su expresión hace que apenas se usen.

Solución del problema de los mínimos cuadrados

La aproximación mínimo cuadrado tiene solución general para el caso de un problema de aproximación lineal en sus coeficientes c_j cualesquiera sean las funciones base $f_j(x)$ antes expuestas. Por lineal se entiende f(x) es una combinación lineal de dichas funciones base.

Para hallar la expresión de la fórmula general, es posible o bien minimizar el error cuadrático arriba expuesto, para lo cual se haría uso del cálculo multivariable (se trataría de un problema de optimización en c_j), o alternativamente hacer uso del álgebra lineal en la llamada deducción geométrica.

Para los Modelos estáticos uniecuacionales, el método de mínimos cuadrados no ha sido superado, a pesar de diversos intentos para ello, desde principios del Siglo XIX. Se puede demostrar que, en su género, es el que proporciona la mejor aproximación.

Deducción geométrica del problema discreto

La mejor aproximación deberá tender a interpolar la función de la que proviene el conjunto de pares (x_k,y_k), esto es, deberá tender a pasar exactamente por todos los puntos. Eso supone que se debería cumplir que:

$$f(x_k) = y_k \quad \text{con } k = 1, 2, \ldots, n$$

Sustituyendo f(x) por su expresión:

$$\sum_{j=1}^{m} c_j f_j(x_k) = y_k \quad \text{con } k = 1, \ldots, n$$

Esto es, se tendría que verificar exactamente un sistema de n ecuaciones y m incógnitas, pero como en general n>m, dicho sistema está sobre-determinado, no tiene solución general. De ahí surge la necesidad de aproximarlo.

Dicho sistema podría expresarse en forma matricial como:

$$\begin{bmatrix} f_1(x_1) & f_2(x_1) & \ldots & f_m(x_1) \\ f_1(x_2) & f_2(x_2) & \ldots & f_m(x_2) \\ \ldots & \ldots & \ldots & \ldots \\ f_1(x_n) & f_2(x_n) & \ldots & f_m(x_n) \end{bmatrix} \times \begin{bmatrix} c_1 \\ c_2 \\ \ldots \\ c_m \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \ldots \\ y_n \end{bmatrix}$$

Esto es:

$$Ac = b$$

La aproximación trata de hallar el vector c aproximante que mejor aproxime el sistema $Ac = b$.

Con dicho vector c aproximante, es posible definir el vector residuo como:

$$r = b - Ac$$

De manera que el mínimo error cuadrático supone minimizar el residuo, definiendo su tamaño en base a la norma euclídea o usual del residuo, que equivale al error cuadrático:

$$||r||_2 = \sqrt{(r,r)_2} = \sqrt{r^t r} = \sqrt{\sum_{k=1}^{n} (r_k)^2}$$

siendo $(r,r)_2$ el producto interior o escalar del vector residuo sobre sí mismo.

Si atendemos al sistema $Ac = b$, entonces se ve claramente que al multiplicar A y c, lo que se realiza es una combinación lineal de las columnas de A:

$$Ac = \begin{bmatrix} A_1 & A_2 & \dots & A_m \end{bmatrix} \times \begin{bmatrix} c_1 \\ c_2 \\ \dots \\ c_m \end{bmatrix} = c_1 A_1 + c_2 A_2 + \dots + c_m A_m$$

El problema de aproximación será hallar aquella combinación lineal de columnas de A lo más cercana posible al vector b. Se comprueba que el conjunto de las columnas de A engendran un Span lineal: *span* $(A_1,A_2,\dots,A_m)$, al que el vector b no tiene porqué pertenecer (si lo hiciera, el sistema Ac=b tendría solución).

Entonces, de los infinitos vectores del *span* $(A_1,A_2,\dots,A_m)$ que son combinación lineal de los vectores de la base, se tratará de hallar el más cercano al vector b.

De entre todos ellos, el que cumple esto con respecto a la norma euclídea es la proyección ortogonal del b sobre *span* $(A_1,A_2,\dots,A_m)$, y que por tanto hace que el tamaño del vector r, que será el vector que una los extremos de los vectores b y proyección ortogonal de b sobre el span, sea mínimo, esto es, que minimiza su norma euclídea.

Es inmediato ver que si el residuo une b con su proyección ortogonal, entonces es a su vez ortogonal al *span* $(A_1,A_2,\dots,A_m)$, y a cada uno de los vectores de la base, esto es, ortogonal a cada columna de A.

La condición de minimización del residuo será:

$$r \perp span(A_1, A_2 \dots, A_m)$$

Esto solo es cierto si:

$$r \perp A_j \iff A_j \perp r \iff (A_j, r)_2 = 0 = A_j^t r, j = 1, 2, \dots, m$$

A su vez, cada una de las m condiciones de perpendicularidad se puede agrupar en una sola:

$$A^t r = 0$$

Sustituyendo el residuo por su expresión:

$$A^t(b - Ac) = 0 \iff A^tAc = A^tb$$

Por tanto, la mejor aproximación mínimo cuadrada lineal para un conjunto de puntos discretos, sean cuales sean las funciones base, se obtiene al resolver el sistema cuadrado:

$$A^tAc = A^tb.$$

A esta ecuación se le llama ecuación normal de Gauss, y es válida para cualquier conjunto de funciones base. Si estas son la unidad y la función x, entonces la aproximación se llama regresión lineal.

En el ejemplo anterior, *f* es lineal para los parámetros *a*, *b* y *c*. El problema se simplifica considerablemente en este caso y esencialmente se reduce a un sistema lineal de ecuaciones. Esto se explica en el artículo de los mínimos cuadrados lineales.

El problema es más complejo si *f* no es lineal para los parámetros a ser determinados. Entonces necesitamos resolver un problema de optimización general (sin restricciones). Se puede usar cualquier algoritmo para tal problema, como el método de Newton y el descenso por gradiente.

Otra posibilidad es aplicar un algoritmo desarrollado especialmente para tratar con los problemas de mínimos cuadrados, como por ejemplo el algoritmo de Gauss-Newton o el algoritmo de Levenberg-Marquardt.

Mínimos cuadrados y análisis de regresión

En el análisis de regresión, se sustituye la relación

$$f(x_i) \approx y_i$$

por

$$f(x_i) = y_i + \varepsilon_i,$$

siendo el término de perturbación ε una variable aleatoria con media cero.

Obsérvese que estamos asumiendo que los valores *x* son exactos, y que todos los errores están en los valores *y*.

De nuevo, distinguimos entre regresión lineal, en cuyo caso la función *f* es lineal para los parámetros a ser determinados (ej., $f(x) = ax^2 + bx + c$), y regresión no lineal. Como antes, la

regresión lineal es mucho más sencilla que la no lineal. (Es tentador pensar que la razón del nombre *regresión lineal* es que la gráfica de la función $f(x) = ax + b$ es una línea.

Ajustar una curva $f(x) = ax^2 + bx + c$, estimando a, b y c por mínimos cuadrados es un ejemplo de regresión *lineal* porque el vector de estimadores mínimos cuadráticos de a, b y c es una transformación lineal del vector cuyos componentes son $f(x_i) + \varepsilon_i$). Los parámetros (a, b y c en el ejemplo anterior) se estiman con frecuencia mediante mínimos cuadrados: se toman aquellos valores que minimicen la suma S.

El teorema de Gauss-Márkov establece que los estimadores mínimos cuadráticos son óptimos en el sentido de que son los estimadores lineales insesgados de menor varianza, y por tanto de menor error cuadrático medio, si tomamos $f(x) = ax + b$ estando a y b por determinar y con los términos de perturbación ε independientes y distribuidos idénticamente (véase el artículo si desea una explicación más detallada y con condiciones menos restrictivas sobre los términos de perturbación).

La estimación de mínimos cuadrados para modelos lineales es notoria por su falta de robustez frente a valores atípicos (*outliers*). Si la distribución de los atípicos es asimétrica, los estimadores pueden estar sesgados.

En presencia de cualquier valor atípico, los estimadores mínimos cuadráticos son ineficientes y pueden serlo en extremo. Si aparecen valores atípicos en los datos, son más apropiados los métodos de regresión robusta.

Factorización de Cholesky

En matemáticas, la **factorización** o **descomposición de Cholesky** toma su nombre del matemático André-Louis Cholesky, quién encontró que una matriz simétrica definida positiva puede ser descompuesta como el producto de una matriz triangular inferior y la traspuesta de la matriz triangular inferior.

La matriz triangular inferior es el **triángulo de Cholesky** de la matriz original positiva definida. El resultado de Cholesky ha sido extendido a matrices con entradas complejas. Es una manera de resolver sistemas de ecuaciones matriciales y se deriva de la factorización LU con una pequeña variación.

Cualquier matriz cuadrada **A** con pivotes no nulos puede ser escrita como el producto de una matriz triangular inferior **L** y una matriz triangular superior **U**; esto es llamada la factorización LU.

Sin embargo, si **A** es simétrica y definida positiva, se pueden escoger los factores tales que **U** es la transpuesta de **L**, y esto se llama la descomposición o factorización de Cholesky.

Tanto la descomposición LU como la descomposición de Cholesky son usadas para resolver sistemas de ecuaciones lineales. **Cuando es aplicable, la descomposición de Cholesky es dos veces más eficiente como la descomposición LU.**

Definición

En general, si **A** es Hermitiana y definida positiva, entonces **A** puede ser descompuesta como

$$\mathbf{A} = \mathbf{L}\mathbf{L}^*,$$

donde **L** es una matriz triangular inferior con entradas diagonales estrictamente positivas, y $\mathbf{L}^*$ representa la conjugada traspuesta de **L**. Esta es la descomposición de Cholesky.

La descomposición de Cholesky es única: dada una matriz Hermitiana positiva definida **A**, hay una única matriz triangular inferior **L** con entradas diagonales estrictamente positivas tales que **A** = $\mathbf{L}\mathbf{L}^*$. El recíproco se tiene trivialmente: si **A** se puede escribir como $\mathbf{L}\mathbf{L}^*$ para alguna matriz invertible **L**, triangular inferior o no, entonces **A** es Hermitiana y definida positiva.

El requerimiento de que **L** tenga entradas diagonales estrictamente positivas puede extenderse para el caso de la descomposición en el caso de ser semi-definida positiva.

La proposición se lee ahora: una matriz cuadrada **A** tiene una descomposición de Cholesky si y sólo si **A** es Hermitiana y semi-definida positiva. Las factorizaciones de Cholesky para matrices semi-definidas positivas no son únicas en general.

En el caso especial que **A** es una matriz positiva definida simétrica con entradas reales, **L** se puede asumir también con entradas reales. Una Matriz D diagonal con entradas positivas en la diagonal, es factorizable como $D = \sqrt{D}\sqrt{D}$, donde $\sqrt{D}$ es matriz cuya diagonal consiste en la raíz cuadrada de cada elemento de D, que tomamos como positivos. Así:

$$A = LU = LDU_0 = LDL^t = L(\sqrt{D}\sqrt{D})L^t = (L\sqrt{D})(\sqrt{D}L^t) = (L\sqrt{D})(L\sqrt{D})^t = KK^t$$

La factorización puede ser calculada directamente a través de las siguientes fórmulas (en este caso realizamos la factorización superior $A = U^t * U$):

$u_{ii}^2 = a_{ii} - \sum_{k=1}^{i-1} u_{ik}^2$ para los elementos de la diagonal principal, y:

$u_{ij} = \frac{a_{ij} - \sum_{k=1}^{i-1} u_{ik} u_{jk}}{u_{jj}}$ para el resto de los elementos. Donde u_{ij} son los elementos de la matriz U.

Aplicaciones

La descomposición de Cholesky se usa principalmente para hallar la solución numérica de ecuaciones lineales **Ax** = **b**. Si **A** es simétrica y positiva definida, entonces se puede solucionar **Ax** = **b** calculando primero la descomposición de Cholesky **A** = **LL**^{T}, luego resolviendo **Ly** = **b** para **y**, y finalmente resolviendo **L**^{T}**x** = **y** para **x**.

Mínimos cuadrados lineales

Sistemas de la forma **Ax** = **b** con **A** simétrica y definida positiva aparecen a menudo en la práctica. Por ejemplo, las ecuaciones normales en problemas de mínimos cuadrados lineales son problemas de esta forma. Podría ocurrir que la matriz **A** proviene de un funcional de energía el cual debe ser positivo bajo consideraciones físicas; esto ocurre frecuentemente en la solución numérica de ecuaciones diferenciales parciales.

Simulación de Monte Carlo

La descomposición de Cholesky se usa comúnmente en el método de Monte Carlo para simular sistemas con variables múltiples correlacionadas: La matriz de correlación intravariables es descompuesta, para obtener la triangular inferior **L**. Aplicando ésta a un vector de ruidos simulados incorrelacionados, **u** produce un vector **Lu** con las propiedades de covarianza del sistema a ser modelado.

Filtro de Kalman

Los filtros de Kalman usan frecuentemente la descomposición de Cholesky para escoger un conjunto de puntos sigma. El filtro de Kalman sigue el estado promedio de un sistema como un vector **x** de longitud **N** y covarianza dada por una matriz **P** de tamaño *N*-by-*N*. La matriz **P** es siempre positiva semidefinida y puede descomponerse como **LL**^{T}. Las columnas de **L** pueden ser adicionadas y restadas de la media **x** para formar un conjunto de 2*N* vectores llamados los *puntos sigma*. Estos puntos sigma capturan la media y la covarianza del estado del sistema.

Transformación de Householder

En matemáticas, una **transformación de Householder** es una transformación lineal del espacio que consiste en una reflexión pura con respecto a un plano. Viene definida por una matriz $\mathbf{H}$ de dimensión (N x N) tal que para cualquier vector $\mathbf{x}$ de dimensión N se cumple que $y = \mathbf{H}x$ es la reflexión de $\mathbf{x}$ respecto a un plano π.

Estas matrices (matrices de Householder) son **ortogonales** (sus vectores forman una base ortonormal) y son **simétricas**. Como consecuencia son iguales a su propia inversa.

$$\mathbf{H}\mathbf{H} = \mathbf{I}$$
$$\mathbf{H}(\mathbf{H}x) = x$$

Esta propiedad es fácil de comprender si, acudiendo al sentido geométrico de la transformación, decimos que el reflejo del reflejo es el espacio original.

El cálculo de la matriz $\mathbf{H}$ asociada a un plano de reflexión π se hace a partir del vector $\mathbf{v}$ normal al plano según:

$$H = I - \frac{2\mathbf{v}\mathbf{v}^T}{\mathbf{v}^T\mathbf{v}}.$$

Se puede comprobar que multiplicar un vector $\mathbf{x}$ por la expresión anterior equivale a restarle el doble de su proyección sobre el vector $\mathbf{v}$; de donde resulta la reflexión.

Usos de la transformación de Householder

Las matrices de transformación de Householder tienen varias propiedades que hacen que su uso en algoritmos matemáticos sea muy ventajoso. En concreto, el ser iguales a su propia inversa ahorra numerosos cálculos por no tener que invertirlas.

El hecho de ser ortogonales las hace idóneas para el cálculo de matrices semejantes. Por último, el tener un único autovalor (de multiplicidad N) hace que tengan buena estabilidad numérica, pues su número de condición es la unidad.

Estas propiedades hacen que la transformación de Householder sea una de las herramientas más sencillas y utilizadas en el cálculo de matrices semejantes con forma de Hessenberg y en la descomposición QR de una matriz, ambos de gran uso e importancia en el cálculo de autovalores.

Matriz de Covarianza

En estadística y teoría de la probabilidad, la **matriz de covarianza** es una matriz que contiene la covarianza entre los elementos de un vector.

Es la generalización natural a dimensiones superiores del concepto de varianza de una variable aleatoria escalar.

Definición

Si las entradas del vector-columna

$$X = \begin{bmatrix} X_1 \\ \vdots \\ X_n \end{bmatrix}$$

son variables aleatorias, cada una con varianza finita, entonces la matriz de covarianza Σ es la matriz cuya entrada (*i*, *j*) es la covarianza

$$\Sigma_{ij} = \mathrm{E}\big[(X_i - \mu_i)(X_j - \mu_j)\big]$$

donde

$$\mu_i = \mathrm{E}(X_i)$$

es el valor esperado de la entrada *i*-ésima del vector *X*. En otras palabras, tenemos

$$\Sigma = \begin{bmatrix} \mathrm{E}[(X_1 - \mu_1)(X_1 - \mu_1)] & \mathrm{E}[(X_1 - \mu_1)(X_2 - \mu_2)] & \cdots & \mathrm{E}[(X_1 - \mu_1)(X_n - \mu_n)] \\ \mathrm{E}[(X_2 - \mu_2)(X_1 - \mu_1)] & \mathrm{E}[(X_2 - \mu_2)(X_2 - \mu_2)] & \cdots & \mathrm{E}[(X_2 - \mu_2)(X_n - \mu_n)] \\ \vdots & \vdots & \ddots & \vdots \\ \mathrm{E}[(X_n - \mu_n)(X_1 - \mu_1)] & \mathrm{E}[(X_n - \mu_n)(X_2 - \mu_2)] & \cdots & \mathrm{E}[(X_n - \mu_n)(X_n - \mu_n)] \end{bmatrix}.$$

Como una generalización de la varianza

La anterior definición es equivalente a la igualdad matricial

$$\Sigma = \mathrm{E}\left[(\mathbf{X} - \mathrm{E}[\mathbf{X}])\,(\mathbf{X} - \mathrm{E}[\mathbf{X}])^{\top}\right]$$

Por tanto, se entiende que esto generaliza a mayores dimensiones el concepto de varianza de una variable aleatoria escalar *X*, definida como

$$\sigma^2 = \operatorname{var}(X) = \mathrm{E}[(X - \mu)^2],$$

donde

$$\mu = \mathrm{E}(X).$$

Propiedades

Para $\Sigma = \mathrm{E}\left[(\mathbf{X} - \mathrm{E}[\mathbf{X}])\,(\mathbf{X} - \mathrm{E}[\mathbf{X}])^{\top}\right]$ y $\mu = \mathrm{E}(\mathbf{X})$, las siguientes propiedades fundamentales se demuestran correctas:

1. $\Sigma = \mathrm{E}(\mathbf{X}\mathbf{X}^{\top}) - \mu\mu^{\top}$
2. $\boldsymbol{\Sigma}$ es semidefinida positiva
3. $\operatorname{var}(\mathbf{A}\mathbf{X} + \mathbf{a}) = \mathbf{A}\operatorname{var}(\mathbf{X})\,\mathbf{A}^{\top}$
4. $\operatorname{cov}(\mathbf{X}, \mathbf{Y}) = \operatorname{cov}(\mathbf{Y}, \mathbf{X})^{\top}$
5. $\operatorname{cov}(\mathbf{X_1} + \mathbf{X_2}, \mathbf{Y}) = \operatorname{cov}(\mathbf{X_1}, \mathbf{Y}) + \operatorname{cov}(\mathbf{X_2}, \mathbf{Y})$
6. Si *p* = *q*, entonces

   $$\operatorname{var}(\mathbf{X} + \mathbf{Y}) = \operatorname{var}(\mathbf{X}) + \operatorname{cov}(\mathbf{X}, \mathbf{Y}) + \operatorname{cov}(\mathbf{Y}, \mathbf{X}) + \operatorname{var}(\mathbf{Y})$$

7. $\operatorname{cov}(\mathbf{A}\mathbf{X}, \mathbf{B}\mathbf{Y}) = \mathbf{A}\operatorname{cov}(\mathbf{X}, \mathbf{Y})\,\mathbf{B}^{\top}$
8. Si $\mathbf{X}$ y $\mathbf{Y}$ son independientes, entonces $\operatorname{cov}(\mathbf{X}, \mathbf{Y}) = 0$

donde $\mathbf{X}, \mathbf{X_1}$ y $\mathbf{X_2}$ son vectores aleatorios de dimensión $(\mathbf{p} \times \mathbf{1})$, $\mathbf{Y}$ es un vector aleatorio $(\mathbf{q} \times \mathbf{1})$, $\mathbf{a}$ es $(\mathbf{p} \times \mathbf{1})$, $\mathbf{A}$ y $\mathbf{B}$ son matrices de $(\mathbf{p} \times \mathbf{q})$.

La matriz de covarianza (aunque muy simple) es una herramienta muy útil en varios campos.

A partir de ella se puede derivar una transformación lineal que puede *de-correlacionar* los datos o, desde otro punto de vista, encontrar una base óptima para representar los datos de forma óptima (véase cociente de Rayleigh para la prueba formal y otras propiedades de las matrices de covarianza).

Esto se llama análisis del componente principal (PCA por sus siglas en inglés) en estadística , y transformada de Karhunen-Loève en procesamiento de la imagen.

Matriz de Correlación

Una **matriz de correlación** es una tabla de doble entrada para A B y C, que muestra una lista multivariable horizontalmente y la misma lista verticalmente y con el correspondiente coeficiente de correlación llamado *r'*.

El análisis factorial se puede utilizar para estudiar series numéricas o de valores cuantitativos para un determinado número de variables cuantitativas y mayor de dos. Por ejemplo, tres características o más para series numéricas con igual número de datos.

Definición

Estas variables independientes o explicativas están dispuestas ya en una **matriz de correlación**, que es una tabla de doble entrada para A B y C, que muestra una lista multivariable horizontalmente y la misma lista verticalmente y con el correspondiente coeficiente de correlación llamado *r* o la relación entre cada pareja en cada celda, expresada con un número que va desde 0 a 1. El modelo mide y muestra la interdependencia en relaciones asociadas o entre cada pareja de variables y todas al mismo tiempo.

Ejemplo

Se han aplicado los resultados de una correlación de datos entre tres variables

Variables	A	B	C
A			
B	0,3		
C	0,75	0,95	

La mejor relación es B C o C B y desde .95 ya es alta. La diagonal de -*unos*- no tiene obviamente significado, únicamente forma una línea divisoria entre valores que se repiten a ambos lados como en un espejo.

Los coeficientes lineales, tal como se encuentran las parejas de datos en las series, forman un cuadrado en la tabla o matriz de correlación, los calculamos con un programa de estadística para ordenador, que tenga una capacidad de utilizar 8 o más variables para series de 500 o más datos cada una y que empleara esta fórmula.

r es igual a la suma de los productos de cada pareja de datos y dividido por el producto del número de datos por la desviación estándar de cada variable o serie de datos.

Aplicaciones

Para hacer más potente el análisis factorial los programas estadísticos incluyen otros análisis multivariables como es el análisis de pautas o camino, *path analysis*, y otros coeficientes de correlación como es el de rangos o la correspondencia en el orden entre cada pareja en las series y se designa por ro.

Puede utilizarse esta técnica para muchos propósitos como es la Escala de actitud o Prioridades sociales. También un análisis diacrónico de triangulación o varias aplicaciones en sucesivos períodos de tiempo y con diferentes métodos para cada aplicación.

La representación gráfica de la matriz de correlación es una línea recta diagonal en los ejes cartesianos en los que las abscisas son las variables y los coeficientes son una nube de puntos.

El test, que está empleando un coeficiente de correlación o asociación, no es inferencial o predictor, ya que es no-paramétrico o libre de probabilidad, y es descriptivo, no causal. Un test del nivel significativo de los coeficientes de correlación valida la prueba.

Las tablas de asociación 2 x 2 es el caso más elemental o simple de variables dicotomizadas, que igualmente miden o describen la significación estadística. A veces las representaciones gráficas son más descriptivas de la asociación entre variables.

Varianza

En teoría de probabilidad, la **varianza** de una variable aleatoria es la esperanza del cuadrado de la desviación de dicha variable respecto a su media. Se trata de una medida de la dispersión de dicha variable aleatoria.

