CÁLCULO NUMÉRICO Y COMPUTACIÓN.

NOTAS DE TEORÍA.

Las Notas de Teoría presentan desarrollos teóricos con demostraciones para los distintos temas; incluyendo siempre algunos ejemplos.

El objetivo de estas Notas de Teoría es indicar el nivel de profundidad que se pretende los alumnos alcancen. No buscan suplantar los muy buenos textos citados en la bibliografía, sino sólo indicar el enfoque de la Cátedra.

Los temas desarrollados son los siguientes:

Motivación y procesos de modelación en ingeniería.
Algoritmia. Conceptos Básicos
Solución Numérica de Raíces de Ecuaciones No Lineales
Sistemas de Ecuaciones Lineales
Valores y Vectores Propios
Interpolación y Aproximación de funciones discretas
Integración Numérica
Derivación Numérica, con aplicación a la solución de Ecuaciones Diferenciales con Valores de Contorno
Solución Numérica de Ecuaciones Diferenciales Ordinarias con Valores Iniciales

El ordenamiento elegido está según el cronograma del año 2016.

1 MOTIVACIÓN

En diversas actividades de nuestros tiempos, son frecuentes las siguientes **necesidades** que en general están vinculadas con aspectos éticos, sociales, políticos y económicos.

1.1 Evaluar Riesgo y Durabilidad

- **Actividades relacionadas con Diseño**
  Casa, Edificios, Vehículos
  Centrales Hidroeléctricas, Térmicas o Nucleares
  Organos Artificiales: implantes odontológicos, bombas de sangre, válvulas cardíacas, etc.
  CAE-CAD-CAM-CFD
- **Actividades relacionadas con Inversiones**
  Administradores de Fondos de Inversión
  Comportamiento de Mercados de Valores
  Economías Emergentes

1.2 Desarrollo de Habilidades

En general para mejorar y hasta asegurar el correcto desempeño en determinadas actividades

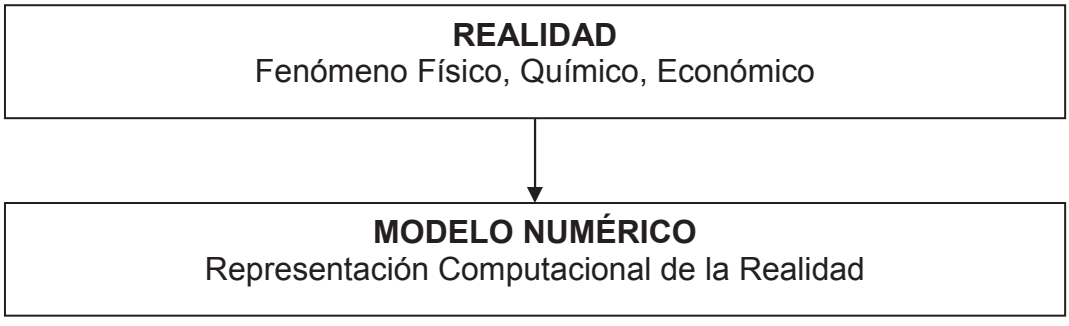
- **Simuladores de Vuelo**
- **Simuladores para Operadores de Centrales Nucleares**