Está medida en unidades distintas de las de la variable. Por ejemplo, si la variable mide una distancia en metros, la varianza se expresa en metros al cuadrado. La desviación estándar, la raíz

cuadrada de la varianza, está sin embargo expresada en las mismas unidades. Tanto la varianza como la desviación estándar miden la variabilidad de la variable aleatoria.

Hay que tener en cuenta de que la varianza puede verse muy influida por los outliers y se desaconseja su uso cuando las distribuciones de las variables aleatorias tienen colas pesadas. En tales casos se recomienda el uso de otras medidas de dispersión más robustas.

El término *varianza* fue acuñado por Ronald Fisher en un artículo de 1918 titulado *The Correlation Between Relatives on the Supposition of Mendelian Inheritance*.

Definición

Si la variable aleatoria x tiene media $\mu = \mathrm{E}(X)$ se define la varianza $\mathrm{Var}(X)$ (también representada como σ_X^2o, simplemente σ^2) de X como

$$\mathrm{Var}(X) = \mathrm{E}[(X - \mu)^2].$$

Desarrollando la definición anterior, se obtiene:

$$\begin{aligned}
\mathrm{Var}(X) &= \mathrm{E}[(X - \mu)^2] \\
&= \mathrm{E}[(X^2 - 2X\mu - \mu^2)] \\
&= \mathrm{E}(X^2) - 2\mu\mathrm{E}(X) + \mu^2 \\
&= \mathrm{E}(X^2) - 2\mu^2 + \mu^2 \\
&= \mathrm{E}(X^2) - \mu^2.
\end{aligned}$$

Si una distribución no tiene esperanza, como ocurre con la de Cauchy, tampoco tiene varianza. Existen otras distribuciones que, aun teniendo esperanza, carecen de varianza.

Un ejemplo de ellas es la de Pareto cuando su índice k satisface $1 < k \leq 2$.

Caso continuo

Si la variable aleatoria X es continua con función de densidad $p(x)$, entonces:

$$\operatorname{Var}(X) = \int (x-\mu)^2 p(x)\,dx\,,$$

donde:

$$\mu = \int x\,p(x)\,dx\,,$$

y las integrales están definidas sobre el rango de X.

Caso discreto

Si la variable aleatoria X es discreta con pesos $x_1 \mapsto p_1$, ..., $x_n \mapsto p_n$, entonces

$$\operatorname{Var}(X) = \sum_{i=1}^{n} (x_i-\mu)^2 p_i\,.$$

Ejemplos

Distribución exponencial

La distribución exponencial de parámetro λ es una distribución continua con soporte en el intervalo $[0,\infty)$ y función de densidad

$$f(x) = \lambda e^{-\lambda x} 1_{[0,\infty)}(x),$$

Tiene media $\mu = \lambda^{-1}$. Por lo tanto, su varianza es:

$$\int_0^\infty f(x)(x-\mu)^2\,dx = \int_0^\infty \lambda e^{-\lambda x}(x-\lambda^{-1})^2\,dx = \lambda^{-2}.$$

Es decir, $\sigma^2 = \mu^2$.

Dado perfecto

Un dado de seis caras puede modelarse como una variable aleatoria discreta que toma los valores del uno al seis con probabilidad igual a $^1/_6$. El valor esperado es (1+2+3+4+5+6)/6 = 3.5. Por lo tanto, su varianza es:

$$\sum_{i=1}^{6} \tfrac{1}{6}(i-3.5)^2 = \tfrac{1}{6}\left((-2.5)^2+(-1.5)^2+(-0.5)^2+0.5^2+1.5^2+2.5^2\right) = \tfrac{1}{6}\cdot 17.50 = \tfrac{35}{12} \approx 2.92\,.$$

Propiedades de la varianza

Algunas propiedades de la varianza son:

- $V(X) \geq 0$
- $V(aX+b) = a^2V(X)$ siendo *a* y *b* números reales cualesquiera. De esta propiedad se deduce que la varianza de una constante es cero, es decir, $V(b) = 0$
- $V(X+Y) = V(X) + V(Y) + 2Cov(X,Y)$, donde Cov(*X*,*Y*) es la covarianza de *X* e *Y*.
- $V(X-Y) = V(X) + V(Y) - 2Cov(X,Y)$, donde Cov(*X*,*Y*) es la covarianza de *X* e *Y*.

Varianza muestral

En muchas situaciones es preciso estimar la varianza de una población a partir de una muestra. Si se toma una muestra con reemplazamiento $(y_1, \ldots, y_n)$ de *n* valores de ella, de entre todos los estimadores posibles de la varianza de la población de partida, existen dos de uso corriente:

$$s_n^2 = \frac{1}{n}\sum_{i=1}^{n}(y_i - \overline{y})^2 = \left(\frac{1}{n}\sum_{i=1}^{n} y_i^2\right) - \overline{y}^2$$

y

$$s^2 = \frac{\sum_{i=1}^{n}(x_i - \overline{x})^2}{n-1} = \frac{1}{n-1}\sum_{i=1}^{n} x_i^2 - \frac{n}{n-1}\overline{x}^2 = \frac{\sum_{i-1}^{n} x_i^2 - n\overline{x}^2}{n-1}$$

Cuando los datos están agrupados:

$$s^2 = \frac{\sum_{i=1}^{n}(f_i) * (x_i - \overline{x})^2}{n-1} = \frac{\sum_{i-1}^{n} f_i * x_i^2 - n\overline{x}^2}{n-1}$$

A los dos (cuando está dividido por n y cuando lo está por n-1) se los denomina **varianza muestral**. Difieren ligeramente y, para valores grandes de *n*, la diferencia es irrelevante.

El primero traslada directamente la varianza de la muestra al de la población y el segundo es un estimador insesgado de la varianza de la población. De hecho,

$$\begin{aligned}
\mathrm{E}[s^2] &= \mathrm{E}\left[\frac{1}{n-1}\sum_{i=1}^{n} Y_i^2 \; - \; \frac{n}{n-1}\overline{Y}^2\right] \\
&= \frac{1}{n-1}\left(\sum \mathrm{E}[Y_i^2] \; - \; n\mathrm{E}[\overline{Y}^2]\right) \\
&= \frac{1}{n-1}\left(n\mathrm{E}[Y_1^2] \; - \; n\mathrm{E}[\overline{Y}^2]\right) \\
&= \frac{n}{n-1}\left(\mathrm{Var}(Y_1) + \mathrm{E}[Y_1]^2 \; - \; \mathrm{Var}(\overline{Y}) - \mathrm{E}[\overline{Y}]^2\right) \\
&= \frac{n}{n-1}\left(\mathrm{Var}(Y_1) + \mu^2 \; - \; \frac{1}{n}\mathrm{Var}(Y_1) - \mu^2\right) \\
&= \frac{n}{n-1}\left(\frac{n-1}{n}\,\mathrm{Var}(Y_1)\right) \\
&= \mathrm{Var}(Y_1) \\
&= \sigma^2
\end{aligned}$$

mientras que

$$E[s_n^2] = \frac{n-1}{n}\sigma^2$$

Distribución de la varianza muestral bajo condiciones de normalidad

Siendo una función de variables aleatorias, la varianza muestral es, en sí misma, otra variable aleatoria. Si y_i son muestras independientes de una variable aleatoria normal, el teorema de Cochran muestra que s^2 sigue una distribución chi-cuadrado

$$(n-1)\frac{s^2}{\sigma^2} \sim \chi^2_{n-1}.$$

Como consecuencia de lo anterior, $\mathrm{E}(s^2) = \sigma^2$.

Si los y_i son independientes y están idénticamente distribuidos, aunque no sean necesariamente normales, entonces s^2 es un estadístico insesgado de σ^2. Si se cumplen las condiciones necesarias para la ley de los grandes números, s^2 es un estimador consistente de σ^2.

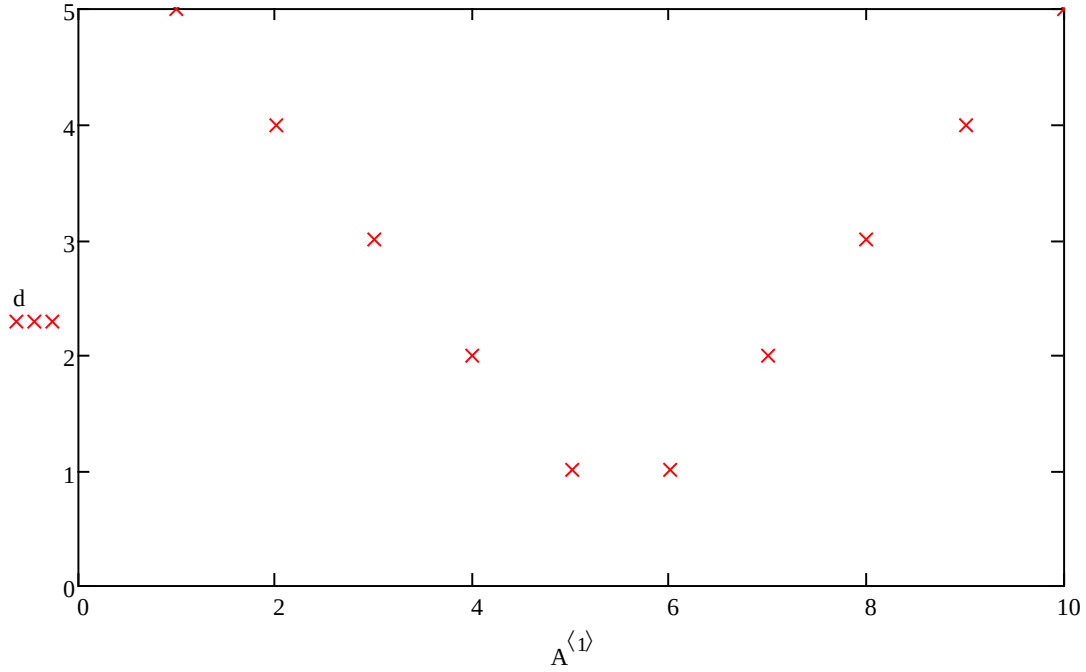
✓ Resolución del Caso Práctico nº 5:

La aproximación es la forma: $f(x) = a.x^2 + b.x + c$ (es decir, una aproximación funcional cuadrática).

Vector incógnita: $x = (a\ \ b\ \ c)^T$

Matriz de datos y función para la interpolación: $Ai = (1\ \ x_i\ \ x_i^2)$

Graficando los punto datos: [Ai, 1; di]



Ahora, buscamos minimizar el error cuadrático. Entonces, y valiéndonos de la teoría, resolvemos el siguiente sistema de ecuaciones:

$$\left(A^T \cdot A\right) \cdot x = A^T \cdot d$$

Aplicamos, primeramente, el método de Cholesky. Es decir, utilizamos un programa que realiza la descomposición de Cholesky para una matriz simétrica. Con el cual obtenemos:

$$S\left[\left(A^T \cdot A\right), 3\right] = \begin{pmatrix} 3.162 & 17.393 & 121.748 \\ 0 & 9.083 & 99.912 \\ 0 & 0 & 22.978 \end{pmatrix} \qquad \underset{\wedge\wedge\wedge}{S} := S\left[\left(A^T \cdot A\right), 3\right]$$

Siendo:

$$\left(S^T \cdot S\right) - \left(A^T \cdot A\right) = \begin{pmatrix} 1.776 \times 10^{-15} & -7.105 \times 10^{-15} & 0 \\ -7.105 \times 10^{-15} & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

Ahora nos disponemos a resolver dos (2) sistemas de ecuaciones basados en matrices triangulares inferiores y superiores respectivamente:

$$\left(S^T \cdot S\right) \cdot x = \left(A^T \cdot d\right)$$

$$S^T \cdot y = \left(A^T \cdot d\right)$$

$$S \cdot x = y$$

Para la resolución de un sistema de ecuaciones basado en una matriz triangular inferior, utilizamos:

$$S^T \cdot y = \left(A^T \cdot d\right)$$

$$TI\left[S^T, \left(A^T \cdot d\right), 3\right] = \begin{pmatrix} 9.487 \\ 3.129 \times 10^{-15} \\ 4.352 \end{pmatrix}$$

Donde:

$$y := TI\left[S^T, \left(A^T \cdot d\right), 3\right]$$

Ahora resolvemos el sistema de ecuaciones basado en una matriz triangular superior:

$$S \cdot x = y$$

$$TS(S, y, 3) = \begin{pmatrix} 7.167 \\ -2.083 \\ 0.189 \end{pmatrix}$$

Entonces: $xchol := TS(S, y, 3)$

Por lo que el polimonio quedaría:

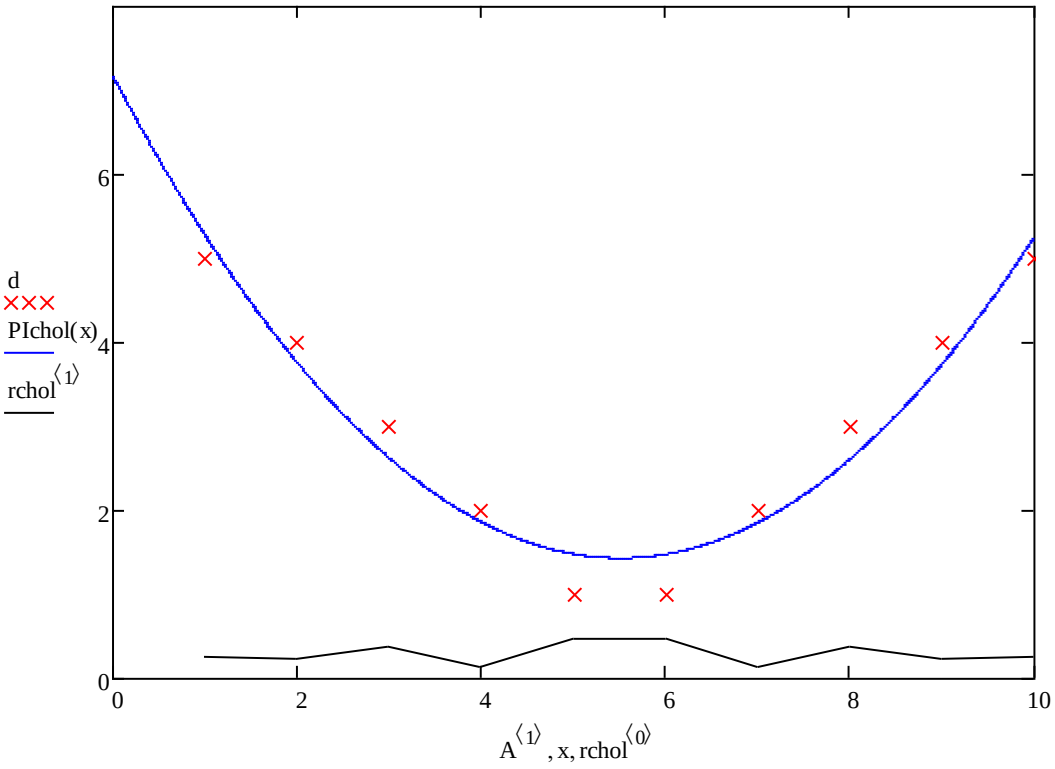
$$PIchol(x) := \sum_{i=0}^{2} \left(xchol_i \cdot x^i \right)$$

Y el vector resto:

$i := 0..9$

$rchol_{i,1} := \left| d_i - PIchol\left[\left(A^{\langle 1 \rangle} \right)_i \right] \right|$

$rchol_{i,0} := \left(A^{\langle 1 \rangle} \right)_i$



Finalmente, con el método de Cholesky, el error cuadrático es:

$$\frac{\sum_{i=0}^{9} \left(rchol_{i,1} \right)^2}{9} = 1.178451178510^{-1}$$

Ahora realizamos una transformación ortogonal, aplicando el método o reflexión de Householder, para obtener una matriz triangular superior:

$$Q := \prod_{i=1}^{(2)} TH\left[\left(A^T \cdot A\right), 3\right]_i$$

$$Q = \begin{pmatrix} 0.026 & -0.448 & 0.894 \\ 0.141 & -0.883 & -0.447 \\ 0.99 & 0.138 & 0.041 \end{pmatrix}$$

En la descomposición de Householder, de acuerdo con la teoría, se cumple que:

$$Q \cdot Q^T = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

En donde llamamos R:

$$R := Q^T \cdot \left(A^T \cdot A\right)$$

$$R = \begin{pmatrix} 389.037 & 30 & 49.451 \\ -0 & 25 & 2.203 \\ 0 & -0 & 21.449 \end{pmatrix} \begin{matrix} 25 & 07.659 \\ 647.012 \\ \end{matrix}$$

$$Q \cdot R = \left(A^T \cdot A\right)$$

En donde vemos que el sistema a resolver es de la forma:

$$Q \cdot R \cdot x = A^T \cdot d$$

Operando:

$$R \cdot x = Q^T \cdot \left(A^T \cdot d\right)$$

Ahora generamos un sistema de ecuaciones basado en una matriz triangular inferior. De donde:

$$TS\left[R, \left[Q^T \cdot \left(A^T \cdot d\right)\right], 3\right] = \begin{pmatrix} 7.167 \\ -2.083 \\ 0.189 \end{pmatrix}$$

Entonces:

$$xhouse := TS\left[R, \left[Q^T \cdot \left(A^T \cdot d\right)\right], 3\right]$$

Por lo que, el polimonio interpolador quedaría:

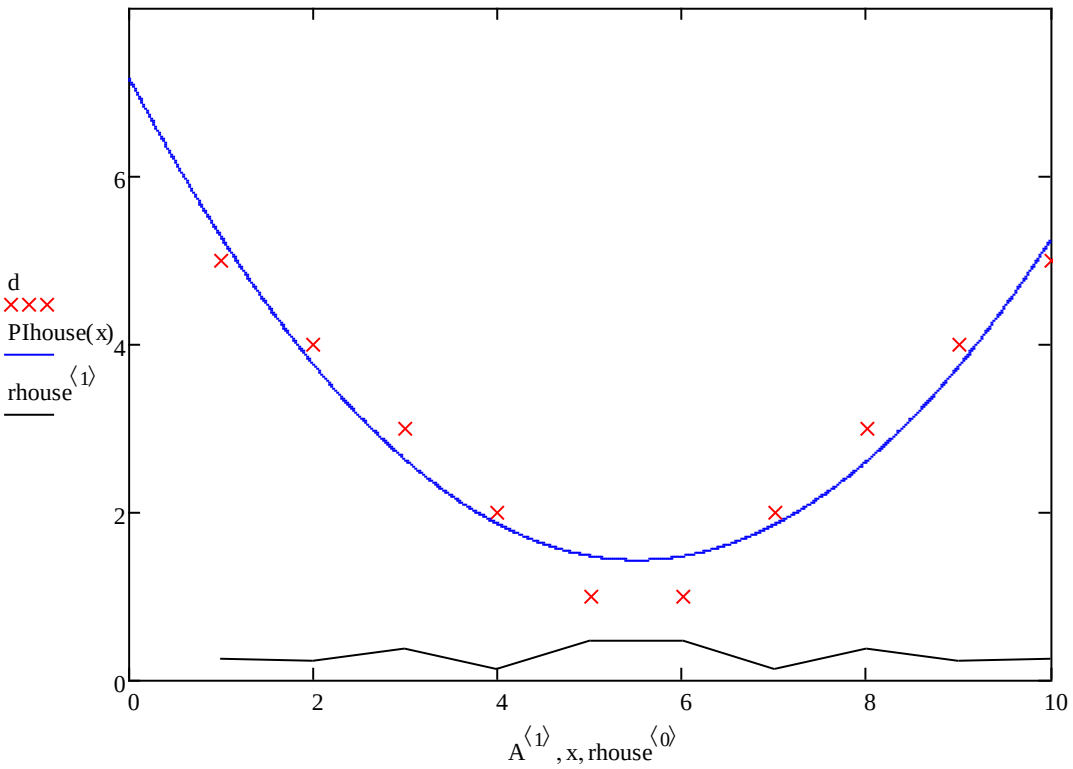
$$PIhouse(x) := \sum_{i=0}^{2} \left(xhouse_i \cdot x^i \right)$$

Y el vector resto:

$i := 0..9$

$$rhouse_{i,1} := \left| d_i - PIhouse\left[\left(A^{\langle 1 \rangle} \right)_i \right] \right|$$

$$rhouse_{i,0} := \left(A^{\langle 1 \rangle} \right)_i$$



Finalmente, con el método o reflexión de Householder, el error cuadrático es:

$$\frac{\sum_{i=0}^{9} \left(rhouse_{i,1} \right)^2}{9} = 1.178451178510^{-1}$$

Como **conclusión**, cabe destacar que, tanto con la utilización del método de Cholesky como con la utilización del método o reflexión de Householder se obtienen los mismos coeficientes con un error cuadrático medio igual para ambos casos. Siendo, para este caso propuesto, más favorable la aplicación del método de Householder.

Matriz de Covarianza (C), Matriz de Correlación (CL) y Varianza

Ahora nos disponemos a realizar el cálculo de la **Matriz de Covarianza (C):**

$C := \left[\left(S^T \cdot S \right)^{-1} \right]$ siendo S la matriz triangular superior (del método de Cholesky)

$$C = \begin{pmatrix} 1.383 & -0.525 & 0.042 \\ -0.525 & 0.241 & -0.021 \\ 0.042 & -0.021 & 0.002 \end{pmatrix} \qquad \left[\left(A^T \cdot A \right)^{-1} \right] = \begin{pmatrix} 1.383 & -0.525 & 0.042 \\ -0.525 & 0.241 & -0.021 \\ 0.042 & -0.021 & 0.002 \end{pmatrix}$$

Ahora calculamos la **Matriz de Correlación (CL):**

$i := 0..2$

$j := 0..2$

$$CL_{i,j} := \left(\frac{C_{i,j}}{\sqrt{C_{i,i} \cdot C_{j,j}}} \right)$$

$$CL = \begin{pmatrix} 1 & -0.909 & 0.814 \\ -0.909 & 1 & -0.975 \\ 0.814 & -0.975 & 1 \end{pmatrix}$$

Ahora calculamos, para los métodos de Cholesky y de Householder, la **Varianza:**

Para el método de **Cholesky:**

$$Ech := \frac{\sum_{i=0}^{9} rchol_{i,1}}{9}$$

todos los residuos de Cholesky tienen la misma probabilidad de ocurrir.

$$Vch := \frac{\sum_{i=0}^{9} \left(rchol_{i,1} - Ech\right)^2}{9}$$

$Ech = 0.337$

$Vch = 0.017$

Para el método de **Householder:**

$$Eh := \frac{\sum_{i=0}^{9} rhouse_{i,1}}{9}$$

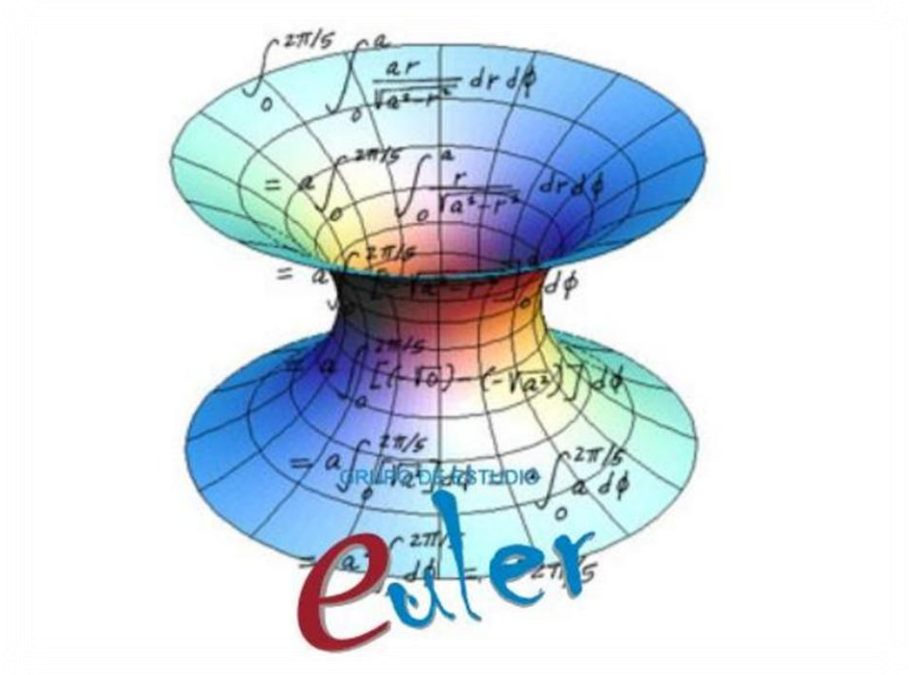
todos los residuos de Householder tienen la misma probabilidad de ocurrir.

$$Vch := \frac{\sum_{i=0}^{9} \left(rchol_{i,1} - Ech\right)^2}{9}$$

$Ech = 0.337$

$Vch = 0.017$

Caso Práctico nº 6



Métodos Numéricos 2009

Master Ing. Pablo DE SIMONE

Ing. Roberto HAARTH

Caso Práctico nº 6

✓ Enunciado:

Hallar un función de mapeo i,j → k (siendo k el factor de desplazamiento), para una matriz Hessenberg superior o inferior (matriz triangular con una sub-diagonal o super-diagonal).

$n \gg 1 \quad \sim \quad 50\%$

$$\frac{n(n+1)}{2} + (n-1)$$

✓ Teoría:

Función de Mapeo

En matemáticas, una **función**, **aplicación** o **mapeo** *f* es una relación entre un conjunto dado *X* (el dominio) y otro conjunto de elementos *Y* (el codominio) de forma que a cada elemento *x* del dominio le corresponde un único elemento del codominio *f(x)*. Se denota por:

$$f: X \to Y$$

Comúnmente, el término *función* se utiliza cuando el codominio son valores numéricos, reales o complejos. Entonces se habla de función real o función compleja mientras que a las funciones entre conjuntos cualesquiera se las denomina **aplicaciones**.

Matriz de Hessenberg

En álgebra lineal, una **matriz de Hessenberg** es una matriz "casi" triangular. Para ser más exactos, una **matriz superior de Hessenberg** tiene todos ceros por debajo de la primera subdiagonal, y una **matriz inferior de Hessenberg** tiene todos ceros por encima de la primera superdiagonal.

Por ejemplo:

$$\begin{bmatrix} 1 & 4 & 2 & 3 \\ 3 & 4 & 1 & 7 \\ 0 & 2 & 3 & 4 \\ 0 & 0 & 1 & 3 \end{bmatrix}$$

es una matriz de Hessenberg superior.

$$\begin{bmatrix} 1 & 2 & 0 & 0 \\ 5 & 2 & 3 & 0 \\ 3 & 4 & 3 & 7 \\ 5 & 6 & 1 & 1 \end{bmatrix}$$

es una matriz Hessenberg inferior.

Matriz triangular

En álgebra lineal, una **matriz triangular** es un tipo especial de matriz cuadrada cuyos elementos por encima o por debajo de su diagonal principal son cero. Debido a que los sistemas de ecuaciones lineales con matrices triangulares son mucho más fáciles de resolver, las matrices triangulares son utilizadas en análisis numérico para resolver sistemas de ecuaciones lineales, calcular inversas y determinantes de matrices. El método de descomposición LU permite descomponer cualquier matriz invertible como producto de una matriz triangular inferior *L* y una superior *U*.

Descripción

Una matriz cuadrada de orden *n* se dice que es **triangular superior** si es de la forma:

$$U = \begin{pmatrix} u_{11} & u_{12} & u_{13} & . & . & . & u_{1n} \\ 0 & u_{22} & u_{23} & . & . & . & u_{2n} \\ 0 & 0 & u_{33} & . & . & . & u_{3n} \\ . & . & .. & . & . & . & . \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ 0 & 0 & 0 & . & . & . & u_{nn} \end{pmatrix}$$

Análogamente, una matriz de la forma:

$$L = \begin{pmatrix} l_{11} & 0 & 0 & . & . & . & 0 \\ l_{21} & l_{22} & 0 & . & . & . & 0 \\ l_{31} & l_{32} & l_{33} & . & . & . & 0 \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ l_{n1} & l_{n2} & l_{n3} & . & . & . & l_{nn} \end{pmatrix}$$

se dice que es una **matriz triangular inferior**.

Se suelen emplear las letras U y L, respectivamente, ya que U es la inicial de "upper triangular matrix" y L de "lower triangular matrix", los nombres que reciben estas matrices en inglés.

Ejemplos

$$\begin{pmatrix} 1 & 4 & 2 \\ 0 & 3 & 4 \\ 0 & 0 & 1 \end{pmatrix}$$

es triangular superior y

$$\begin{pmatrix} 1 & 0 & 0 \\ 2 & 8 & 0 \\ 4 & 9 & 7 \end{pmatrix}$$

es triangular inferior.

Propiedades de las matrices triangulares

- Una matriz triangular superior e inferior es una matriz diagonal.
- El producto de dos matrices triangulares superiores (inferiores) es un matriz triangular superior (inferior).
- La transpuesta de una matriz triangular superior es una matriz triangular inferior y viceversa.
- El determinante de una matriz triangular es el producto de los elementos de la diagonal.
- Una matriz triangular es invertible si y solo si todos los elementos de la diagonal son no nulos. En este caso, la inversa de una matriz triangular superior (inferior) es otra matriz superior (inferior).
- Los valores propios de una matriz triangular son los elementos de la diagonal principal.

Aplicaciones

Un sistema de ecuaciones lineales en forma matricial

$$\mathbf{Lx} = \mathbf{b}$$

o

$$\mathbf{Ux} = \mathbf{b}$$

es muy fácil de resolver. El primer sistema puede escribirse como

$$\begin{array}{ccccccc} x_1 & & & & & = & b_1 \\ l_{2,1}x_1 & + & x_2 & & & = & b_2 \\ \vdots & & \vdots & \ddots & & & \vdots \\ l_{m,1}x_1 & + & l_{m,2}x_2 & + \ldots + & x_m & = & b_m \end{array}$$

que puede resolverse siguiendo un simple algoritmo recursivo

$$\begin{array}{rcl} x_1 & = & b_1 \\ x_2 & = & b_2 - l_{2,1}x_1 \\ & \vdots & \\ x_m & = & b_m - \sum_{i=1}^{m-1} l_{m,i}x_i \end{array} \quad :$$

De forma análoga puede resolverse un sistema dado por una matriz triangular superior.

✓ Resolución del Caso Práctico nº 6:

Para la resolución del caso en estudio, primeramente realizaremos la descomposición de una matriz de Hessenberg superior en un vector el cuál contendrá los valores de dicha matriz distintos de cero (0). Seguidamente re-compondremos, a partir de dicho vector, la matriz original.

Primero definimos la matriz de Hessenberg. La misma es creada con número aleatorios que van de 0 a 100.

A =

	1	2	3	4	5	6	7	8	9	10
1	0.127	19.332	58.501	35.031	82.284	17.413	71.05	30.399	9.141	14.731
2	15.326	98.851	11.908	0.892	53.166	60.176	16.625	45.079	5.706	78.332
3	0	82.167	51.988	87.597	95.59	53.934	46.207	86.222	77.966	99.68
4	0	0	19.135	61.149	26.621	84.012	37.586	67.719	0.882	27.589
5	0	0	0	81.718	58.791	83.761	48.493	74.373	45.798	74.442
6	0	0	0	0	15.556	59.904	73.5	57.24	15.156	42.516
7	0	0	0	0	0	73.201	51.712	75.154	16.9	49.188
8	0	0	0	0	0	0	27.959	69.975	14.752	14.16
9	0	0	0	0	0	0	0	68.224	69.288	42.655
10	0	0	0	0	0	0	0	0	72.191	96.66

Luego, creamos el vector cU que recogerá los valores distintos de cero (0).

$cU(A, Nmax)^T =$

	1	2	3	4	5	6	7	8	9	10	11
1	0.127	19.332	98.851	58.501	11.908	51.988	35.031	0.892	87.597	61.149	...

Ahora, a partir de los datos obtenidos en el vector (cU), compondremos la matriz.

Primeramente, asignamos a un vector (v) los valores de la matriz original A.

$v := cU(A, Nmax)$

Y, finalmente (y a partir del vector dado), mapeamos la matriz original, A:

$McU(v, Nmax) =$

	1	2	3	4	5	6	7	8	9	10
1	0.127	19.332	58.501	35.031	82.284	17.413	71.05	30.399	9.141	14.731
2	15.326	98.851	11.908	0.892	53.166	60.176	16.625	45.079	5.706	78.332
3	0	82.167	51.988	87.597	95.59	53.934	46.207	86.222	77.966	99.68
4	0	0	19.135	61.149	26.621	84.012	37.586	67.719	0.882	27.589
5	0	0	0	81.718	58.791	83.761	48.493	74.373	45.798	74.442
6	0	0	0	0	15.556	59.904	73.5	57.24	15.156	42.516
7	0	0	0	0	0	73.201	51.712	75.154	16.9	49.188
8	0	0	0	0	0	0	27.959	69.975	14.752	14.16
9	0	0	0	0	0	0	0	68.224	69.288	42.655
10	0	0	0	0	0	0	0	0	72.191	96.66

Para finalizar, y a modo de verificación, realizamos la resta entre la matriz final y la matriz original, corroborando que sus elementos máximos y sus determinantes tiendan a cero (0):

$$\max(McU(v, Nmax) - A) = 0$$

$$\left|McU(v, Nmax) - A\right| = 0$$

Caso Práctico nº 7

Métodos Numéricos 2009

Master Ing. Pablo DE SIMONE

Ing. Roberto HAARTH

Caso Práctico nº 7

✓ Enunciado:

$$A := \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 2 & 3 & 4 & 5 \\ 1 & 3 & 6 & 10 & 15 \\ 1 & 4 & 10 & 20 & 35 \\ 1 & 5 & 15 & 35 & 70 \end{pmatrix}$$ Matriz de Pascal

a. A partir de la matriz de Pascal, calcular los autovalores y sus autovectores.
b. Demostrar que las filas cumplen con el teorema de Gershgorin.
c. Agregar un error de 1% y demostrar que hay una cota para el error de los autovalores.

✓ Teoría:

Pascal´s Matrix

In mathematics, particularly matrix theory and combinatorics, the **Pascal matrix** is an infinite matrix containing the binomial coefficients as its elements. There are 3 ways this can be achieved - either as an upper-triangular matrix, a lower-triangular matrix, or as a symmetric matrix. The 5×5 truncations of these are shown below.

Upper triangular:

$$U_5 = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 2 & 3 & 4 \\ 0 & 0 & 1 & 3 & 6 \\ 0 & 0 & 0 & 1 & 4 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix};$$

Lower triangular:

$$L_5 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 2 & 1 & 0 & 0 \\ 1 & 3 & 3 & 1 & 0 \\ 1 & 4 & 6 & 4 & 1 \end{pmatrix};$$

Symmetric:

$$S_5 = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 2 & 3 & 4 & 5 \\ 1 & 3 & 6 & 10 & 15 \\ 1 & 4 & 10 & 20 & 35 \\ 1 & 5 & 15 & 35 & 70 \end{pmatrix}.$$

These matrices have the pleasing relationship $S_n = L_n U_n$. From this it is easily seen that all three matrices have determinant 1, as the determinant of a triangular matrix is simply the product of its diagonal elements, which are all 1 for both L_n and U_n. In other words, matrices S_n, L_n, and U_n are unimodular, with L_n and U_n having trace n.

The elements of the symmetric Pascal matrix are the binomial coefficients, i.e.

$$S_{ij} = \binom{n}{r} = \frac{n!}{r!(n-r)!}, \quad \text{where} \quad n = i + j - 2, \quad r = i - 1.$$

In other words,

$$S_{ij} = {}^{i+j-2}\mathbf{C}_{i-1} = \frac{(i+j-2)!}{(i-1)!(j-1)!}.$$

Thus the trace of S_n is given by

$$\operatorname{tr}(S_n) = \sum_{i=1}^{n} \frac{[2(i-1)]!}{[(i-1)!]^2} = \sum_{k=0}^{n-1} \frac{(2k)!}{(k!)^2}$$

with the first few terms given by the sequence 1, 3, 9, 29, 99, 351, 1275, ...

Construction

The Pascal matrix can actually be constructed by taking the matrix exponential of a special subdiagonal or superdiagonal matrix. The example below constructs a 7-by-7 Pascal matrix, but the method works for any desired $n \times n$ Pascal matrices. (Note that dots in the following matrices represent zero elements.)

$$L_7 = \exp\left(\begin{bmatrix} . & . & . & . & . & . & . \\ 1 & . & . & . & . & . & . \\ . & 2 & . & . & . & . & . \\ . & . & 3 & . & . & . & . \\ . & . & . & 4 & . & . & . \\ . & . & . & . & 5 & . & . \\ . & . & . & . & . & 6 & . \end{bmatrix}\right) = \begin{bmatrix} 1 & . & . & . & . & . & . \\ 1 & 1 & . & . & . & . & . \\ 1 & 2 & 1 & . & . & . & . \\ 1 & 3 & 3 & 1 & . & . & . \\ 1 & 4 & 6 & 4 & 1 & . & . \\ 1 & 5 & 10 & 10 & 5 & 1 & . \\ 1 & 6 & 15 & 20 & 15 & 6 & 1 \end{bmatrix};$$

$$U_7 = \exp\left(\begin{bmatrix} . & 1 & . & . & . & . & . \\ . & . & 2 & . & . & . & . \\ . & . & . & 3 & . & . & . \\ . & . & . & . & 4 & . & . \\ . & . & . & . & . & 5 & . \\ . & . & . & . & . & . & 6 \\ . & . & . & . & . & . & . \end{bmatrix}\right) = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ . & 1 & 2 & 3 & 4 & 5 & 6 \\ . & . & 1 & 3 & 6 & 10 & 15 \\ . & . & . & 1 & 4 & 10 & 20 \\ . & . & . & . & 1 & 5 & 15 \\ . & . & . & . & . & 1 & 6 \\ . & . & . & . & . & . & 1 \end{bmatrix};$$

$$\therefore \quad S_7 = \exp\left(\begin{bmatrix} . & . & . & . & . & . & . \\ 1 & . & . & . & . & . & . \\ . & 2 & . & . & . & . & . \\ . & . & 3 & . & . & . & . \\ . & . & . & 4 & . & . & . \\ . & . & . & . & 5 & . & . \\ . & . & . & . & . & 6 & . \end{bmatrix}\right) \exp\left(\begin{bmatrix} . & 1 & . & . & . & . & . \\ . & . & 2 & . & . & . & . \\ . & . & . & 3 & . & . & . \\ . & . & . & . & 4 & . & . \\ . & . & . & . & . & 5 & . \\ . & . & . & . & . & . & 6 \\ . & . & . & . & . & . & . \end{bmatrix}\right) = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 1 & 3 & 6 & 10 & 15 & 21 & 28 \\ 1 & 4 & 10 & 20 & 35 & 56 & 84 \\ 1 & 5 & 15 & 35 & 70 & 126 & 210 \\ 1 & 6 & 21 & 56 & 126 & 252 & 462 \\ 1 & 7 & 28 & 84 & 210 & 462 & 924 \end{bmatrix}.$$

It is important to note that you cannot simply assume exp(*A*)exp(*B*) = exp(*A* + *B*), for *A* and *B* *n*×*n* matrices. Such an identity only holds when *AB* = *BA* (i.e. the matrices *A* and *B* commute). In the construction of symmetric Pascal matrices like that above, the sub- and superdiagonal matrices do not commute, so the (perhaps) tempting simplification involving the addition of the matrices cannot be made.

A useful property of the sub- and superdiagonal matrices used in the construction is that both are nilpotent; that is, when raised to a sufficiently high integer power, they degenerate into the zero matrix. As the *n*×*n* generalised shift matrices we are using become zero when raised to power *n*, when calculating the matrix exponential we need only consider the first *n* + 1 terms of the infinite series to obtain an exact result.

Variants

Interesting variants can be obtained by obvious modification of the matrix-logarithm PL_7 and then application of the matrix exponential.

The first example below uses the squares of the values of the log-matrix and constructs a 7-by-7 "Laguerre"- matrix (or matrix of coefficients of Laguerre-polynomials)

$$LAG_7 = \exp\left(\begin{bmatrix} . & . & . & . & . & . & . \\ 1 & . & . & . & . & . & . \\ . & 4 & . & . & . & . & . \\ . & . & 9 & . & . & . & . \\ . & . & . & 16 & . & . & . \\ . & . & . & . & 25 & . & . \\ . & . & . & . & . & 36 & . \end{bmatrix}\right) = \begin{bmatrix} 1 & . & . & . & . & . & . \\ 1 & 1 & . & . & . & . & . \\ 2 & 4 & 1 & . & . & . & . \\ 6 & 18 & 9 & 1 & . & . & . \\ 24 & 96 & 72 & 16 & 1 & . & . \\ 120 & 600 & 600 & 200 & 25 & 1 & . \\ 720 & 4320 & 5400 & 2400 & 450 & 36 & 1 \end{bmatrix};$$

The Laguerre-matrix is actually used with some other scaling and/or the scheme of alternating signs. (Literature about generalizations to higher powers is not found yet)

The second example below uses the products v*(v+1) of the values of the log-matrix and constructs a 7-by-7 "Lah"- matrix (or matrix of coefficients of Lah numbers)

$$LAH_7 = \exp\left(\begin{bmatrix} 2 & . & . & . & . & . & . \\ . & 6 & . & . & . & . & . \\ . & . & 12 & . & . & . & . \\ . & . & . & 20 & . & . & . \\ . & . & . & . & 30 & . & . \\ . & . & . & . & . & 42 & . \end{bmatrix}\right) = \begin{bmatrix} 1 & . & . & . & . & . & . \\ 2 & 1 & . & . & . & . & . \\ 6 & 6 & 1 & . & . & . & . \\ 24 & 36 & 12 & 1 & . & . & . \\ 120 & 240 & 120 & 20 & 1 & . & . \\ 720 & 1800 & 1200 & 300 & 30 & 1 & . \\ 5040 & 15120 & 12600 & 4200 & 630 & 42 & 1 \\ 40320 & 141120 & 141120 & 58800 & 11760 & 1176 & 56 \end{bmatrix}$$

Using v*(v-1) instead provides a diagonal shifting to bottom-right.

Triángulo de Pascal

1
1 1
1 2 1
1 3 3 1
1 4 6 4 1
1 5 10 10 5 1
1 6 15 20 15 6 1
1 7 21 35 35 21 7 1
1 8 28 56 70 56 28 8 1
1 9 36 84 126 126 84 36 9 1

Triángulo de Pascal o de Tartaglia

El **triángulo de Pascal** en matemáticas es un conjunto infinito de números enteros ordenados en forma de triángulo que expresan coeficientes binomiales. El interés del Triángulo de Pascal radica en su aplicación en álgebra y permite calcular de forma sencilla números combinatorios lo que sirve para aplicar el binomio de Newton.

También es conocido como **Triángulo de Tartaglia**. En países orientales como China, India o Persia, este triángulo se conocía y fue estudiado por matemáticos como Al-Karaji, cinco siglos antes de que Pascal expusiera sus aplicaciones, o por el astrónomo y poeta persa Omar Jayyam (1048-1123). En China es conocido como **Triángulo de Yanghui**, en honor al matemático Yang Hui, quien lo describió el año 1303.[1]

Composición del Triángulo de Pascal

El Triángulo se construye de la siguiente manera: escribimos el número «1» centrado en la parte superior; después, escribimos una serie de números «1» en las casillas situadas en sentido diagonal descendente, a ambos lados; sumamos las parejas de cifras situadas horizontalmente (1 + 1), y el resultado (2) lo escribimos debajo de dichas casillas; continuamos el proceso escribiendo en las casillas inferiores la suma de las dos cifras situadas sobre ellas (1 + 2 = 3)...

Las cifras escritas en las filas, tales como: «1 2 1» y «1 3 3 1» recuerdan los coeficientes de las identidades:

$$(a+b)^2 = a^2 + 2ab + b^2$$
$$(a+b)^3 = a^3 + 3a^2b + 3ab^2 + b^3$$

pues son los coeficientes de sus monomios y, además, se puede generalizar para cualquier potencia del binomio: $(a+b)$

Interpretación en combinatoria

Los coeficientes binomiales son la base misma de la combinatoria. Veamos por qué: Tomemos de nuevo un binomio, por ejemplo $(a + b)^3$, y desarrollémoslo, pero de una manera distinta del párrafo anterior:

$$(a+b)^3 = (a+b)\cdot(a+b)\cdot(a+b)$$

luego quitemos las paréntesis, pero sin cambiar el orden en los productos, es decir sin aplicar la conmutatividad:

$$\begin{aligned}(a+b)\cdot(a+b)\cdot(a+b) &= (aa+ab+ba+bb)\cdot(a+b)\\ &= aaa+aab+aba+abb+baa+bab+bba+bbb\end{aligned}$$

Y agrupemos los términos que contienen el mismo número de a, (y de b):

$$= aaa + (aab+aba+baa) + (abb+bab+bba) + bbb$$

El primer paréntesis contiene todas las palabras constituidas de un *b* y dos *a*. En este caso, es fácil ver que hay exactamente tres. En el caso general, para contar las palabras, hay que aplicar la conmutatividad, pues las palabras que contienen el mismo número de a y b darán el mismo término:

$$= 1\cdot a^3 + 3\cdot a^2b + 3\cdot ab^2 + 1\cdot b^3$$

El primer factor 3, que es $\binom{3}{1}$ cuenta las tres palabras mencionadas (aab, aba y baa).

El segundo factor 3, que es $\binom{3}{2}$ cuenta las palabras hechas de dos *b* y un *a* (abb, bab y bba).

Obviamente, sólo hay una palabra de tres letras constituidas de *a* solamente, y esto corresponde al monomio **1·a³**, con 1 = $\binom{3}{0}$ («0 » por *ninguna b*).

En vez de hablar de palabras formadas con *a* y *b*, es equivalente imaginar una hilera de *n* cajones inicialmente vacíos, y *p* bolas intercambiables que se tienen que repartir, en cada cajón no cabiendo más de una. Se trata en todos casos de repartir *p* objetos entre *n* sitios posibles, o de escoger un grupo de *p* objetos/sitios entre *n* objetos/sitios. De ahí la apelación *p entre n*.

Todo lo anterior lleva al **teorema**:

Hay exactamente $\binom{n}{p}$ maneras de escoger un conjunto de *p* elementos entre *n* elementos.

En matemática formal, se prefiere hablar de conjuntos:

Existen $\binom{n}{p}$ subconjuntos de cardinal p en un conjunto de cardinal n.

Este punto de vista permite hallar la fórmula para los coeficientes binomiales. En efecto, para elegir el « primer » elemento, hay *n* posibilidades, luego para escoger el segundo quedan *n-1* posibilidades y así sucesivamente hasta el elemento número *p*, que tiene *n-p+1*. El orden en el que se ha elegido estos *p* elementos no importa, se podía haber obtenido el mismo subconjunto de *p* elementos en otro orden. Hay p! permutaciones posibles de estos *p* elementos, es decir p! maneras de obtener el mismo conjunto.

Por tanto hay $\frac{n \cdot (n-1) \cdot (n-2)...(n-p+1)}{p!}$ subconjuntos posibles.

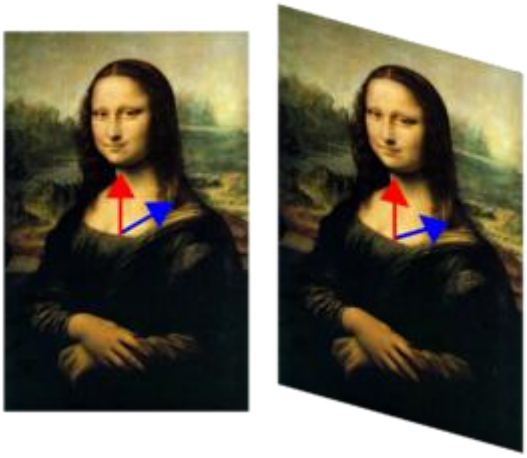
En conclusión:

$$\binom{n}{p} = \frac{n \cdot (n-1) \cdot (n-2)...(n-p+1)}{p \cdot (p-1) \cdot (p-2)...2 \cdot 1} = \frac{n!}{p! \cdot (n-p)!}$$

Verifiquémoslo en un ejemplo:

$$\binom{5}{2} = \frac{5!}{2!3!} = \frac{5 \times 4 \times 3 \times 2}{2 \times 3 \times 2} = 10$$

Vector propio y Valor propio



En esta transformación de la Mona Lisa, la imagen se ha deformado de tal forma que su eje vertical no ha cambiado. (nota: se han recortado las esquinas en la imagen de la derecha). El vector azul, representado por la flecha azul que va desde el pecho hasta el hombro, ha cambiado de dirección, mientras que el rojo, representado por la flecha roja, no ha cambiado. El vector rojo es entonces un **vector propio** de la transformación, mientras que el azul no lo es. Dado que el vector rojo no ha cambiado de longitud, su **valor propio** es 1. Todos los vectores de esta misma dirección son vectores propios, con el mismo valor propio. Forman el **espacio propio** de este valor propio.

En álgebra lineal, los **vectores propios**, **autovectores** o **eigenvectores** de un operador lineal son los vectores no nulos que, cuando son transformados por el operador, dan lugar a un múltiplo escalar de sí mismos, con lo que no cambian su dirección. Este escalar λ recibe el nombre **valor propio**, **autovalor**, **valor característico** o **eigenvalor**. A menudo, una transformación queda completamente determinada por sus vectores propios y valores propios. Un **espacio propio**, **autoespacio** o **eigenespacio** es el conjunto de vectores propios con un valor propio común.

La palabra alemana *eigen*, que se traduce en español como *propio* se usó por primera vez en este contexto por David Hilbert en 1904 (aunque Helmholtz la usó previamente con un significado parecido). *Eigen* se ha traducido también como *inherente*, *característico* o el prefijo *auto-*, donde se aprecia el énfasis en la importancia de los valores propios para definir la naturaleza única de una determinada transformación lineal. Las denominaciones vector y valor *característicos* también se utilizan habitualmente.

Definiciones

Las transformaciones lineales del espacio, como la rotación, la reflexión, el ensanchamiento, o cualquier combinación de las anteriores; en esta lista podrían incluirse otras transformaciones, pueden interpretarse mediante el efecto que producen en los vectores. Los vectores pueden visualizarse como flechas de una cierta longitud apuntando en una dirección y sentido determinados.

- Los **vectores propios** de las transformaciones lineales son vectores que, o no se ven afectados por la transformación o se ven multiplicados por un escalar, y por tanto no varían su dirección.
- El **valor propio** de un vector propio es el factor de escala por el que ha sido multiplicado.
- Un **espacio propio** es un espacio formado por todos los vectores propios del mismo valor propio, además del vector nulo, que no es un vector propio.
- La **multiplicidad geométrica** de un valor propio es la dimensión del espacio propio asociado.
- El **espectro** de una transformación en espacios vectoriales finitos es el conjunto de todos sus valores propios.

Por ejemplo, un *vector propio* de una rotación en tres dimensiones es un vector situado en el eje de rotación sobre el cual se realiza la rotación. El *valor propio* correspondiente es 1 y el *espacio propio* contiene a todos los vectores paralelos al eje. Como es un espacio de una dimensión, su *multiplicidad geométrica* es uno. Es el único valor propio del *espectro* (de esta rotación) que es un número real.

Formalmente, se definen los vectores propios y valores propios de la siguiente manera: Si **A**: $V \to V$ es un operador lineal en un cierto espacio vectorial V, **v** es un vector diferente de cero en V y c es un escalar tales que

$$\mathbf{A}\mathbf{v} = c\mathbf{v},$$

entonces decimos que **v** es un vector propio del operador **A**, y su valor propio asociado es c. Observe que si **v** es un vector propio con el valor propio c entonces cualquier múltiplo diferente de cero de **v** es también un vector propio con el valor propio c. De hecho, todos los vectores propios con el valor propio asociado c junto con **0**, forman un subespacio de V, el **espacio propio** para el valor propio c.

Ejemplo

A medida que la tierra rota, los vectores en el eje de rotación permanecen invariantes. Si se considera la transformación lineal que sufre la tierra tras una hora de rotación, una flecha que partiera del centro de la tierra al Polo Sur geográfico sería un vector propio de esta transformación, pero una flecha que partiera del centro a un punto del ecuador no sería un vector propio. Dado que la flecha que apunta al polo no cambia de longitud por la rotación, su valor propio es 1.

Ecuación del valor propio o autovalor

Matemáticamente, $\mathbf{v}_\lambda$ es un vector propio y λ el valor propio correspondiente de una transformación T si verifica la ecuación:

$$T(\mathbf{v}_\lambda) = \lambda\,\mathbf{v}_\lambda$$

donde $T(\mathbf{v}_\lambda)$ es el vector obtenido al aplicar la transformación T a v_λ.

Supóngase que T es una transformación lineal (lo que significa que $T(a\mathbf{v} + b\mathbf{w}) = aT(\mathbf{v}) + bT(\mathbf{w})$ para todos los escalares a, b, y los vectores $\mathbf{v}$, $\mathbf{w}$). Considérese una base en ese espacio vectorial. Entonces, T y $\mathbf{v}_\lambda$ pueden representarse en relación a esa base mediante una matriz A_T y un vector columna v_λ, un vector vertical unidimensional. La ecuación de valor propio en esta representación matricial se representa de la siguiente forma:

$$A_T\, v_\lambda = \lambda\, v_\lambda$$

donde la yuxtaposición es un producto de matrices. Dado que en esta circunstancia la transformación T y su representación matricial A_T son equivalentes, es proporcionalmente a sí misma si λ es negativa. Por ejemplo, una población ideal de conejos engendra con más frecuencia a medida que hay más conejos, y por tanto satisface la ecuación para lambda positivo.

La solución a la ecuación de valor propio es $g(t) = \exp(\lambda t)$, la función exponencial; pues esa función es una función propia del operador diferencial d/dt con el valor propio λ. Si λ es negativa, la evolución de g se denomina decaimiento exponencial; si es positiva se denomina crecimiento exponencial. El valor de λ puede ser cualquier número complejo. El espectro de d/dt es entonces el plano complejo en su totalidad. En este ejemplo el espacio vectorial en el que actúa d/dt es el espacio de las funciones derivables de una variable. Este espacio tiene una dimensión infinita (pues no es posible expresar cada función diferenciable como combinación lineal de un número finito de funciones base). No obstante, el espacio propio asociado a un valor propio determinado λ es unidimensional. Es el conjunto de todas las funciones $g(t) = A\exp(\lambda t)$, donde A es una constante arbitraria, la población inicial en $t=0$.

Teorema espectral

El *teorema espectral* muestra la importancia de los valores propios y vectores propios para caracterizar una transformación lineal de forma única. En su versión más simple, el teorema espectral establece que, bajo unas condiciones determinadas, una transformación lineal de un vector puede expresarse como la combinación lineal de los vectores propios con coeficientes de valor igual a los valores propios por el producto escalar de los vectores propios por el vector al que se aplica la transformación, lo que puede escribirse como:

$$\mathcal{T}(\mathbf{v}) = \lambda_1(\mathbf{v}_1 \cdot \mathbf{v})\mathbf{v}_1 + \lambda_2(\mathbf{v}_2 \cdot \mathbf{v})\mathbf{v}_2 + \dots$$

donde $\mathbf{v}_1, \mathbf{v}_2, \cdots$ y $\lambda_1, \lambda_2, \cdots$ representan a los vectores propios y valores propios de $\mathcal{T}$. El caso más simple en el que tiene validez el teorema es cuando la transformación lineal viene dada por una matriz simétrica real o una matriz hermítica compleja.

Si se define la enésima potencia de una transformación como el resultado de aplicarla *n* veces sucesivas, se puede definir también el polinomio de las transformaciones. Una versión más general del teorema es que cualquier polinomio *P* de T es igual a:

$$P(T)(\mathbf{v}) = P(\lambda_1)(\mathbf{v}_1 \cdot \mathbf{v})\mathbf{v}_1 + P(\lambda_2)(\mathbf{v}_2 \cdot \mathbf{v})\mathbf{v}_2 + \dots$$

El teorema puede extenderse a otras funciones o transformaciones tales como funciones analíticas, siendo el caso más general las funciones de Borel.

Vectores propios y valores propios de matrices

Cálculo de valores propios y vectores propios de matrices

Si se quiere calcular los valores propios de una matriz dada y ésta es pequeña, se puede calcular simbólicamente usando el polinomio característico. Sin embargo, a menudo resulta imposible para matrices extensas, caso en el que se debe usar un método numérico.

Cálculo simbólico.

Encontrando valores propios.

Una herramienta importante para encontrar valores propios de matrices cuadradas es el polinomio característico: decir que λ es un valor propio de A es equivalente a decir que el sistema de ecuaciones lineales $(A - \lambda I)\, v = 0$ (donde I es la matriz identidad) tiene una solución no nula v (un vector propio), y de esta forma es equivalente al determinante:

$$\det (A - \lambda I) = 0$$

La función $p(\lambda) = \det(A - \lambda I)$ es un polinomio de λ pues los determinante se definen como sumas de productos. Éste es el **polinomio característico** de A: los valores propios de una matriz son los ceros de su polinomio característico.

Todos los valores propios de una matriz A pueden calcularse resolviendo la ecuación $p_A(\lambda) = 0$.

Si A es una matriz $n \times n$, entonces p_A tiene grado n y A tiene como máximo n valores propios.

El teorema fundamental del álgebra dice que esta ecuación tiene exactamente n raíces (ceros), teniendo en cuenta su multiplicidad. Todos los polinomios reales de grado impar tienen un número real como raíz, así que para n impar toda matriz real tiene al menos valor propio real. En el caso de las matrices reales, para n par e impar, los valores propios no reales son pares conjugados.

Encontrando vectores propios.

Una vez que se conocen los valores propios λ, los vectores propios se pueden hallar resolviendo el sistema de ecuaciones homogéneo:

$$(A - \lambda I)\, v = 0$$

Una forma más sencilla de obtener vectores propios sin resolver un sistema de ecuaciones lineales se basa en el teorema de Cayley-Hamilton que establece que cada matriz cuadrada satisface su propio polinomio característico. Así, si $\lambda_1,\lambda_2,...,\lambda_n$ son los valores propios de A se cumple que:

$$(A - \lambda_1 I)(A - \lambda_2 I)...(A - \lambda_n I) = 0$$

por lo que los vectores columna de (A - \lambda_2 I)...(A - \lambda_n I) son vectores propios de λ_1.

Ejemplo de matriz sin valores propios reales.

Un ejemplo de matriz sin valores propios reales es la rotación de 90 grados en el sentido de las manecillas del reloj:

$$\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$$

cuyo polinomio característico es $-\lambda^2 - 1$ y sus valores propios son el par de conjugados complejos *i*, *-i*. Los vectores propios asociados tampoco son reales.

Ejemplo.

Considérese la matriz

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

que representa un operador lineal $\mathbf{R}^3 \rightarrow \mathbf{R}^3$. Si se desea computar todos los valores propios de **A**, se podría empezar determinando el polinomio característico:

$$p(x) = \det(A - xI) = \begin{vmatrix} -x & 1 & -1 \\ 1 & 1-x & 0 \\ -1 & 0 & 1-x \end{vmatrix}$$

$$= -x^3 + 2x^2 + x - 2$$

y porque $p(x) = -(x - 2)(x - 1)(x + 1)$ se ve que los valores propios de **A** son 2, 1 y -1. El teorema de Cayley-Hamilton establece que cada matriz cuadrada satisface su propio polinomio característico. Es decir:

$$(A - 2I)(A - I)(A + I) = 0$$

Efectivamente, para el caso del valor propio 2, se puede comprobar que:

$$(A - I)(A + I) = \begin{bmatrix} -1 & 1 & -1 \\ 1 & 0 & 0 \\ -1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 & 1 & -1 \\ 1 & 2 & 0 \\ -1 & 0 & 2 \end{bmatrix} = \begin{bmatrix} 1 & 1 & -1 \\ 1 & 1 & -1 \\ -1 & -1 & 1 \end{bmatrix}$$

de donde (1, 1, -1) es un vector propio asociado a 2.

$$A \begin{bmatrix} 1 \\ 1 \\ -1 \end{bmatrix} = \begin{bmatrix} 2 \\ 2 \\ -2 \end{bmatrix} = 2 \begin{bmatrix} 1 \\ 1 \\ -1 \end{bmatrix}$$

Cálculo numérico.

En la práctica, los valores propios de las matrices extensas no se calculan usando el polinomio característico. Calcular el polinomio resulta muy costoso, y extraer las raíces exactas de un polinomio de grado alto puede ser difícil de calcular y expresar: el teorema de Abel-Ruffini implica que las raíces de los polinomios de grado alto (5 o superior) no pueden expresarse usándose simplemente raíces enésimas. Existen algoritmos eficientes para aproximar raíces de polinomios, pero pequeños errores en la estimación de los valores propios pueden dar lugar a errores grandes en los vectores propios. En consecuencia, los algoritmos generales para encontrar vectores propios y valores propios son iterativos. La manera más fácil es el método de las potencias: se escoge un vector aleatorio *v* y se calcula una secuencia de vectores unitarios:

$$\frac{Av}{||Av||}, \frac{A^2v}{||A^2v||}, \frac{A^3v}{||A^3v||}, \ldots$$

Esta sucesión casi siempre convergerá a un vector propio correspondiente al mayor valor propio. Este algoritmo es sencillo, pero no demasiado útil aisladamente. Sin embargo, hay métodos más populares, como la descomposición QR, que se basan en él.

Propiedades.

Multiplicidad algebraica.

La **multiplicidad algebraica** de un valor propio λ de *A* es el orden de λ como cero del polinomio característico de *A*; en otras, palabras, si λ es una de las raíces del polinomio, es el número de factores $(t - \lambda)$ en el polinomio característico tras la factorización. Una matriz $n \times n$ tiene n valores propios, contados de acuerdo con su multiplicidad algebraica, ya que su polinomio característico tiene grado n.

Un valor propio de multiplicidad algebraica 1 recibe el nombre de "valor propio simple".

Por ejemplo, se pueden encontrar exposiciones como la siguiente en artículos de teoría de matrices: "los valores propios de una matriz *A* son 4,4,3,3,3,2,2,1," lo que significa que la multiplicidad algebraica de 4 es dos, la de 3 es tres, la de 2 es dos y la de 1 es uno. Se emplea este estilo porque la multiplicidad algebraica es la clave de muchas demostraciones matemáticas en teoría de matrices.

Anteriormente se ha definido la multiplicidad *geométrica* de un vector propio como la dimensión del espacio propio asociado, o el núcleo (espacio propio de los vectores propios del valor propio nulo) de λI - *A*. La multiplicidad algebraica también puede entenderse como una dimensión: es la dimensión del *espacio propio generalizado* (1er sentido) asociado, que es el núcleo de la matriz $(\lambda I - A)^k$ para *k suficientemente grande*. Es decir, es el espacio de los *vectores propios generalizados* (1er sentido), donde un vector propio generalizado es cualquier vector que toma valor 0 sí λI - *A* se aplica suficientes veces en sucesión. Cualquier vector propio es un vector propio generalizado, así que cualquier espacio propio está contenido en el espacio propio generalizado asociado. Esto proporciona una demostración simple de que la multiplicidad geométrica es siempre menor o igual a la algebraica. El primer sentido no debe de confundirse con el problema de valores propios generalizados tal y como se muestra más adelante.

Por ejemplo:

$$A = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}.$$

Sólo tiene un valor propio λ = 1. El polinomio característico es $(\lambda - 1)^2$, así que este valor propio tiene multiplicidad algebraica 2. Sin embargo, el espacio propio asociado es el eje, que normalmente recibe el nombre de eje *x*, generado por el vector unitario $\begin{bmatrix} 1 \\ 0 \end{bmatrix}$, así que la multiplicidad geométrica es 1.

Los vectores propios generalizados pueden usarse para calcular la forma normal de Jordan de una matriz. El hecho de que los bloques de Jordan en general no son diagonales sino nilpotentes

está directamente relacionado con la distinción entre vectores propios y vectores propios generalizados.

Teoremas de descomposición para matrices generales.

El **teorema de descomposición** es una versión del teorema espectral en una clase concreta de matrices. Este teorema se explica normalmente en términos de transformación coordinada. Si U es una matriz invertible, puede verse como una transformación entre un sistema de coordenadas a otro, donde las columnas de U son las componentes de la nueva base de vectores expresados en términos de la base anterior. En este nuevo sistema las coordenadas del vector v se representan por v', que puede obtenerse mediante la relación $v' = Uv$ y, por otra parte, se tiene $v = U^{-1}v'$. Aplicando sucesivamente $v' = Uv$, $w' = Uw$ y $U^{-1}U = I$, a la relación $Av = w$ proporciona $A'v' = w'$ con $A' = UAU^{-1}$, la representación de A en la nueva base. En esta situación, se dice que las matrices A y A' son semejantes.

El teorema de descomposición declara que, si se eligen como columnas de U^{-1} n vectores propios linealmente independientes de A, la nueva matriz $A' = UAU^{-1}$ es diagonal y sus elementos en la diagonal son los valores propios de A. Si esto es posible, entonces A es una *matriz diagonalizable*. Un ejemplo de una matriz no diagonalizable es la matriz A ya mostrada:

$$A = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}.$$

Hay muchas generalizaciones de esta descomposición que pueden tratar con el caso no diagonalizable, diseñadas con diferentes propósitos:

- la descomposición de Schur declara que toda matriz es equivalente a una matriz triangular.
- la descomposición en valores singulares, $A = U\Sigma V^*$ donde Σ es diagonal con U y V matrices unitarias, los elementos de la diagonal de $A = U\Sigma V^*$ no son negativos y reciben el nombre de valores singulares de A. Esta descomposición también puede hacerse en matrices no cuadradas.
- la forma normal de Jordan, donde $A = X\Lambda X^{-1}$ y Λ no es diagonal sino diagonal por bloques. El número y tamaño de los bloques de Jordan están determinados por las multiplicidades geométrica y algebraica de los valores propios. La descomposición de Jordan es un resultado fundamental. A partir de ella se puede deducir inmediatamente que una matriz cuadrada está descrita completamente por sus valores propios, incluyendo la multiplicidad. Esto muestra matemáticamente el importante papel que desempeñan los valores propios en el estudio de matrices.
- como consecuencia inmediata de la descomposición de Jordan, cualquier matriz A puede escribirse *de forma única* como $A=S + N$ donde S es diagonalizable, N es nilpotente (por ejemplo, tal que $N^q=0$ para un cierto q), y S cumple la propiedad conmutativa del producto ($SN=NS$).

Otras propiedades de los valores propios.

El espectro es invariante bajo transformaciones semejantes: las matrices A y $P^{-1}AP$ tienen los mismos valores propios para cualquier matriz A y cualquier matriz invertible P. El espectro es también invariante a la trasposición de las matrices: A y A^{T} tienen los mismos valores propios.

Dado que una transformación lineal en espacios de dimensiones finitas es biyectiva si y sólo si es inyectiva, una matriz es invertible si y sólo si cero no es un valor propio de la matriz.

Otras consecuencias de la descomposición de Jordan son:

- una matriz es matriz diagonalizable si y sólo si las multiplicidades geométrica y algebraica coinciden para todos sus valores propios. En particular una matriz $n \times n$ que tiene n valores propios diferentes es siempre diagonalizable;
- Dado que la traza, o la suma de elementos de la diagonal principal de una matriz se preserva en la equivalencia unitaria, la forma normal de Jordan constata que es igual a la suma de sus valores propios.
- De forma similar, dado que los valores propios de una matriz triangular son las entradas de la diagonal principal su determinante es igual al producto de los valores propios (contados de acuerdo con su multiplicidad algebraica).

Algunos ejemplos de la localización del espectro de ciertas subclases de matrices normales son:

- Todos los valores propios de una matriz hermítica ($A = A^*$) son reales. Además, todos los valores propios de una matriz definida positiva son positivos.
- Todos los valores propios de una matriz antihermítica ($A = -A^*$) son imaginarios puros.
- Todos los valores propios de una matriz unitaria ($A^{-1} = A^*$) tienen valor absoluto uno.

Si A es una matriz $m \times n$ con $m \leq n$, y B es una matriz $n \times m$, entonces BA tiene los mismos valores propios de AB más $n - m$ valores propios nulos.

A cada matriz se le puede asociar una norma vectorial, que depende de la norma de su dominio, el operador norma de una matriz cuadrada es una cota superior del módulo de sus valores propios, y por tanto de su radio espectral. Esta norma está directamente relacionada con el método de las potencias para calcular el valor propio de mayor módulo. Para matrices normales, el operador norma (la norma euclídea) es el mayor módulo entre de sus valores propios.

Factorización QR

En álgebra lineal, la **descomposición** o **factorización QR** de una matriz es una descomposición de la misma como producto de una matriz ortogonal por una triangular superior. La descomposición QR es la base del algoritmo QR utilizado para el cálculo de los vectores y valores propios de una matriz.

Definición

La **descomposición QR** de una matriz cuadrada real *A* es:

$$A = QR,$$

donde *Q* es una matriz ortogonal ($Q^{T}Q = I$) y *R* es una matriz triangular superior.

Cálculo de la descomposición QR

Mediante el método de ortogonalización de Gram-Schmidt.

Recurriendo al método de ortogonalización de Gram-Schmidt, con las columnas de A como los vectores a procesar.

$$A = (\mathbf{a}_1 | \cdots | \mathbf{a}_n).$$

$$\begin{aligned}
\mathbf{u}_1 &= \mathbf{a}_1, & \mathbf{e}_1 &= \frac{\mathbf{u}_1}{\|\mathbf{u}_1\|} \\
\mathbf{u}_2 &= \mathbf{a}_2 - \operatorname{proj}_{\mathbf{e}_1} \mathbf{a}_2, & \mathbf{e}_2 &= \frac{\mathbf{u}_2}{\|\mathbf{u}_2\|} \\
\mathbf{u}_3 &= \mathbf{a}_3 - \operatorname{proj}_{\mathbf{e}_1} \mathbf{a}_3 - \operatorname{proj}_{\mathbf{e}_2} \mathbf{a}_3, & \mathbf{e}_3 &= \frac{\mathbf{u}_3}{\|\mathbf{u}_3\|} \\
&\vdots \\
\mathbf{u}_k &= \mathbf{a}_k - \sum_{j=1}^{k-1} \operatorname{proj}_{\mathbf{e}_j} \mathbf{a}_k, & \mathbf{e}_k &= \frac{\mathbf{u}_k}{\|\mathbf{u}_k\|}
\end{aligned}$$

Naturalmente, utilizamos los $\boldsymbol{a}_i$s de A para obtener:

$$\begin{aligned}
\mathbf{a}_1 &= \mathbf{e}_1 \|\mathbf{u}_1\| \\
\mathbf{a}_2 &= \operatorname{proj}_{\mathbf{e}_1} \mathbf{a}_2 + \mathbf{e}_2 \|\mathbf{u}_2\| \\
\mathbf{a}_3 &= \operatorname{proj}_{\mathbf{e}_1} \mathbf{a}_3 + \operatorname{proj}_{\mathbf{e}_2} \mathbf{a}_3 + \mathbf{e}_3 \|\mathbf{u}_3\| \\
&\vdots
\end{aligned}$$

$$\mathbf{a}_k = \sum_{j=1}^{k-1} \operatorname{proj}_{\mathbf{e}_j} \mathbf{a}_k + \mathbf{e}_k \|\mathbf{u}_k\|$$

Ahora estas ecuaciones pueden ser escritas en forma matricial de esta manera:

$$(\mathbf{e}_1 | \ldots | \mathbf{e}_n) \begin{pmatrix} \|\mathbf{u}_1\| & \langle \mathbf{e}_1, \mathbf{a}_2 \rangle & \langle \mathbf{e}_1, \mathbf{a}_3 \rangle & \ldots \\ 0 & \|\mathbf{u}_2\| & \langle \mathbf{e}_2, \mathbf{a}_3 \rangle & \ldots \\ 0 & 0 & \|\mathbf{u}_3\| & \ldots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix} :::::::::$$

El producto de cada fila con cada columna de las matrices de arriba, nos da la respectiva columna de *A* con la que comenzamos y, por tanto, dada la matriz *A*, la hemos factorizado en una matriz ortogonal *Q* (la matriz de $\boldsymbol{e}_k$s), aplicando el proceso de Gram-Schmidt, y la matriz resultante triangular superior es *R*.

Alternativamente, la matriz R puede clacularse de la siguiente manera:

Recordemos que: $Q = (\mathbf{e}_1 | \ldots | \mathbf{e}_n)$. Entonces, tenemos:

$$R = Q^T A = \begin{pmatrix} \langle \mathbf{e}_1, \mathbf{a}_1 \rangle & \langle \mathbf{e}_1, \mathbf{a}_2 \rangle & \langle \mathbf{e}_1, \mathbf{a}_3 \rangle & \ldots \\ 0 & \langle \mathbf{e}_2, \mathbf{a}_2 \rangle & \langle \mathbf{e}_2, \mathbf{a}_3 \rangle & \ldots \\ 0 & 0 & \langle \mathbf{e}_3, \mathbf{a}_3 \rangle & \ldots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix}.$$

Se observa que $\langle \mathbf{e}_j, \mathbf{a}_j \rangle = \|\mathbf{u}_j\|, \langle \mathbf{e}_j, \mathbf{a}_k \rangle = 0 \quad \text{for} \quad j > k,$ y que $QQ^T = I$, entonces $Q^T = Q^{-1}$.

Ejemplo.

Si se considera la descomposición de:

$$A = \begin{pmatrix} 12 & -51 & 4 \\ 6 & 167 & -68 \\ -4 & 24 & -41 \end{pmatrix}.$$

Se busca la matriz ortogonal *Q* tal que:

$$Q\,Q^T = I.$$

Por lo que calculamos Q mediante Gram-Schmidt como sigue:

$$U = \begin{pmatrix} \mathbf{u}_1 & \mathbf{u}_2 & \mathbf{u}_3 \end{pmatrix} = \begin{pmatrix} 12 & -69 & -58 \\ 6 & 158 & 6 \\ -4 & 30 & -165 \end{pmatrix};$$

$$Q = \begin{pmatrix} \frac{\mathbf{u}_1}{\|\mathbf{u}_1\|} & \frac{\mathbf{u}_2}{\|\mathbf{u}_2\|} & \frac{\mathbf{u}_3}{\|\mathbf{u}_3\|} \end{pmatrix} = \begin{pmatrix} 6/7 & -69/175 & -58/175 \\ 3/7 & 158/175 & 6/175 \\ -2/7 & 6/35 & -33/35 \end{pmatrix};$$

Por lo tanto, tenemos:

$$A = Q\,Q^T A = QR;$$

$$R = Q^T A = \begin{pmatrix} 14 & 21 & -14 \\ 0 & 175 & -70 \\ 0 & 0 & 35 \end{pmatrix}.$$

Considerando errores numéricos de operar con precisión finita en MATLAB, tenemos que:

$$Q = \begin{pmatrix} 0.857142857142857 & -0.394285714285714 & -0.331428571428571 \\ 0.428571428571429 & 0.902857142857143 & 0.034285714285714 \\ -0.285714285714286 & 0.171428571428571 & -0.942857142857143 \end{pmatrix};$$

$$R = \begin{pmatrix} 14 & 21 & -14 \\ 1.11022302462516 \times 10^{-16} & 175 & -70 \\ -1.77635683940025 \times 10^{-15} & -5.32907051820075 \times 10^{-14} & 35 \end{pmatrix}.$$

Mediante el uso de reflexiones de Householder.

Una *transformación de Householder* o *reflexión de Householfer* es una transformación que refleja el espacio con respecto a un plano determinado. Esta propiedad se puede utilizar para realizar la **transformación QR** de una matriz si tenemos en cuenta que es posible elegir la matriz de Householder de manera que un vector elegido quede con una única componente no nula tras ser transformado (es decir, premultiplicando por la matriz de Householder). Gráficamente, esto significa que es posible reflejar el vector elegido respecto de un plano de forma que el reflejo quede sobre uno de los ejes de la base cartesiana.

La manera de elegir el plano de reflexión y formar la matiz de Householder asociada es el siguiente:

Sea $\mathbf{x}$ un vector columna arbitrario *m*-dimensional tal que $\|\mathbf{x}\| = |\alpha|$, donde α es un escalar; (si el algoritmo se implementa utilizando aritmética de coma flotante, entonces α debe adoptar el signo contrario que x_1 para evitar pérdida de precisión).

Entonces, siendo $\mathbf{e}_1$ el vector $(1,0,...,0)^T$, y $\|\cdot\|$ la norma euclídea, se define:

$$\begin{aligned}\mathbf{u} &= \mathbf{x} - \alpha\mathbf{e}_1,\\ \mathbf{v} &= \frac{\mathbf{u}}{\|\mathbf{u}\|},\\ Q &= I - 2\mathbf{v}\mathbf{v}^T.\end{aligned}$$

v es un vector unitario perpendicular al plano de reflexión elegido. *Q* es una matriz de Householder asociada a dicho plano.

$$Qx = (\alpha, 0, \cdots, 0)^T.$$

Esta propiedad se puede usar para transformar gradualmente los vectores columna de una matriz *A* de dimensiones *m* por *n* en una matriz triangular superior. En primer lugar se multiplica *A* con la matriz de Householder Q_1 que obtenemos al elegir como vector $\mathbf{x}$ la primera columna de la matriz. Esto proporciona una matriz *QA* con ceros en la primera columna (excepto el elemento de la primera fila).

$$Q_1 A = \begin{bmatrix} \alpha_1 & \star & \dots & \star \\ 0 & & & \\ \vdots & & A' & \\ 0 & & & \end{bmatrix}$$

El procedimiento se puede repetir para A' (que se obtiene de *A* eliminando la primera fila y columna), obteniendo así una matriz de Householder Q'_2. Hay que tener en cuenta que Q'_2 es menor que Q_1. Para conseguir que esta matriz opere con Q_1A en lugar de A' es necesario expandirla hacia arriba a la izquierda, completando con un uno en la diagonal, o en general:

$$Q_k = \begin{pmatrix} I_{k-1} & 0 \\ 0 & Q'_k \end{pmatrix}.$$

Tras repetir el proceso *t* veces, donde $t = \min(m-1, n)$,

$$R = Q_t \cdots Q_2 Q_1 A$$

es una matriz triangular superior. De forma que tomando:

$$Q = Q_1 Q_2 \cdots Q_t$$

A = *QR* es una descomposición QR de la matriz *A*.

Este método tiene una estabilidad numérica mayor que la del método de Gram-Schmidt descrito arriba.

Una pequeña variación de este método se utiliza para obtener matrices semejentes con forma de Hessenberg, muy útiles en el cálculo de autovalores por acelerar la convergencia del algoritmo QR reduciendo así enormemente su coste computacional.

Ejemplo.

Vamos a calcular la descomposición de la matriz:

$$A = \begin{pmatrix} 12 & -51 & 4 \\ 6 & 167 & -68 \\ -4 & 24 & -41 \end{pmatrix}.$$

En primer lugar necesitamos encontrar una reflexión que transforme la primera columna de la matriz *A*, vector $\mathbf{a}_1 = (12, 6, -4)^T$, en $\|\mathbf{a}_1\| \, \mathbf{e}_1 = (14, 0, 0)^T$.

Usando la expresión:

$$\mathbf{u} = \mathbf{x} - \alpha \mathbf{e}_1,$$

y,

$$\mathbf{v} = \frac{\mathbf{u}}{|\mathbf{u}|},$$

en nuestro caso :

$$\alpha = 14 \quad \text{y} \quad \mathbf{x} = \mathbf{a}_1 = (12, 6, -4)^T$$

Por lo tanto:

$$\mathbf{u} = (-2, 6, -4)^T \text{ y } \quad \mathbf{v} = \frac{1}{\sqrt{14}}(-1, 3, -2)^T$$

$$Q_1 = I - \frac{2}{\sqrt{14}\sqrt{14}} \begin{pmatrix} -1 \\ 3 \\ -2 \end{pmatrix} \begin{pmatrix} -1 & 3 & -2 \end{pmatrix}$$

$$= I - \frac{1}{7}\begin{pmatrix} 1 & -3 & 2 \\ -3 & 9 & -6 \\ 2 & -6 & 4 \end{pmatrix}$$

$$= \begin{pmatrix} 6/7 & 3/7 & -2/7 \\ 3/7 & -2/7 & 6/7 \\ -2/7 & 6/7 & 3/7 \end{pmatrix}.$$

Ahora observamos:

$$Q_1 A = \begin{pmatrix} 14 & 21 & -14 \\ 0 & -49 & -14 \\ 0 & 168 & -77 \end{pmatrix},$$

con lo que ya casi tenemos una matriz triangular. Sólo necesitamos hacer cero en el elemento (3,2).

Tomando la submatriz bajo el (1, 1) y aplicando de nuevo el proceso a:

$$A' = M_{11} = \begin{pmatrix} -49 & -14 \\ 168 & -77 \end{pmatrix}.$$

Mediante el mismo método que antes obtenemos la matriz de Householder:

$$Q_2 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & -7/25 & 24/25 \\ 0 & 24/25 & 7/25 \end{pmatrix}$$

Finalmente obtenemos:

$$Q = Q_1 Q_2 = \begin{pmatrix} 6/7 & -69/175 & 58/175 \\ 3/7 & 158/175 & -6/175 \\ -2/7 & 6/35 & 33/35 \end{pmatrix}$$

$$R = Q^\top A = \begin{pmatrix} 14 & 21 & -14 \\ 0 & 175 & -70 \\ 0 & 0 & -35 \end{pmatrix}.$$

La matriz *Q* es ortogonal y *R* es triangular superior, de forma que *A* = *QR* es la descomposición QR buscada.

Mediante rotaciones de Givens.

Las descomposiciones *QR* también pueden calcularse utilizando una serie de rotaciones de Givens. Cada rotación anula (hace cero) un elemento en la subdiagonal de la matriz, formando de este modo la matriz *R*. La concatenación de todas las rotaciones de Givens realizadas, forma la matriz ortogonal *Q*.

En la práctica, las rotaciones de Givens no se utilizan en la actualidad para construir una matriz completa y realizar un producto de matrices. En su lugar, se utiliza un procedimiento de rotación de Givens, que es equivalente a la multiplicación reducida de matrices de Givens, sin el trabajo extra de manejar los elementos reducidos. El procedimiento de rotación de Givens es útil en situaciones donde sólo pocos elementos fuera de la diagonal necesitan ser anulados y es más facil de paralelizar que las transformaciones de Householder.

Ejemplo.

Calculemos la descomposición de:

$$A = \begin{pmatrix} 12 & -51 & 4 \\ 6 & 167 & -68 \\ -4 & 24 & -41 \end{pmatrix}$$

Primero, necesitamos formar una matriz de rotación tal que hagamos cero el elemento más inferior a la izquierda, $\mathbf{a}_{31} = -4$. Construimos esta matriz empleando el método de la rotación de Givens y llamamos la matriz resultante G_1. Rotamos primero el vector (6, − 4), representándolo a lo largo del eje *X*. Este vector forma un ángulo $\theta = \arctan(\frac{-4}{6})$. Creamos la matriz ortogonal de rotación de Givens, G_1:

$$G_1 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta) & \sin(\theta) \\ 0 & -\sin(\theta) & \cos(\theta) \end{pmatrix}$$

$$\approx \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0.83205 & -0.55470 \\ 0 & 0.55470 & 0.83205 \end{pmatrix}$$

Y el resultado de G_1A tiene ahora un cero en el $\mathbf{a}_{31}$elemento.

$$G_1 A \approx \begin{pmatrix} 12 & -51 & 4 \\ 7.21110 & 125.63959 & -33.83671 \\ 0 & 112.60414 & -71.83368 \end{pmatrix}$$

Procedemos análogamente con las matrices de Givens G_2 y G_3, que hacen cero los elementos subdiagonales a_{21} y a_{32}, formando una matriz rectangular R. La matriz ortogonal Q^T es formada a partir del producto en cadena de todas las matrices de Givens $Q^T = G_3G_2G_1$. Luego tenemos $G_3G_2G_1A = Q^TA = R$, y la descomposición QR es $A = QR$.

Relación con el determinante

Es posible utilizar la descomposición QR para encontrar el valor absoluto del determinante de una matriz. Suponiendo que una matriz se descompone según $A = QR$. Entonces se tiene:

$$\det(A) = \det(Q) \cdot \det(R).$$

Puesto que Q es unitaria, $|\det(Q)| = 1$. Por tanto,

$$|\det(A)| = |\det(R)| = \left|\prod_i r_{ii}\right|,$$

donde r_{ii} son los valores de la diagonal de R.

Teorema de Gerschgorin

El **teorema de Gerschgorin** nos dice que los autovalores de una matriz compleja (esto incluye también a las reales) de orden *n+1*x*n*, se encuentran en el espacio del plano complejo delimitado por la unión de los círculos D_i.

Un círculo D_i tiene el centro en el valor del elemento a_{ii} de la matriz, y su radio se obtiene sumando el resto de los elementos de la fila en valor absoluto, es decir:

$$c_i = a_{ii}$$
$$r_i = \sum_{j=1}^{n} |a_{ij}|$$

Entonces los autovalores de la matriz A se encuentran en la unión de los *n* círculos. Además, cada componente conexa de esa unión contiene tantos autovalores como círculos haya en ella, tanto círculos como autovalores contados con multiplicidad.

Ejemplo

Utilizar el Teorema de Gerschgorin para estimar los autovalores de:

$$A = \begin{bmatrix} 10 & -1 & 0 & 1 \\ .2 & 8 & .2 & .2 \\ 1 & 1 & 2 & 1 \\ -1 & -1 & -1 & -11 \end{bmatrix}.$$

Empezando con la fila 1, tomamos el elemento de la diagonal, a_{ii}, como el centro del círculo. Luego, tomamos el resto de los elementos de la fila y aplicamos la siguiente fórmula:

$$\sum_{j \neq i} |a_{ij}| = R_i$$

De esta manera obtenemos los siguientes cuatro círculos:

***D*(10,2)**
***D*(8,0.6)**
***D*(2,3)**
***D*(− 11,3)**.

Cada autovalor de *A* se encuentra dentro de uno de estos cuatro círculos.
Los autovalores son: -10.86; 1.895; 9.8218; 8.1478.

✓ Resolución del Caso Práctico nº 7:

En el presente caso obtendremos, a partir de la Matriz de Pascal, los autovalores y sus correspondientes autovectores.

Primeramente, utilizaremos el método directo de Mathcad. Luego, aplicaremos los métodos de Givens (triangulación) y Householder (diagonalización) para, posteriormente, compararlos con los resultados obtenidos en la aplicación del método directo de Mathcad.

$$\text{eigenvals}(A) = \begin{pmatrix} 92.29 \\ 5.517 \\ 1 \\ 0.181 \\ 0.011 \end{pmatrix} \qquad \text{eigenvecs}(A) = \begin{pmatrix} -0.017 & 0.243 & -0.766 & 0.571 & 0.168 \\ -0.075 & 0.481 & -0.383 & -0.559 & -0.552 \\ -0.205 & 0.611 & 0.164 & -0.253 & 0.703 \\ -0.452 & 0.413 & 0.438 & 0.518 & -0.407 \\ -0.865 & -0.407 & -0.219 & -0.173 & 0.09 \end{pmatrix}$$

Autovalores y Autovectores obtenidos con Mathcad

Obtención de autovalores y autovectores aplicando el Método de Givens

$$\text{autovaloresGivens}(A, 50) = \begin{pmatrix} 92.29 & -0 & 0 & 0 & 0 \\ 0 & 5.517 & -0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0.181 & 0 \\ 0 & 0 & 0 & 0 & 0.011 \end{pmatrix}$$

$$\text{autovectoresGivens}(A, 50) = \begin{pmatrix} 0.017 & -0.243 & 0.766 & -0.571 & 0.168 \\ 0.075 & -0.481 & 0.383 & 0.559 & -0.552 \\ 0.205 & -0.611 & -0.164 & 0.253 & 0.703 \\ 0.452 & -0.413 & -0.438 & -0.518 & -0.407 \\ 0.865 & 0.407 & 0.219 & 0.173 & 0.09 \end{pmatrix}$$

Finalmente, verificamos haciendo: $Q \cdot \Sigma \cdot Q^T = A$

$$\text{autovectoresGivens}(A, 50) \cdot \text{autovaloresGivens}(A, 50) \cdot \text{autovectoresGivens}(A, 50)^T = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 2 & 3 & 4 & 5 \\ 1 & 3 & 6 & 10 & 15 \\ 1 & 4 & 10 & 20 & 35 \\ 1 & 5 & 15 & 35 & 70 \end{pmatrix}$$

Obtención de autovalores y autovectores aplicando el Método de Householder

$$\text{autovaloresHouseholder}(A,50) = \begin{pmatrix} 92.29 & 0 & -0 & -0 & 0 \\ 0 & 5.517 & 0 & 0 & 0 \\ -0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0.181 & 0 \\ 0 & 0 & 0 & 0 & 0.011 \end{pmatrix}$$

$$\text{autovectoresHouseholder}(A,50) = \begin{pmatrix} 0.017 & -0.243 & 0.766 & -0.571 & 0.168 \\ 0.075 & -0.481 & 0.383 & 0.559 & -0.552 \\ 0.205 & -0.611 & -0.164 & 0.253 & 0.703 \\ 0.452 & -0.413 & -0.438 & -0.518 & -0.407 \\ 0.865 & 0.407 & 0.219 & 0.173 & 0.09 \end{pmatrix}$$

Finalmente, verificamos haciendo: $Q \cdot \Sigma \cdot Q^T = A$

$$\text{autovectoresHouseholder}(A,50) \cdot \text{autovaloresHouseholder}(A,50) \cdot \text{autovectoresHouseholder}(A,50)^T = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 2 & 3 & 4 & 5 \\ 1 & 3 & 6 & 10 & 15 \\ 1 & 4 & 10 & 20 & 35 \\ 1 & 5 & 15 & 35 & 70 \end{pmatrix}$$

Conclusión:

Como conclusión cabe destacar que, una manera de medir los atributos de los métodos utilizados es, como se explica anteriormente en la teoría, la de recomponer la matriz A usando la ecuación:

$$Q \cdot \Sigma \cdot Q^T = A$$

y cuantificar el error máximo respecto de los valores aij originales de A; teniendo en cuenta el factor K que es quién domina el número de loop´s para, durante el desarrollo, ir anulando los valores fuera de la diagonal principal de A.

Cuantificación de errores de cada Método:

Error en el Método de Givens:

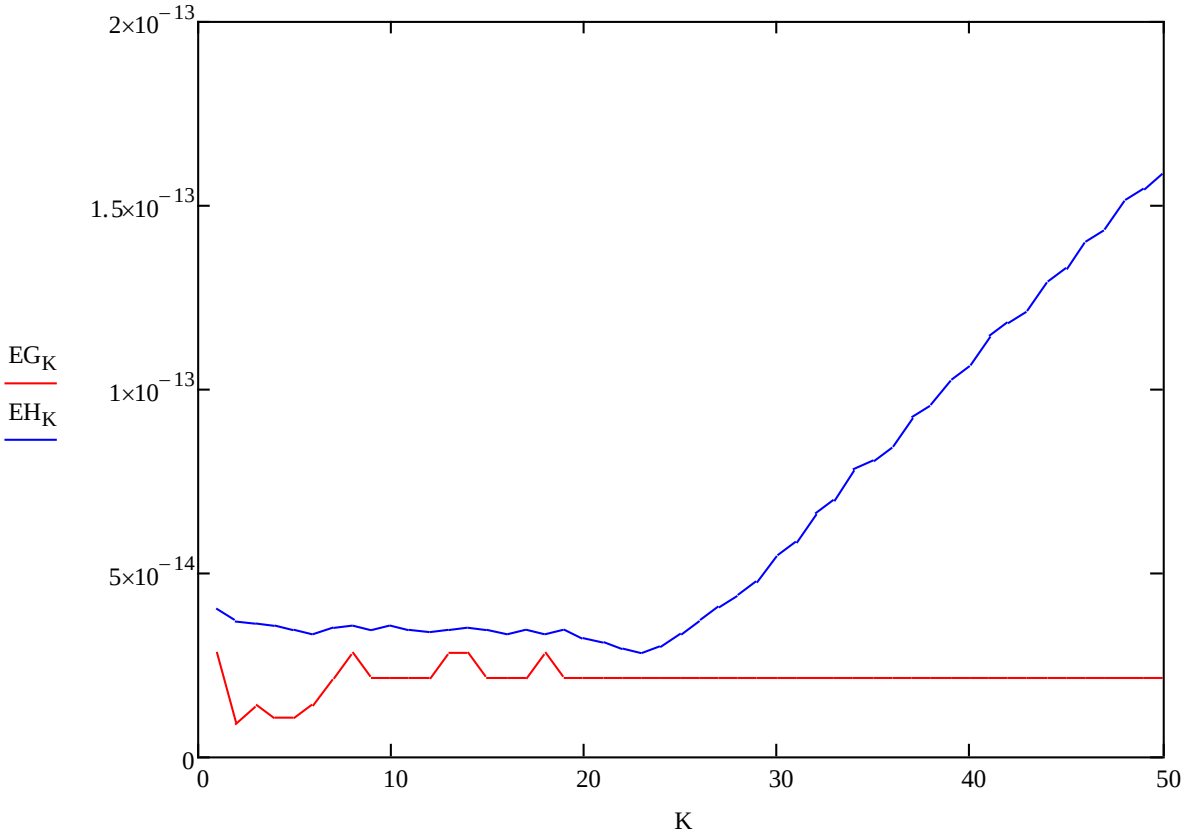
$$K := 1..50$$

$$EG_K := \max\left(\text{autovectoresGivens}(A,K) \cdot \text{autovaloresGivens}(A,K) \cdot \text{autovectoresGivens}(A,K)^T - A\right)$$

Error en el Método de Householder:

$$EH1_K := autovectoresHouseholder(A, K) \cdot autovaloresHouseholder(A, K) \cdot autovectoresHouseholder(A, K)^T$$

$$EH_K := \max\left(EH1_K - A\right)$$



Análisis del gráfico:

Antes de analizar el gráfico, debemos recordar que los mencionados métodos han sido utilizados para la resolución de la aplicación para una matriz simétrica y cuadrada.

Al comparar gráficamente los Métodos utilizados para el estudio de este caso, vemos que el Método de Givens (en rojo) muestra una convergencia rápida tendiente a un valor bajo del error máximo, mientras que el Método de Householder (en azul) oscila ampliamente y necesita mayor cantidad de iteraciones (el doble) para converger hacia un valor similar al de Givens.

El Método de Householder requiere de más memoria y realiza, aproximadamente, entre un 8% y 12% más de cuentas intermedias que el Método de Givens.

Aplicación del Teorema de Gershgorin

En primer lugar definimos el vector r que en cada componente r_i, como se explica en la parte de teoría, es la sumatoria de toda la fila menos el término diagonal:

$$r(A)^T = (4 \quad 13 \quad 29 \quad 50 \quad 56)$$

Luego, ordenamos y vectorizamos la matriz de autovalores que dan el Método de Givens y el Método de Householder:

$$\text{OrdenVD}(\text{autovaloresHouseholder}(A, 50)) = \begin{pmatrix} 0.011 \\ 0.181 \\ 1 \\ 5.517 \\ 92.29 \end{pmatrix}$$

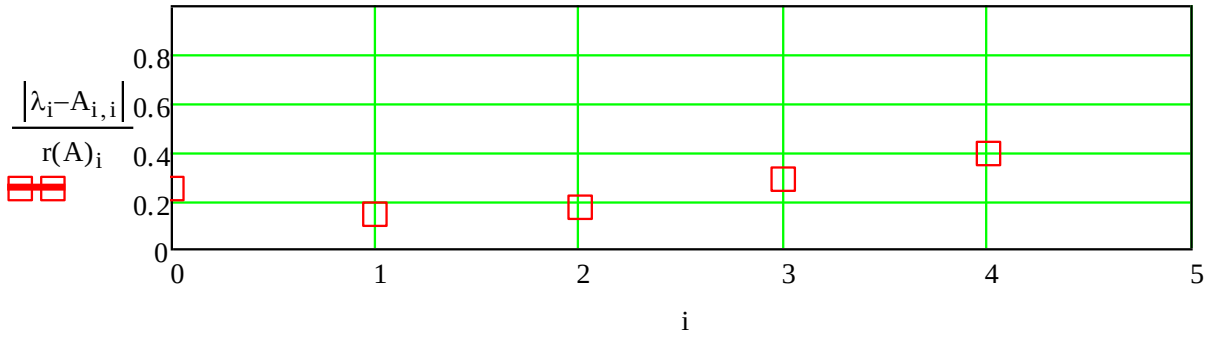
$$\text{OrdenVD}(\text{autovaloresGivens}(A, 50)) = \begin{pmatrix} 0.011 \\ 0.181 \\ 1 \\ 5.517 \\ 92.29 \end{pmatrix}$$

Ahora:

$$\lambda := \text{OrdenVD}(\text{autovaloresHouseholder}(A, 50))$$

Aplicamos el Teorema de Gershgorin: $\left|\lambda_i - A_{i,i}\right| \le r(A)_i$

Y probamos, gráficamente, la igualdad: $\dfrac{\left|\lambda_i - A_{i,i}\right|}{r(A)_i} \le 1$



Variación de los valores de la matriz A, error del 1%

Ahora nos disponemos a realizar el estudio de la variación de los valores de la matriz A, con un error del 1%; ya sea en la medición de los datos, en la transcripción de los mismos o por cualquier razón que hubiera afectado a los mismos.

Primeramente introducimos el error (ruido) de manera aleatoria en los valores de la matriz A:

$i := 0..rows(A) - 1$

$j := 0..rows(A) - 1$

$\Delta A_{i,j} := A_{i,j} \cdot (rnd(error))$

$$A = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 2 & 3 & 4 & 5 \\ 1 & 3 & 6 & 10 & 15 \\ 1 & 4 & 10 & 20 & 35 \\ 1 & 5 & 15 & 35 & 70 \end{pmatrix} \qquad \Delta A = \begin{pmatrix} 0 & 0.002 & 0.006 & 0.004 & 0.008 \\ 0.002 & 0.014 & 0.009 & 0.004 & 0.007 \\ 0.01 & 0.004 & 0.001 & 0.053 & 0.09 \\ 0.002 & 0.018 & 0.006 & 0.157 & 0.182 \\ 0.009 & 0.048 & 0.081 & 0.162 & 0.604 \end{pmatrix}$$

Para realizar este análisis supondremos que existe una función f(A), que nos permita obtener los autovalores y autovectores de la matriz A. El desarrollo en serie de Taylor en un entorno A cercano a la matriz A sería:

$$f(A + \Delta A) = f(A) + x \cdot \Delta A + O\left(\Delta A^2\right)$$

En donde,

$$f(A) = \lambda \cdot x = A \cdot x$$

$f(A + \Delta A) = A \cdot y$ (siendo y el autovector de la matriz A + ΔA)

Reemplazamos en la Serie de Taylor propuesta:

$$A \cdot y = \lambda \cdot x + \Delta A^T \cdot x + O\left(\Delta A^2\right)$$

$$y = \frac{\lambda \cdot x}{A} + \frac{\Delta A^T}{A} \cdot x + O\left(\Delta A^2\right)$$

$$y = x + \left(\frac{\Delta A^T}{A}\right) \cdot x + O\left(\Delta A^2\right)$$

Vemos que si A es pequeña, los errores no producen alteraciones en los resultados finales. No obstante, este método no es el más eficaz para trabajar con la matriz inversa de A.

$$m := 0..cols(A) - 1$$

$$y^{\langle m \rangle} := eigenvecs(A)^{\langle m \rangle} + \frac{\Delta A^{T}}{A} \cdot eigenvecs(A)^{\langle m \rangle}$$

$$v^{\langle m \rangle} := \frac{y^{\langle m \rangle}}{\left| y^{\langle m \rangle} \right|}$$

Para mostrar la veracidad del desarrollo propuesto anteriormente, realizaremos la comparación con los autovectores que se obtuvieron con Mathcad:

$$v = \begin{pmatrix} -0.017 & 0.242 & -0.745 & 0.459 & 0.161 \\ -0.075 & 0.476 & -0.381 & -0.499 & -0.285 \\ -0.205 & 0.602 & 0.137 & -0.272 & 0.181 \\ -0.451 & 0.416 & 0.462 & 0.616 & -0.36 \\ -0.865 & -0.424 & -0.258 & -0.296 & 0.854 \end{pmatrix}$$

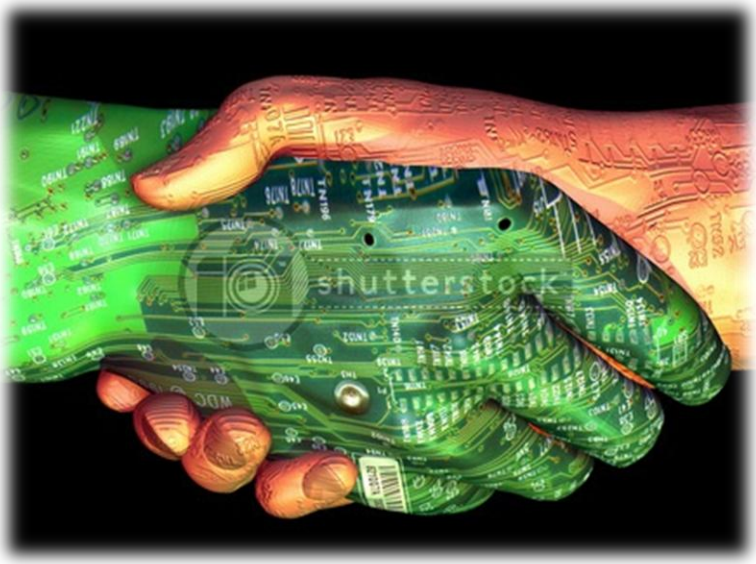
$$eigenvecs(A + \Delta A) = \begin{pmatrix} -0.017 & 0.241 & -0.75 & 0.599 & 0.222 \\ -0.074 & 0.48 & -0.397 & -0.528 & -0.626 \\ -0.205 & 0.608 & 0.139 & -0.286 & 0.672 \\ -0.451 & 0.419 & 0.459 & 0.504 & -0.323 \\ -0.865 & -0.408 & -0.223 & -0.161 & 0.059 \end{pmatrix}$$

Se observa un desfasaje en ciertos valores que se hacen menores a medida que A tiende a cero.

$$v^{T} \cdot (A + \Delta A) \cdot v = \begin{pmatrix} 92.949 & 1.516 & 2.659 & 5.9 & -55.348 \\ 1.511 & 5.585 & 0.173 & 0.603 & -3.629 \\ 2.633 & 0.155 & 1.105 & 0.359 & -1.98 \\ 5.881 & 0.571 & 0.35 & 0.65 & -3.831 \\ -55.354 & -3.615 & -1.995 & -3.85 & 34.426 \end{pmatrix} \qquad eigenvals(A + \Delta A) = \begin{pmatrix} 92.949 \\ 5.561 \\ 1.028 \\ 0.211 \\ 0.025 \end{pmatrix}$$

El desfasaje se debe a que la aproximación de Taylor tiene un error de: $O(\Delta A^{2})$

Caso Práctico nº 8



Métodos Numéricos 2009

Master Ing. Pablo DE SIMONE

Ing. Roberto HAARTH

Caso Práctico nº 8

✓ Enunciado:

Aproximar una función usando cualquier método de los discutidos durante el curso.

✓ Teoría:

Interpolación Polinómica

En análisis numérico, la **interpolación polinómica** es una técnica de interpolación de un conjunto de datos o de una función por un polinomio. Es decir, dado cierto número de puntos obtenidos por muestreo o a partir de un experimento se pretende encontrar un polinomio que pase por todos los puntos.

Definición

Dada una función *f* de la cual se conocen sus valores en un número finito de abscisas $x_0,x_1,...,x_m$, se llama **interpolación polinómica** al proceso de hallar un polinomio $p_m(x)$ de grado menor o igual a m, cumpliendo $p_m(x_k) = f(x_k),\ \forall k = 0, 1, ..., m$.

A este polinomio se le llama **Polinomio interpolador de grado m de la función f**.

Motivación del polinomio interpolador

La **interpolación polinómica** es un método usado para conocer, de un modo aproximado, los valores que toma cierta función de la cual sólo se conoce su imagen en un número finito de abscisas. A menudo, ni siquiera se conocerá la expresión de la función y sólo se dispondrá de los valores que toma para dichas abscisas.

El objetivo será hallar un polinomio que cumpla lo antes mencionado y que permita hallar aproximaciones de otros valores desconocidos para la función con una precisión deseable fijada. Por ello, para cada polinomio interpolador se dispondrá de una fórmula del error de interpolación que permitirá ajustar la precisión del polinomio.

Cálculo del polinomio interpolador

Se dispone de dos métodos generales de interpolación polinómica que permiten aproximar una función por un polinomio de grado m. Uno de los métodos es la **interpolación de Lagrange**, siendo el otro la **interpolación de Hermite**.

Interpolación de Lagrange

Sea f la función a interpolar, sean $x_0,x_1,...,x_m$ las abscisas conocidas de f y sean $f_0,f_1,...,f_m$ los valores que toma la función en esas abscisas, el polinomio interpolador de grado n de Lagrange es un polinomio de la forma:

$$\sum_{j=0}^{n} f_j l_j(x),\ n \leq m$$

donde $l_j(x)$ son los llamados polinomios de Lagrange, que se calculan de este modo:

$$l_j(x) = \prod_{i \neq j} \frac{x - x_i}{x_j - x_i} = \frac{(x - x_0)(x - x_1)...(x - x_{j-1})(x - x_{j+1})...(x - x_n)}{(x_j - x_0)(x_j - x_1)...(x_j - x_{j-1})(x_j - x_{j+1})...(x_j - x_n)}$$

Nótese que en estas condiciones, los coeficientes $l_j(x)$ están bien definidos y son siempre distintos de cero.

Disponemos de un método alternativo para calcular el polinomio interpolador de una función f dada: **el Método de las Diferencias Divididas de Newton**.

Éste método es más algorítmico y resulta sumamente cómodo en determinados casos, sobre todo cuando queremos calcular un polinomio interpolador de grado elevado.

Tomemos f una función y escribamos su polinomio interpolador de Lagrange de grado m como sigue:

$$p_m(x) = a_0 + \sum_{i=1}^{m} a_i \left(\prod_{j=0}^{i-1}(x - x_j)\right) = a_0 + a_1(x - x_0) + ... + a_m(x - x_0)(x - x_1)...(x - x_{m-1})$$

Los coeficientes a_i son las llamadas **diferencias divididas**.

Estos coeficientes se calculan mediante los datos que conocemos de la función f como sigue:

$$f[x_i, ..., x_{i+j+1}] = \frac{f[x_{i+1}, ..., x_{i+j+1}] - f[x_i, ..., x_{i+j}]}{x_{i+j+1} - x_i}$$

Como se ve en la fórmula, las diferencias divididas se calculan de modo recursivo usando coeficientes anteriores. Comenzamos el cálculo entendiendo que: $f[x_i] \equiv f(x_i)$.

Una vez hayamos realizado todos los cálculos, notaremos que hay (muchas) más diferencias divididas que coeficientes a_i. El cálculo de todos los términos intermedios debe realizarse simplemente porqué son necesarios para poder formar todos los términos finales. Sin embargo,

los términos usados en la construcción del polinomio interpolador son todos aquéllos que involucren a x_0, tal que así:

$$a_0 = f[x_0],\ a_1 = f[x_0, x_1], \ldots,\ a_i = f[x_0, x_1, \ldots, x_i]$$

Mostramos ahora una tabla mnemotécnica con las diferencias divididas de una cierta función *f* dada para construir un polinomio interpolador de grado 2:

$$\begin{array}{ccccc} f[x_0] & \searrow & & & \\ & \nearrow & f[x_0, x_1] & \searrow & \\ f[x_1] & & & & f[x_0, x_1, x_2] \\ & \searrow & & \nearrow & \\ & & f[x_1, x_2] & & \\ f[x_2] & \nearrow & & & \end{array}$$

Veamos en el ejemplo siguiente el cálculo de un polinomio interpolador de Lagrange usando los métodos mencionados:

Ejemplo: Queremos hallar el valor de la función $f(x) = e^{x+1}$ para *x* = 0.75 usando un polinomio interpolador de Lagrange de grado 2.

Para ello usamos los siguientes datos:

$$f(0) = e$$

$$f\left(\frac{1}{2}\right) = e^{\frac{3}{2}}$$

$$f(1) = e^2$$

- Usamos primero el método directo para calcular el polinomio interpolador de Lagrange. Con las condiciones dadas, los polinomios de Lagrange son:

$$l_0(x) = \frac{\left(x - \frac{1}{2}\right)(x - 1)}{\frac{1}{2}} = 2x^2 - 3x + 1$$

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

$$l_2(x) = \frac{x\left(x-\frac{1}{2}\right)}{\frac{1}{2}} = 2x^2 - x$$

- Calculamos ahora el polinomio interpolador de grado 2:

$$p_2(x) = \sum_{j=0}^{2} f_j l_j(x) = (2e - 4e^{\frac{3}{2}} + 2e^2)x^2 + (-3e + 4e^{\frac{3}{2}} - e^2)x + e$$

- Ahora evaluamos este polinomio en $x = 0.75$ para obtener un valor aproximado de $e^{1.75}$:

$$f(0.75) = f\left(\frac{3}{4}\right) = e^{\frac{7}{4}} \simeq p_2\left(\frac{3}{4}\right) \simeq 5.792377$$

- Si usamos una calculadora para efectuar el cálculo obtenemos:

$f\left(\frac{3}{4}\right) = e^{\frac{7}{4}} = 5.754602676...$, por lo que el error cometido es:

$$e_a \simeq |5.792377 - 5.754602616| = 0.037774324 \Rightarrow e_r = \frac{0.037774324}{5.754602616} = 0.006564193$$

Se trata de un error del orden del 0.66 %.

Vamos a realizar ahora la interpolación mediante el Método de las Diferencias Divididas de Newton:

- Diseñamos una tabla de Diferencias Divididas esquemática y realizamos los pertinentes cálculos para obtener los siguientes coeficientes:

$$f[x_0] = e \qquad f[x_1] = e^{\frac{3}{2}} \qquad f[x_2] = e^2$$

$$f[x_0, x_1] = 2(e^{\frac{3}{2}} - e) \qquad f[x_1, x_2] = 2(e^2 - e^{\frac{3}{2}})$$

$$f[x_0, x_1, x_2] = 2(e - 2e^{\frac{3}{2}} + e^2)$$

- Ahora debemos tomar de estos coeficientes los que necesitamos para escribir el polinomio interpolador. Recordemos que sólo utilizamos aquéllos coeficientes que

involucren a x_0. De esa manera, obtenemos el polinomio interpolador de Lagrange de grado 2:

$$p_2(x) = f[x_0] + f[x_0, x_1](x - x_0) + f[x_0, x_1, x_2](x - x_0)(x - x_1) =$$

$$= e + 2(e - e^{\frac{3}{2}})x + 2(e^{\frac{3}{2}} - e)x(x - \frac{1}{2}) =$$

$$\boxed{= (2e - 4e^{\frac{3}{2}} + 2e^2)x^2 + (-3e + 4e^{\frac{3}{2}} - e^2)x + e}$$

Conclusión:

Como podemos observar, con el Método de Interpolación de las Diferencias Divididas de Newton llegamos al mismo polinomio que con el Método de Interpolación de Lagrange, pero con menos trabajo.

Interpolación de Hermite

La interpolación de Hermite, llamada así en honor a su inventor Charles Hermite, es similar a la de Newton pero con el añadido de que ahora también conocemos los valores que toma la derivada de la función *f* en las abscisas conocidas $x_0,x_1,...,x_m$.

El Polinomio Interpolador de Hermite de grado $2m + 1$ de la función *f* es un polinomio de la forma:

$$p_{2m+1}(x) = \sum_{i=0}^{m} f_i \Phi_i(x) + \sum_{i=0}^{m} f_i' \Psi_i(x)$$

con

$$\Phi_i(x) = (1 - 2l_i'(x_i)(x - x_i))l_i^2(x)$$
$$\Psi_i(x) = (x - x_i)l_i^2(x),\ i = 0, ..., m$$

La interpolación de Hermite puede extenderse al conocimiento de las derivadas sucesivas de la función a interpolar en las abscisas tomadas, de modo que se puede obtener un polinomio cada vez más ajustado a la función real, ya que éste podrá cumplir otros requisitos como una determinada monotonía, concavidad, etc.

En este caso, estaremos hablando de interpolación de Hermite generalizada y su cálculo se llevará a cabo de forma similar a la apuntada, pero obteniendo polinomios de grado cada vez mayor debido a las sucesivas derivadas de los coeficientes $l_i(x)$.

Nota: la interpolación de Lagrange puede considerarse como un caso particular de la interpolación de Hermite generalizada (el caso en el que "conocemos" cero derivadas de *f*).

Tal y como ocurría con la Interpolación de Lagrange, para la interpolación de Hermite también disponemos una fórmula del error de interpolación que, naturalmente, tiene en cuenta factores relacionados con las derivadas de f. Más concretamente, se dispone de una fórmula del error en el caso en que la función *f* sea 2m+2 veces diferenciable en un intervalo *I* mediante la siguiente expresión:

$$f(x) - p_{2m+1}(x) = \frac{f^{(2m+2)}(\xi(x))}{(2m+2)!}(x-x_0)^2(x-x_1)^2...(x-x_m)^2$$

para $x \in I$ y donde $\xi(x) \in < x_0, x_1, ..., x_m, x >$

La diferencia esencial entre la Interpolación de Hermite y la Interpolación de Lagrange reside en el cálculo a través de la construcción de los Polinomios de Lagrange. En este caso, su cálculo es arduo, largo y complicado; por lo que el uso de las llamadas **diferencias divididas generalizadas** simplifica mucho el cálculo del polinomio interpolador.

Las diferencias divididas generalizadas se construyen de igual modo que las Diferencias Divididas de Newton, salvo que ahora necesitaremos escribir f_i tantas veces más una como derivadas de f conozcamos. Aquí sólo veremos el caso en el que conocemos la primera derivada, siendo el resto una generalización de este.

Como en la Interpolación de Lagrange, el Polinomio Interpolador de Hermite de grado $2m + 1$ se escribirá, una vez calculadas las Diferencias Divididas, de este modo:

$$p_{2m+1}(x) = f[x_0] + f[x_0, x_0](x-x_0) + ... + f[x_0, x_0, ..., x_m, x_m](x-x_0)^2...(x-x_{m-1})^2(x-x_m)$$

Nótese que, aparentemente, los coeficientes $f[x_i, x_i]$ no están bien definidos, ya que:

$$f[x_i, x_i] = \frac{f[x_i] - f[x_i]}{x_i - x_i} = \frac{0}{0}$$

Sin embargo, podemos tomar límites y escribir esta expresión así:

$$f[x_i, x_i] = \frac{f[x_i] - f[x_i]}{x_i - x_i} = \lim_{x \to x_i} \frac{f(x) - f(x_i)}{x - x_i}$$

Pero esto no es más que la definición de la derivada de *f* en el punto x_i, de modo que:

$$f[x_i, x_i] = f'(x_i)$$

Por ello, incluiremos en nuestra tabla de Diferencias Divididas los datos sobre todas las derivadas conocidas de la función a interpolar.

Interpolación segmentaria

Existen métodos de Interpolación segmentaria que nos permiten aproximar funciones de un modo eficaz. Entre ellos cabe destacar la interpolación de Taylor y la interpolación por Splines.

La Interpolación de Taylor usa el Desarrollo de Taylor de una función en un punto para construir un polinomio de grado m que se aproxima a la función dada. Tiene dos ventajas esenciales sobre otras formas de interpolación:

- Requiere sólo de un punto x_0 conocido de la función para su cálculo, si bien se pide que la función sea suficientemente diferenciable en un entorno de ese punto.
- El cálculo del Polinomio de Taylor es sumamente sencillo comparado con otras formas de interpolación polinómica:

$$p_{x_0}(x) = \sum_{i=1}^{n} \frac{f^{(i)}(x_0)}{i!}(x - x_0)^i$$

Sin embargo, en ocasiones no será deseable su uso dado que el error de interpolación puede alcanzar cotas demasiado elevadas.

Es especialmente útil para emplearse en lugar de métodos de interpolación de Hermite generalizada sobre derivadas de orden superior de la función *f*.

La Interpolación por Splines es un refinamiento de la interpolación polinómica que usa "pedazos" de varios polinomios en distintos intervalos de la función a interpolar para evitar problemas de oscilación como el llamado Fenómeno de Runge.

La idea es que agrupamos las abscisas $x_0, x_1, \ldots, x_m$ en distintos intervalos según el grado del spline que convenga emplear en cada uno. Así, un spline será un polinomio interpolador de grado n de *f* para cada intervalo. Al final, los distintos splines quedarán "unidos" recubriendo todas las abscisas e interpolando a la función.

El principal problema que presenta la interpolación por splines reside en los puntos que son comunes a dos intervalos (extremos). Por esos puntos deben pasar los splines de ambos intervalos, pero para que la interpolación sea ajustada, conviene que el punto de unión entre dos splines sea lo más "suave" posible (ej. evitar puntos angulosos), por lo que se pedirá también que en esos puntos ambos splines tengan derivada común. Esto no será siempre posible y, a menudo, se empleará otro tipo de interpolación, quizás una interpolación no-polinómica.

Polinomios de Chebyshov

En matemática, los **polinomios de Chebyshov**, nombrados en honor a Pafnuti Chebyshov, son una familia de polinomios ortogonales que están relacionados con la fórmula de De Moivre y son definidos de forma recursiva con facilidad, tal como ocurre con los números de Fibonacci o los números de Lucas. Usualmente se hace una distinción entre **polinomios de Chebyshov de primer tipo** que son denotados T_n y **polinomios de Chebyshov de segundo tipo**, denotados U_n. La letra T es usada por la transliteración alternativa del nombre *Chebyshov* como *Tchebychef* o *Tschebyscheff*.

Los polinomios de Chebyshov T_n o U_n son polinomios de grado n y la sucesión de polinomios de Chebyshov de cualquier tipo conforma una familia de polinomios.

Los polinomios de Chebyshov son importantes en la teoría de la aproximación porque las raíces de los polinomios de Chebyshov de primer tipo, también llamadas nodos de Chebyshov, son usadas como nodos en interpolación polinómica. El polinomio de interpolación resultante minimiza del problema del fenómeno de Runge y entrega una aproximación cercana del polinomio a la mejor aproximación a una función continua bajo la norma maximal. Esta aproximación conduce directamente al método de la cuadratura de Clenshaw-Curtis.

En el estudio de ecuaciones diferenciales surgen como la solución a las ecuaciones diferenciales de Chebyshov:

$$(1-x^2)\,y'' - x\,y' + n^2\,y = 0$$

y

$$(1-x^2)\,y'' - 3x\,y' + n(n+2)\,y = 0$$

para polinomios del primer y segundo tipo, respectivamente. Estas ecuaciones son casos particulares de la ecuación diferencial de Sturm-Liouville.

Definición

Los polinomios de Chebyshov de primer tipo son definidos mediante la relación de recurrencia:

$$\begin{aligned} T_0(x) &= 1 \\ T_1(x) &= x \\ T_{n+1}(x) &= 2xT_n(x) - T_{n-1}(x). \end{aligned}$$

Un ejemplo de función generatriz para T_n es:

$$\sum_{n=0}^{\infty} T_n(x)t^n = \frac{1-tx}{1-2tx+t^2}.$$

Los polinomios de Chebyshov de segundo tipo son definidos mediante la relación de recurrencia

$$U_0(x) = 1$$
$$U_1(x) = 2x$$
$$U_{n+1}(x) = 2xU_n(x) - U_{n-1}(x).$$

Un ejemplo de función generatriz para Un es:

$$\sum_{n=0}^{\infty} U_n(x)t^n = \frac{1}{1 - 2tx + t^2}.$$

Definición trigonométrica

Los polinomios de Chebyshov de primer tipo pueden ser definidos por la identidad trigonométrica:

$$T_n(x) = \cos(n \arccos x) = \cosh(n \operatorname{arccosh} x)$$

de donde:

$$T_n(\cos(\theta)) = \cos(n\theta)$$

para n = 0, 1, 2, 3,..., mientras que los polinomios de segundo tipo satisfacen:

$$U_n(\cos(\theta)) = \frac{\sin((n+1)\theta)}{\sin\theta}$$

que es estructuralmente similar al núcleo de Dirichlet.

Ese cos(*nx*) es un polinomio de grado *n*-ésimo en cos(*x*) que puede obtenerse observando que cos(*nx*) es la parte real de un lado de la fórmula de De Moivre, y que la parte real del otro lado es un polinomio en cos(*x*) y sin(*x*), en el que todas las potencias de sin(*x*) son pares, luego reemplazables vía la identidad $\cos^2(x) + \sin^2(x) = 1$.

Esta identidad es muy útil en conjunto con la fórmula generatriz recursiva, permitiendo calcular el coseno de cualquier integral múltiple de un ángulo únicamente en términos del coseno del ángulo basal. Evaluando los dos primeros polinomios de Chebyshov:

$$T_0(x) = \cos\ 0x\ = 1$$

y:

$$T_1(\cos(x)) = \cos\ (x)$$

uno puede directamente determinar que:

$$\cos(2\theta) = 2\cos\theta\cos\theta - \cos(0\theta) = 2\cos^2\theta - 1$$
$$\cos(3\theta) = 2\cos\theta\cos(2\theta) - \cos\theta = 4\cos^3\theta - 3\cos\theta$$

y así sucesivamente. Para probar trivialmente si los resultados parecen razonables, basta sumar los coeficientes en ambos lados del signo igual (es decir, fijando theta igual a cero, caso en que el coseno equivale a la unidad), obteniendo que 1 = 2 - 1 en la primera expresión y 1 = 4 - 3 en la segunda.

Un corolario inmediato es la identidad de composición:

$$T_n(T_m(x)) = T_{n\cdot m}(x).$$

Explícitamente:

$$T_n(x) = \begin{cases} \cos(n\arccos(x)), & x \in [-1,1] \\ \cosh(n\operatorname{arccosh}(x)), & x \geq 1 \\ (-1)^n\cosh(n\operatorname{arccosh}(-x)), & x \leq -1 \end{cases}$$

(sin olvidar que los cosenos hiperbólicos inversos de x y −x difieren por la constante π). A partir de un razonamiento similar al anterior, es posible desarrollar una forma cerrada para la generatriz de polinomios de Chebyshov de tercer tipo:

$$\cos(n\theta) = \frac{e^{in\theta} + e^{-in\theta}}{2} = \frac{(e^{i\theta})^n + (e^{i\theta})^{-n}}{2}$$

la cual, combinada con la fórmula de De Moivre:

$$e^{i\theta} = \cos\theta + i\sin\theta = \cos\theta + i\sqrt{1-\cos^2\theta} = \cos\theta + \sqrt{\cos^2\theta - 1}$$

entrega:

$$\cos(n\theta) = \frac{\left(\cos\theta + \sqrt{\cos^2\theta - 1}\right)^n + \left(\cos\theta + \sqrt{\cos^2\theta - 1}\right)^{-n}}{2}$$

expresión que, por supuesto, es una forma mucho más expedita para determinar el coseno de N veces un ángulo dado que iterar cerca de N veces en la forma recursiva. Finalmente, si reemplazamos cos(θ) por x, podemos escribir:

$$T_n(x) = \frac{\left(x + \sqrt{x^2 - 1}\right)^n + \left(x + \sqrt{x^2 - 1}\right)^{-n}}{2}.$$

Relación entre los polinomios de Chebyshov de primer y segundo tipo

Los polinomios de Chebyshov de primer y segundo tipo están relacionados a través de la siguientes ecuaciones:

$$\frac{d}{dx} T_n(x) = nU_{n-1}(x) \ , \ n = 1, \dots$$
$$T_n(x) = \frac{1}{2}(U_n(x) - U_{n-2}(x)).$$
$$T_{n+1}(x) = xT_n(x) - (1 - x^2)U_{n-1}(x)$$
$$T_n(x) = U_n(x) - x\,U_{n-1}(x).$$

La relación de recurrencia para la derivada de los polinomios de Chebyshov puede ser obtenida de estas relaciones:

$$2T_n(x) = \frac{1}{n+1}\,\frac{d}{dx}T_{n+1}(x) - \frac{1}{n-1}\,\frac{d}{dx}T_{n-1}(x) \ , \quad n = 1, \dots$$

Esta relación es usada en el método espectral de Chebyshov de resolución de ecuaciones diferenciales.

Equivalentemente, las dos sucesiones pueden también ser definidas a partir de un par de ecuaciones de recurrencia mutua:

$$T_0(x) = 1$$
$$U_{-1}(x) = 0$$
$$T_{n+1}(x) = xT_n(x) - (1 - x^2)U_{n-1}(x)$$
$$U_n(x) = xU_{n-1}(x) + T_n(x)$$

Estas pueden ser obtenidas desde fórmulas trigonométricas; por ejemplo, si $x = \cos\vartheta$, entonces:

$$\begin{aligned} T_{n+1}(x) &= T_{n+1}(\cos(\vartheta)) \\ &= \cos((n+1)\vartheta) \\ &= \cos(n\vartheta)\cos(\vartheta) - \sin(n\vartheta)\sin(\vartheta) \\ &= T_n(\cos(\vartheta))\cos(\vartheta) - U_{n-1}(\cos(\vartheta))\sin^2(\vartheta) \\ &= xT_n(x) - (1 - x^2)U_{n-1}(x). \end{aligned}$$

Notar que tanto estas ecuaciones como las trigonométricas adquieren una forma más simple si seguimos la convención alternativa de escribir U_n (el polinomio de grado n) como U_{n+1}.

Propiedades:

Ortogonalidad

Tanto T_n como U_n forman una familia de polinomios ortogonales. Los polinomios de primer tipo son ortogonales con respecto al peso:

$$\frac{1}{\sqrt{1-x^2}},$$

en el intervalo [−1,1], i.e. tenemos:

$$\int_{-1}^{1} T_n(x)T_m(x)\frac{dx}{\sqrt{1-x^2}} = \begin{cases} 0 & : n \neq m \\ \pi & : n = m = 0 \\ \pi/2 & : n = m \neq 0 \end{cases}$$

Esto puede ser demostrado tomando $x = \cos(\theta)$ y usando la identidad $T_n(\cos(\theta)) = \cos(n\theta)$. Similarmente, los polinomios de segundo tipo son ortogonales con respecto al peso:

$$\sqrt{1-x^2}$$

en el intervalo [−1,1], i.e. tenemos:

$$\int_{-1}^{1} U_n(x)U_m(x)\sqrt{1-x^2}\,dx = \begin{cases} 0 & : n \neq m \\ \pi/2 & : n = m \end{cases}$$

(que, al ser normalizado para formar una medida de probabilidad, es la distribución semicircular de Wigner).

Norma mínima

Dado cualquier $1 \leq n$, entre los polinomios de grado n con primer coeficiente 1, $f(x) = \frac{1}{2^{n-1}}T_n(x)$ es tal que el valor absoluto máximo en el intervalo $[-1,1]$ es mínimo. Este valor absoluto maximal es $\frac{1}{2^{n-1}}$ y $|f(x)|$ alcanza este máximo exactamente $n + 1$ veces: en -1 y 1 y los otros $n - 1$ puntos extremos de f.

Raíces y extremos

Un polinomio de Chebyshov de cualquier tipo con grado *n* tiene *n* raíces simples distintas, llamadas **nodos de Chebyshov**, en el intervalo [−1,1]. Usando la definición trigonométrica y dado que:

$$\cos\left(\frac{\pi}{2}(2k+1)\right) = 0$$

es fácil demostrar que las raíces de T_n son:

$$x_k = \cos\left(\frac{\pi}{2}\frac{2k-1}{n}\right) \ , \ k = 1, \ldots, n.$$

Similarmente, las raíces de U_n son:

$$x_k = \cos\left(\frac{k}{n+1}\pi\right) \ , \ k = 1, \ldots, n.$$

Una propiedad única de los polinomios de Chebyshov de primer tipo es que en el intervalo $-1 \leq x \leq 1$ todos los valores extremos tienen valores iguales a −1 o 1. Tanto los de primer y segundo tipo tienen extremos en los puntos de borde, dados por:

$$T_n(1) = 1$$
$$T_n(-1) = (-1)^n$$
$$U_n(1) = n + 1$$
$$U_n(-1) = (n+1)(-1)^n$$

✓ Resolución del Caso Práctico nº 8:

Para el análisis del caso propondremos dos funciones: la primera que es ***f(x) = tan(x)*** que la analizaremos con el *Método de Interpolación de Lagrange*. Y, la segunda, su función inversa. Es decir ***f(x) = atan(x)*** que la analizaremos con el *Método de Interpolación usando Spline*.

Método de Interpolación de Lagrange para evaluar: *f(x) = tan(x)*

Se desea interpolar ***f(x) = tan(x)*** en los puntos:

$x_0 = -1.5$ $\quad f(x_0) = -14.1014$

$x_1 = -0.75$ $\quad f(x_1) = -0.931596$

$x_2 = 0$ $\quad f(x_2) = 0$

$x_3 = 0.75$ $\quad f(x_3) = 0.931596$

$x_4 = 1.5$ $\quad f(x_4) = 14.1014$

Con cinco puntos, el polinomio interpolador tendrá, como máximo, grado cuatro (es decir, la máxima potencia será cuatro), al igual que cada componente de la base polinómica.

La base polinómica es:

$$\ell_0(x) = \frac{x-x_1}{x_0-x_1}\cdot\frac{x-x_2}{x_0-x_2}\cdot\frac{x-x_3}{x_0-x_3}\cdot\frac{x-x_4}{x_0-x_4} = \frac{1}{243}x(2x-3)(4x-3)(4x+3)$$
$$\ell_1(x) = \frac{x-x_0}{x_1-x_0}\cdot\frac{x-x_2}{x_1-x_2}\cdot\frac{x-x_3}{x_1-x_3}\cdot\frac{x-x_4}{x_1-x_4} = -\frac{8}{243}x(2x-3)(2x+3)(4x-3)$$
$$\ell_2(x) = \frac{x-x_0}{x_2-x_0}\cdot\frac{x-x_1}{x_2-x_1}\cdot\frac{x-x_3}{x_2-x_3}\cdot\frac{x-x_4}{x_2-x_4} = \frac{1}{243}(243-540x^2+192x^4)$$
$$\ell_3(x) = \frac{x-x_0}{x_3-x_0}\cdot\frac{x-x_1}{x_3-x_1}\cdot\frac{x-x_2}{x_3-x_2}\cdot\frac{x-x_4}{x_3-x_4} = -\frac{8}{243}x(2x-3)(2x+3)(4x+3)$$
$$\ell_4(x) = \frac{x-x_0}{x_4-x_0}\cdot\frac{x-x_1}{x_4-x_1}\cdot\frac{x-x_2}{x_4-x_2}\cdot\frac{x-x_3}{x_4-x_3} = \frac{1}{243}x(2x+3)(4x-3)(4x+3)$$

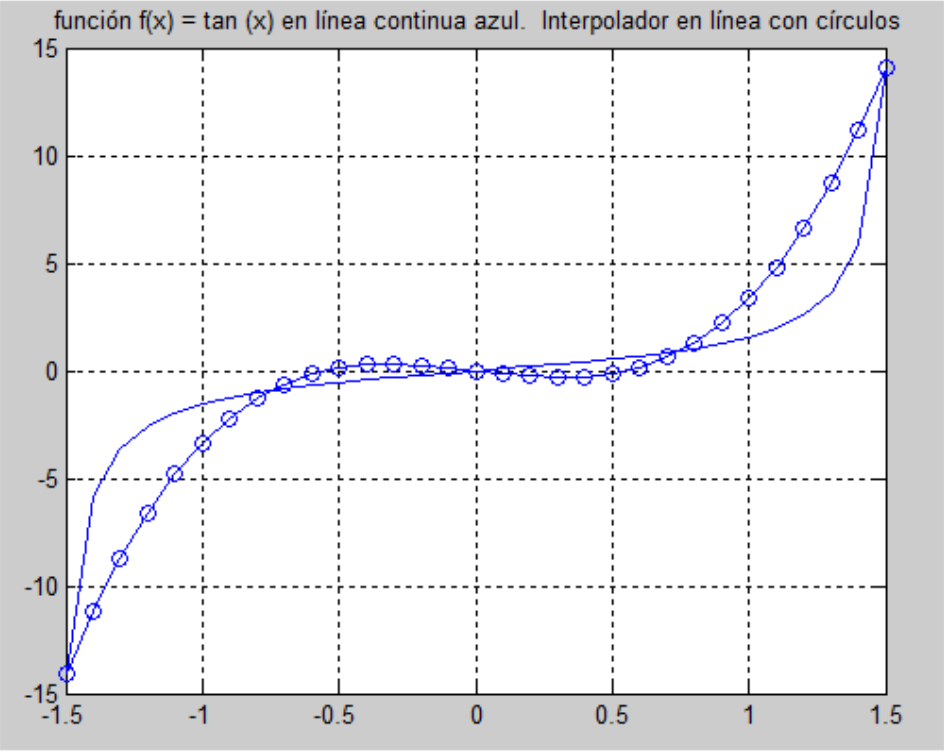
Así, el polinomio interpolador se obtiene simplemente como la combinación lineal entre los $\ell_i(x)$ y los valores de las abscisas:

$$\frac{1}{243}\big(f(x_0)x(2x-3)(4x-3)(4x+3) - 8f(x_1)x(2x-3)(2x+3)(4x-3)$$

$$+f(x_2)(243-540x^2+192x^4)-8f(x_3)x(2x-3)(2x+3)(4x+3)$$
$$+f(x_4)x(2x+3)(4x-3)(4x+3)\big)$$
$$=-1.47748x+4.83456x^3.$$

```
% Matlab grafica
% Grafico de la función natural
x= (-1.5:.1:1.5)'
y=tan(x)
%hold on
plot (x,y)
%title('Datos de Población años 1900-1990');
%ylabel('Millones de habitantes');
%hold off
grid on
%grafico con interpolador lagrange
t= (-1.5:.1:1.5)'
p=(-1.4778*t)+(4.8345*t.^3)
%p=(2.08908*t.^4)+(3.4818*t.^5)
hold on
plot (t,p,'bo-')
hold off
title(' función f(x) = tan (x) en línea continua azul.  Interpolador en línea con círculos')
```

Graficamos:



La función ***f(x) = tan (x)*** y su interpolador

Conclusión:

Debido a que el polinomio interpolador de Lagrange se ajusta a todos los puntos que le son especificados (datos), en aquellas situaciones con una gran cantidad de datos se obtiene un polinomio de grado muy alto, lo cual normalmente resulta poco práctico. Es por esta razón que, en la práctica, no es común utilizar este método sino que se prefiere ajustar los datos lo mejor posible utilizando un polinomio de menor grado, incluso si este polinomio no pasa por ninguno de los puntos que le son especificados (pero ajusta en forma aproximada siguiendo algún criterio de optimalidad).

Otro problema del polinomio interpolador de Lagrange es que, a medida que crece el grado del polinomio interpolador, se percibe una creciente variación entre puntos de control consecutivos, lo que produce que la aproximación entre dos puntos continuos sea muy distinta a la que uno esperaría. Esto se conoce como *overfitting* (*sobre fiteo*).

A pesar de estos problemas, el polinomio interpolador de Lagrange es muy simple de implementar y tiene interés teórico más que práctico por su sencillez.

Método de Interpolación usando Splines para evaluar: *f(x) = atan(x)*

La función *spline de MATLAB* se utiliza para calcular el spline s(x) directamente.

Pasos para la siguiente evaluación:

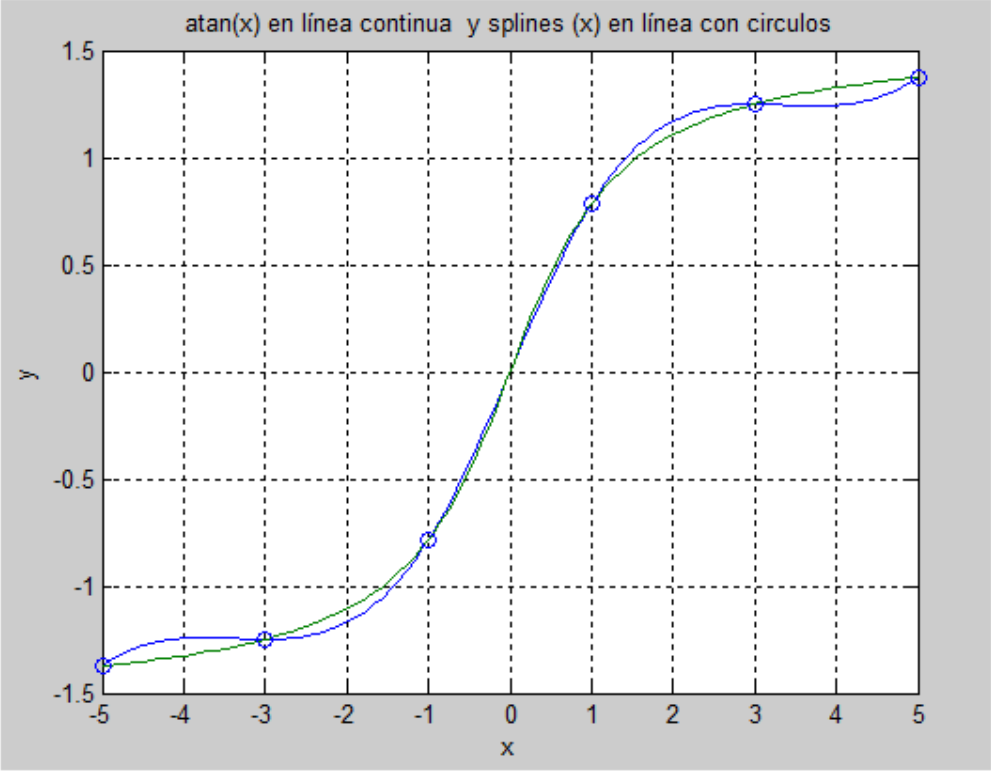
1. Primero, dividiremos el intervalo [-5,5] en seis sub-intervalos.
2. Luego, evaluamos la función *atan(x)* (tangente inversa) en los puntos de la partición.
3. Y, finalizando, construimos el spline s(x) que interpola es estos puntos.

Función *spline*:

```
%
% Divide el intervalo [-5,5] en cinco pedazos generando así seis puntos
%
x=linspace(-5,5,6);
%
% Evalúa la función atan en los puntos de la partición
%
y=atan(x);
%
% Calcula la representación del spline que interpola a los datos
%
pp=spline(x,y);
%
% Calcula 100 puntos en el intervalo [-5,5] para las graficas
%
z=linspace(-5,5,100);
```

```
% Evalua el spline y la función atan en los 100 puntos
%
sval=ppval(pp,z);
y1=atan(z);
%
% Grafica el spline, atan, y los puntos de interpolaci&oacuten en un mismo
% sistema de coordenadas
%
plot(z,sval,z,y1,x,y,'+')
xlabel('x');
ylabel('y');
title('atan(x) en violeta y s(x) en amarillo')
```

Graficamos:



Conclusión:

El principal problema que presenta la interpolación por splines reside en los puntos que son comunes a dos intervalos (extremos). Por esos puntos deben pasar los splines de ambos intervalos. Pero, para que la interpolación sea ajustada, conviene que el punto de unión entre dos splines sea lo más "suave" posible (es decir, evitar puntos angulosos), por lo que se pedirá también que en esos puntos ambos splines tengan derivada común. Esto no será siempre posible y, a menudo, se empleará otro método de interpolación.

Caso Práctico nº 9

Métodos Numéricos 2009

Master Ing. Pablo DE SIMONE

Ing. Roberto HAARTH

Caso Práctico nº 9

✓ Enunciado:

El siguiente caso plantea estudiar/investigar, el siguiente sistema de ecuaciones diferenciales:

$$\left.\begin{aligned} \frac{dL(t)}{dt} &= 2 * L(t) - \alpha * L(t) * Z(t) \\ \frac{dZ(t)}{dt} &= -Z(t) + \alpha * L(t) * Z(t) \end{aligned}\right\} \quad \begin{aligned} L(t) &= Población\ de\ liebres \\ Z(t) &= Población\ de\ zorros \\ L_{(0)} &= L_0 \quad ; \quad Z_{(0)} = Z_0 \end{aligned}$$

a. Investigar usando varios valores de α. ($\alpha > 0$)
b. ¿Hay soluciones periódicas?

✓ Teoría:

Modelización de Ecuaciones Diferenciales

¿Qué es un modelo? Es el proceso de representación del "mundo real" en términos matemáticos. Los modelos matemáticos que estudiamos son sistemas que evolucionan con el tiempo, pero con frecuencia, también están supeditados a otras variables. Una vez elaborado el modelo, debemos comparar las predicciones de éste con los datos del sistema. Si el modelo y el sistema concuerdan, tenemos confianza en que las hipótesis hechas al crear el modelo son razonables y que podemos usarlo para hacer predicciones. Si no concuerdan, entonces debemos estudiar y mejorar nuestras suposiciones.

Los tipos de predicciones que son razonables dependen de nuestras hipótesis.

Si nuestro modelo se basa en reglas precisas, como las leyes de Newton sobre el movimiento, o las de interés compuesto, entonces podemos usarlo para hacer predicciones cuantitativas muy exactas.

¿Cómo se construye un modelo?

Los pasos básicos para elaborar un modelo son:

1. Establezca claramente las hipótesis en las cuáles se basará el modelo. Estas deben describir las relaciones entre las cantidades por estudiarse.
2. Defina completamente las variables y los parámetros que se utilizarán en el modelo.

3. Use las hipótesis formuladas en el punto 1. Para obtener ecuaciones que relacionen las cantidades del punto 2.

Las cantidades en nuestros modelos se agrupan en tres categorías: *la variable independiente, las variables dependientes y los parámetros.*

En ecuaciones diferenciales *la variable independiente casi siempre es el tiempo. Las variables dependientes son cantidades que son funciones de la variable "independiente".* Por ejemplo, en física, "la posición es una función del tiempo". Es posible enunciar vagamente el objetivo del modelo expresado en términos de una ecuación diferencial, por ejemplo: *"describa el comportamiento de la variable dependiente, conforme cambie la variable independiente"*. Podemos preguntar si la variable dependiente aumenta o disminuye, o si oscila, o tiende a un límite.

Los parámetros son cantidades que no cambian con el tiempo (o con la variable dependiente) pero que pueden ajustarse (por causas naturales, o por un experimento científico). Por ejemplo, si estamos analizando la cantidad de ozono en las capas superiores de la atmósfera, entonces la velocidad con que se libran los fluorocarbonos de los refrigeradores, es un parámetro.

En el punto 3., formulamos las ecuaciones. La mayor parte de los modelos que consideremos son expresados como ecuaciones diferenciales. En otras palabras, esperamos encontrar derivadas en nuestras ecuaciones. Ponga atención a frases como *"razón de cambio de…", "tasa de crecimiento de…".* Ya que razón de cambio es sinónimo de derivada. Por supuesto, ponga atención también a *"velocidad"* (derivada de la posición) y *"aceleración"* (derivada de la velocidad) en modelos de física.

Una importante regla empírica que usamos al formular modelos es: *Simplifique siempre que pueda el álgebra*. Por ejemplo, al modelar la velocidad *v* de un gato al caer de un edificio alto, podemos suponer que, la resistencia del aire crece al aumentar la velocidad del gato:

$$\text{resistencia del aire} = \begin{cases} kv \leftarrow (en\ el\ caso\ de\ la\ nieve) \\ kv^2 \leftarrow (en\ el\ caso\ del\ gato) \end{cases}$$

Veamos otro ejemplo interesante. Mediante la adopción de las prácticas babilónicas de la medición cuidadosa y las observaciones detalladas, los antiguos griegos, trataron de comprender la naturaleza a partir del análisis lógico. Los convincentes argumentos de Aristóteles, de que el mundo no era plano sino esférico, llevaron a los científicos de aquella época a considerar la siguiente pregunta: *¿A qué equivale la circunferencia de La Tierra?* Y resulta asombroso que Eratóstenes haya logrado obtener una respuesta bastante precisa para este problema sin tener que salir de la antigua ciudad de Alejandría. Su método implicaba ciertas suposiciones y simplificaciones: La Tierra es una esfera perfecta, los rayos del sol viajan en trayectorias paralelas, la ciudad de Siena se encuentra a 5000 estadios (1 estadio = 220 yardas = 201.16 metros) exactamente hacia el sur de Alejandría, etc. Con estas idealizaciones, Eratóstenes, creó un contexto matemático en el cuál pudieron aplicarse los principios de la geometría.

Prueba del modelo

Antes de verificar un modelo, se deben tener en cuenta los siguientes interrogantes:

- ¿Son razonables las hipótesis?
- ¿Son correctas las dimensiones físicas de las variables?
- ¿Es el modelo internamente consistente (en el sentido de que las ecuaciones no se contradigan entre si)?
- ¿Las ecuaciones pertinentes poseen solución?
- ¿Qué tan fácil resulta obtener soluciones?
- ¿Proporcionan las soluciones una respuesta al problema estudiado?

Tipos de modelos

Existe tantos modelos como se quieran construir. A modo de ejemplo citaremos algunos:

- Modelo del crecimiento ilimitado de la población.
- Solución analítica del modelo poblacional.
- Modelo logístico de la población.
- Análisis cualitativo del modelo logístico.
- Sistema depredador-presa.
- Modelo de ahorro.
- Modelo: Mezcla en un tanque.

✓ Resolución del Caso Práctico nº 9:

En nuestro caso particular nos interesa analizar el ***"Sistema o Modelo Depredador-Presa".***

Modelo Depredador-Presa

Ninguna especie vive aislada y las interacciones entre especies proporcionan algunos de los modelos más interesantes para estudiar. Presentamos un sistema *depredador-presa* de ecuaciones diferenciales donde una especie "se come" a otra. La diferencias más obvia entre éste y los otros modelos es que tenernos dos cantidades que dependen del tiempo. Nuestro modelo tiene entonces dos variables dependientes que son ambas funciones del tiempo. En este caso llamaremos a la presa "liebre" y a los depredadores "zorros", y denotaremos la presa por L y a los depredadores por Z. Las hipótesis de nuestro modelo son:

- Si no hay zorros presentes, las liebres se reproducen a una tasa proporcional a su población y no les afecta la sobrepoblación.
- Los zorros se comen a las liebres, y la razón por la cual las liebres son devoradas, es proporcional a la tasa por la cual los zorros y las liebres interactúan.
- Sin liebres que comer, la población de zorros declina a una razón proporcional a ella misma.
- La tasa de nacimiento de los zorros va en proporción al número de liebres comidas por los zorros, que pos la segunda hipótesis, es proporcional a la razón por la cual los zorros y liebres interactúan.

Para formular este modelo en términos matemáticos, necesitamos cuatro parámetros adicionales a nuestra variable independiente t y a nuestras dos variables dependientes Z y L. Los parámetros son:

$\alpha =$ coeficiente de la razón de crecimiento de liebres.
$\beta =$ constante de proporcionalidad que mide el número de interacciones liebre /zorros en la que las liebres son devoradas.
$\gamma = coeficiente\ de\ la\ razón\ de\ muertes\ de\ zorros.$
$\delta = constante\ de\ proporcionalidad\ que\ mide\ el\ beneficio, a\ la\ población\ de$ zorros, por una liebre devorada.

Cuando formulamos nuestro modelo, seguimos la convención de que, $\alpha, \beta, \gamma, \delta$ son todas positivas.

Nuestras primera y tercera hipótesis, anteriores, son similares a la que plantea el modelo de crecimiento ilimitado. En consecuencia de ello dan términos de la forma $\alpha * L$ en la ecuación para $\frac{dL}{dt}$ y $-\gamma * Z$ (ya que la población de zorros declina) en la ecuación para $\frac{dZ}{dt}$.

La razón por la que las liebres son devoradas es proporcional a la razón de interacción entre los zorros y las liebres, por lo que necesitamos un término que modele la razón de interacción de ambas poblaciones. Es decir, que crezca si L ó Z aumenta, pero que desaparezca si $L = 0$ ó $Z = 0$. Una notación que incorpora esas hipótesis es $L * Z$. Modelemos así los efectos de las interacciones liebre/zorro sobre $\frac{dL}{dt}$, por medio de un enunciado de la forma $-\beta * L * Z$. La cuarta hipótesis da un término similar en la ecuación para $\frac{dZ}{dt}$. En este caso, cazar liebres ayuda a los zorros, por lo que añadimos un término de la forma $\delta * L * Z$.

Al plantear esas hipótesis, obtenemos el siguiente modelo:

$$\begin{cases} \dfrac{dL}{dt} = \alpha * L - \beta * L * Z \\ \dfrac{dZ}{dt} = -\gamma * Z + \delta * L * Z \end{cases}$$

Considerada juntas, este par de expresiones se llaman ***sistema de primer orden*** de ecuaciones diferenciales ordinarias. Se dice que el sistema es ***acoplado*** porque las razones de cambio de *L* y *Z* dependen tanto de *L* como de *Z*. Una solución para este sistema de ecuaciones es, a diferencia de nuestro modelos previo, un par de funciones, *L(t)* y *Z(t)*, que describen las poblaciones de liebres y zorros como funciones del tiempo. Como es acoplado, no podemos determinar cada una de esas funciones en forma aislada. Más bien, debemos resolver ambas ecuaciones en forma simultánea.

Para nuestro caso particular de estudio vemos que:

$\alpha = 2 \longrightarrow$ Es decir que, la razón de crecimiento de liebres es igual a 1.

$\gamma = 1 \longrightarrow$ Es decir que, la razón de crecimiento de zorros es igual a 1.

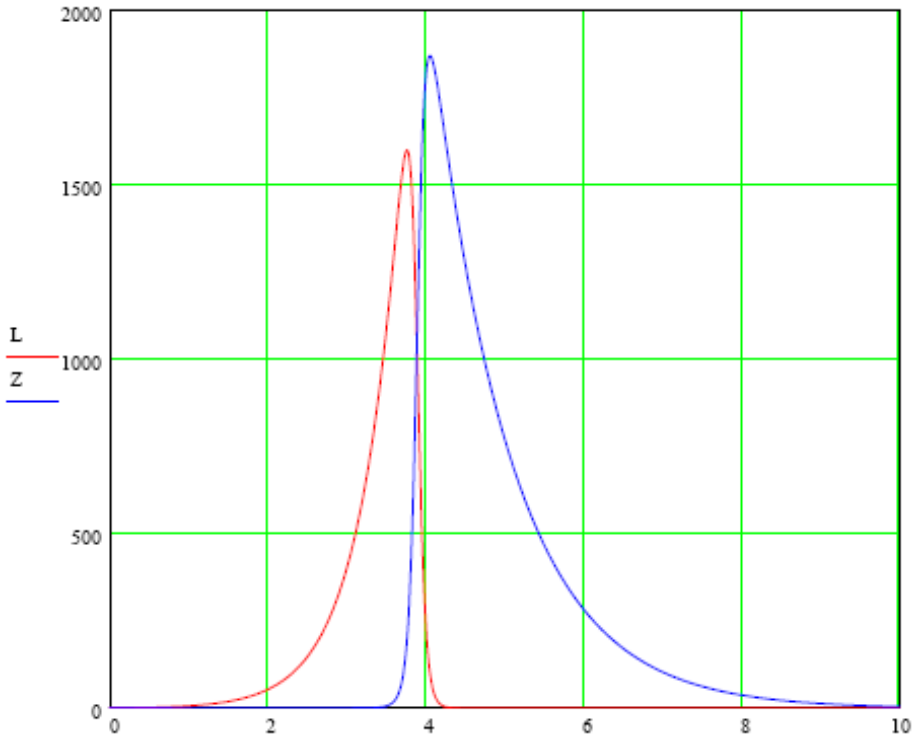
$\beta = \delta \longrightarrow$ Es decir que, la proporcionalidad que mide las interacciones entre liebres y zorros cuando las liebres son devoradas es igual a la proporcionalidad que mide el beneficio, para la población de zorros, por cada liebre devorada. En nuestro caso particular, como $\alpha = 1$, $\gamma = 1$ y siendo $\beta = \delta$, llamamos a ambas constantes (β y δ), directamente α.

Entonces, nuestro sistema de ecuaciones queda definido:

$$\begin{cases} \frac{dL(t)}{dt} = 2 * L(t) - \alpha * L(t) * Z(t) \\ \frac{dZ(t)}{dt} = -Z(t) + \alpha * L(t) * Z(t) \end{cases}$$

Ahora, estudiamos el sistema para diferentes valores de α y graficamos:

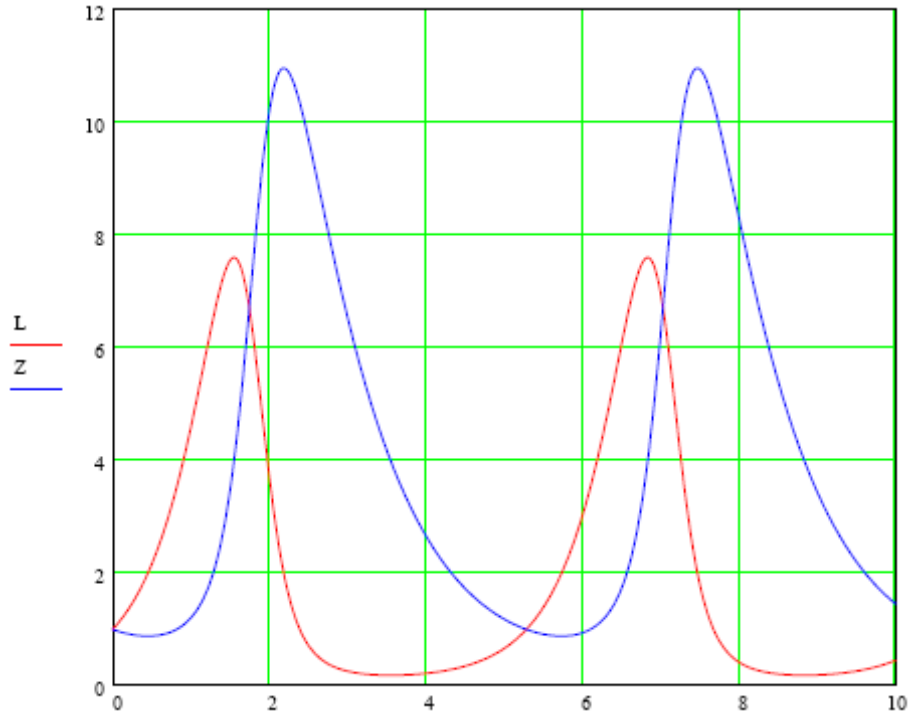
$\boldsymbol{\alpha = 0.01}$



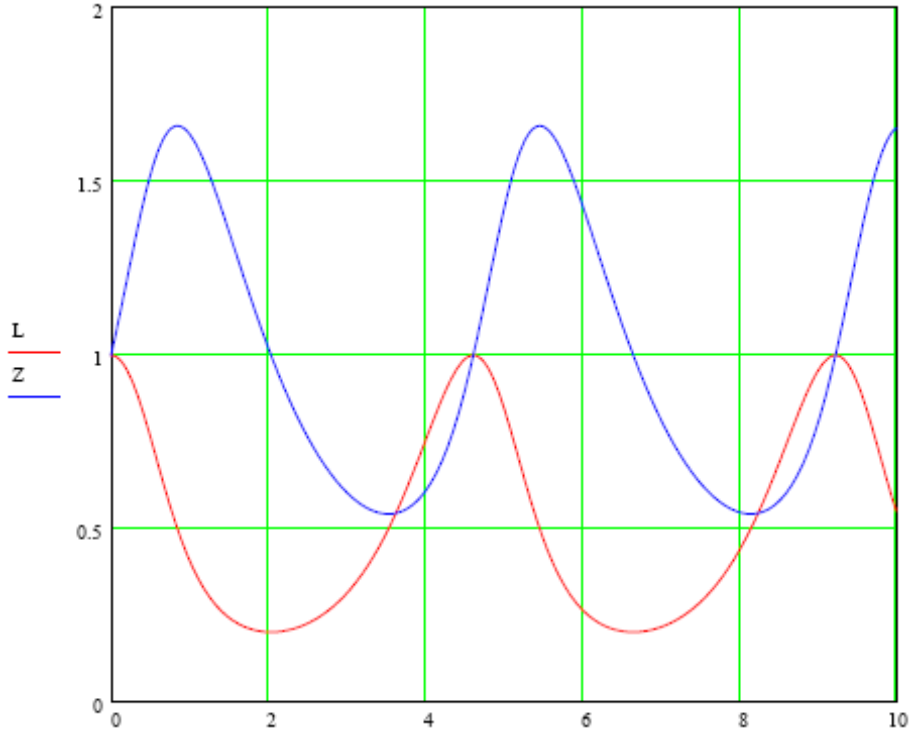
Conclusión y Análisis del gráfico:

Se observa que, para pequeños valores de α, como por ejemplo el caso de $\boldsymbol{\alpha = 0.01}$, la población de liebres crece rápidamente. También, con algo más de tiempo, ocurre lo mismo con la población de zorros. Esto provoca un decrecimiento de ambas poblaciones ya que, los zorros se comen "todas" las liebres y después no tienen que comer.

$\alpha = 0.5$



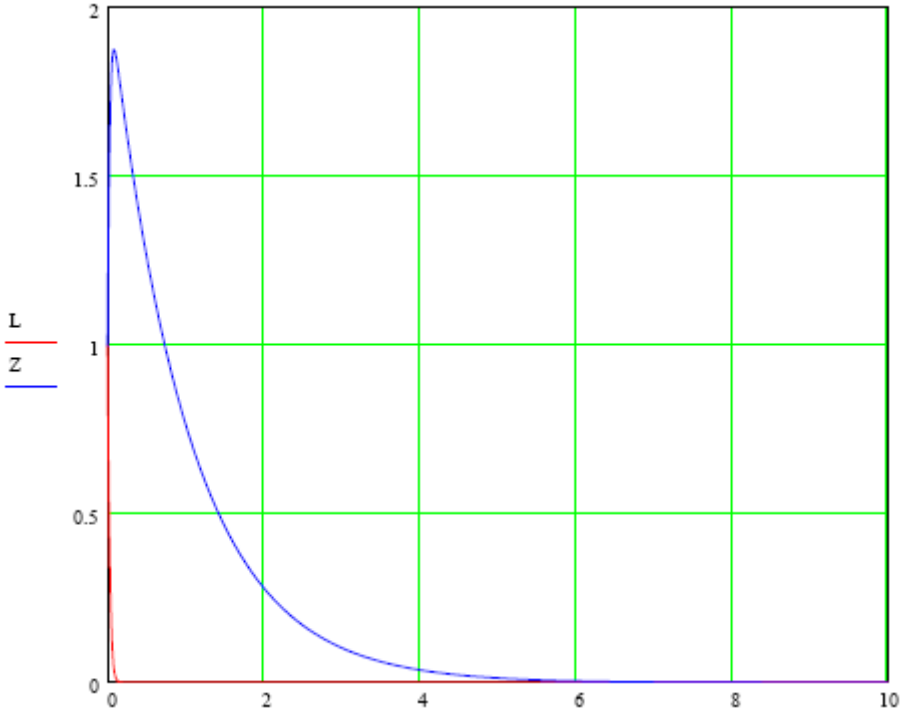
$\alpha = 2$



Conclusión y Análisis del gráfico:

Se observa que, para valores no muy pequeños y no muy grandes de α, como por ejemplo el caso de $\mathbf{0.5} < \alpha < \mathbf{2}$, se presentan oscilaciones entre ambas poblaciones. De todas maneras, y al analizar esto para la vida cotidiana de liebres y zorros, se prefiere que $\mathbf{1.35} < \alpha < 1.6$. Esto es, se presentan oscilaciones pero, de alguna manera, se respeta o mantiene un equilibrio.

$\boldsymbol{\alpha = 25}$



Conclusión y Análisis del gráfico:

Se observa que, para grandes valores de α, como por ejemplo el caso de $\boldsymbol{\alpha = 25}$, tanto la población de liebres como la población de zorros crecen rápidamente; aunque esta última lo hace algo de tiempo después. El rápido crecimiento de ambas poblaciones provoca también una rápida disminución en ambas poblaciones ya que, los zorros se comen "todas" las liebres y después no tienen que comer.

BIBLIOGRAFIA DE REFERENCIA

[1]. Scheid, Francis. Di Costanzo Lorencez, Rosa Elena. *Métodos numéricos.* México, D.F. McGraw-Hill. 1991.
[2]. Luthe, Rodolfo. Olivera, AntonioSchutz, Fernando. *Métodos numéricos*. México, D.F. Limusa. 1991.
[3]. Hoffman, Joe D. *Numerical methods for engineers and scientists.* 2a. ed. New York. CRC Press. 2001.
[4]. Atkinson, L. V. Harley, P. J. *Introducción a los métodos numéricos con pascal*. Buenos Aires. Addison-Wesley. 1987.
[5]. McCracken, Daniel D. Dorn, William S. *Métodos numéricos y programación fortran*. México. D.F. Limusa-Wiley. 1966.
[6]. Zill, Dennis G. *Ecuaciones diferenciales con aplicaciones de modelado*. México, D.F. Thomson. 2006
[7]. Luthe, Rodolfo. Olivera, AntonioSchutz, Fernando. *Métodos numéricos*. México, D.F. Limusa. 1991.
[8]. Chapra, Steven C. Canale, Raymond P. *Métodos numéricos para ingenieros: con* aplicaciones *en computadoras personales*. Naucalpan de Juárez. McGraw-Hill. 1988.
[9]. Nakamura, Shoichiro. *Métodos numéricos aplicados con software*. Naucalpan de Juárez. Prentice-Hall. 1992.
[10]. Gil Ruiz, Antonio Javier. *Métodos numéricos para el diseño de estructuras traccionadas*: *membranas y redes de cables*. Granada. Miguel A. Losada. 2001.
[11]. Mathews, John H. Fink, Kurtis D. *Métodos numéricos con matlab.* 3a. ed. Madrid. Prentice-Hall. 2000.
[12]. Chapra, Steven C. Canale, Raymond P. *Métodos numéricos para ingenieros*. 3a. ed. México. D.F. McGraw-Hill. 1999.
[13]. Chapra, Steven C. Canale, Raymond P. *Métodos numéricos para ingenieros*. 4a. ed. México. D.F. McGraw-Hill, , c2003.
[14]. Chapra, Steven C. Canale, Raymond P. *Métodos numéricos para ingenieros*. 5a ed. México. D.F. McGraw-Hill/Interamericana. 2007.
[15]. Nieves Hurtado, Antonio. Domínguez Sánchez, Federico C. *Métodos numéricos aplicados a la ingeniería*. 2a. ed. México, D.F. CECSA. 2002.
[16]. García Merayo, Félix. Nevot Luna, Antonio. *Análisis numérico*. Madrid. Paraninfo. 1992.
[17]. Hildebrand, Francis B. *Introduction to numerical analysis*. 2a. ed. New York. McGraw-Hill. 1974.
[18]. Mason, J. C. *Cálculo Numérico: teoría, problemas y aplicaciones en basic*. Madrid. Anaya-Multimedia. 1983.
[18]. Scheid, Francis. Di Costanzo Lorencez, Rosa Elena. *Métodos numéricos*. México, D.F. McGraw-Hill. 1991.
[19]. Burden, Richard L. Faires, J. Douglas. *Análisis numérico.* México, D.F. Iberoamerica. 1985.

[20]. Courant, Richard. John, Fritz. *Introducción al cálculo y al análisis matemático – I*. México. D.F. Limusa. 1993.
[21]. Kincaid, David. Cherey, Ward. *Análisis numérico*. Buenos Aires. Addison-Wesley. 1994.
[22]. De Boor, Carl. *Elementary numerical analysis an algorithmic approach*. 3a. ed. New York. McGraw-Hill. 1980.
[23]. Bulirsch, R. *Introduction to numerical analysis*. 2a. ed. New York. Springer-Verlag. 1980. CDROM.
[24]. Schmitt, Klaus. Thomson, Russell C. *Non-linear analysis and differential equations and introduction*. Utah. [s.n.]. 1998.CDROM.
[25]. Gerald, Curtis F. Wheatley, Patrick O. *Análisis numérico con aplicaciones*. 6a. ed. Naucalpán de Juárez. Pearson Educación. 2000.
[26]. Etter, Delores M. *Solución de problemas de ingeniería con matlab*. 2a. ed. Naucalpan de Juárez. Prentice Hall. 1998.
[27].Forsythe G.E. Malcolm M.A.Moler C.B. *Computer methods for matematical computations*. Pretince-Hall. 1977.
[28].Higham Nicholas. *Accuray and stability of numerical algoritms*. 2a ed. Siam. 2002.
[29].Ortega James. *Numerical Analisys a Second Course*. Siam. 1990.
[30].Press W.Teukolsky S. Vetterling W. Flannery B. *Numerical Recipes in Fortran 77*.vol1. Press Syndicate of the University of Cambridge. 1992.
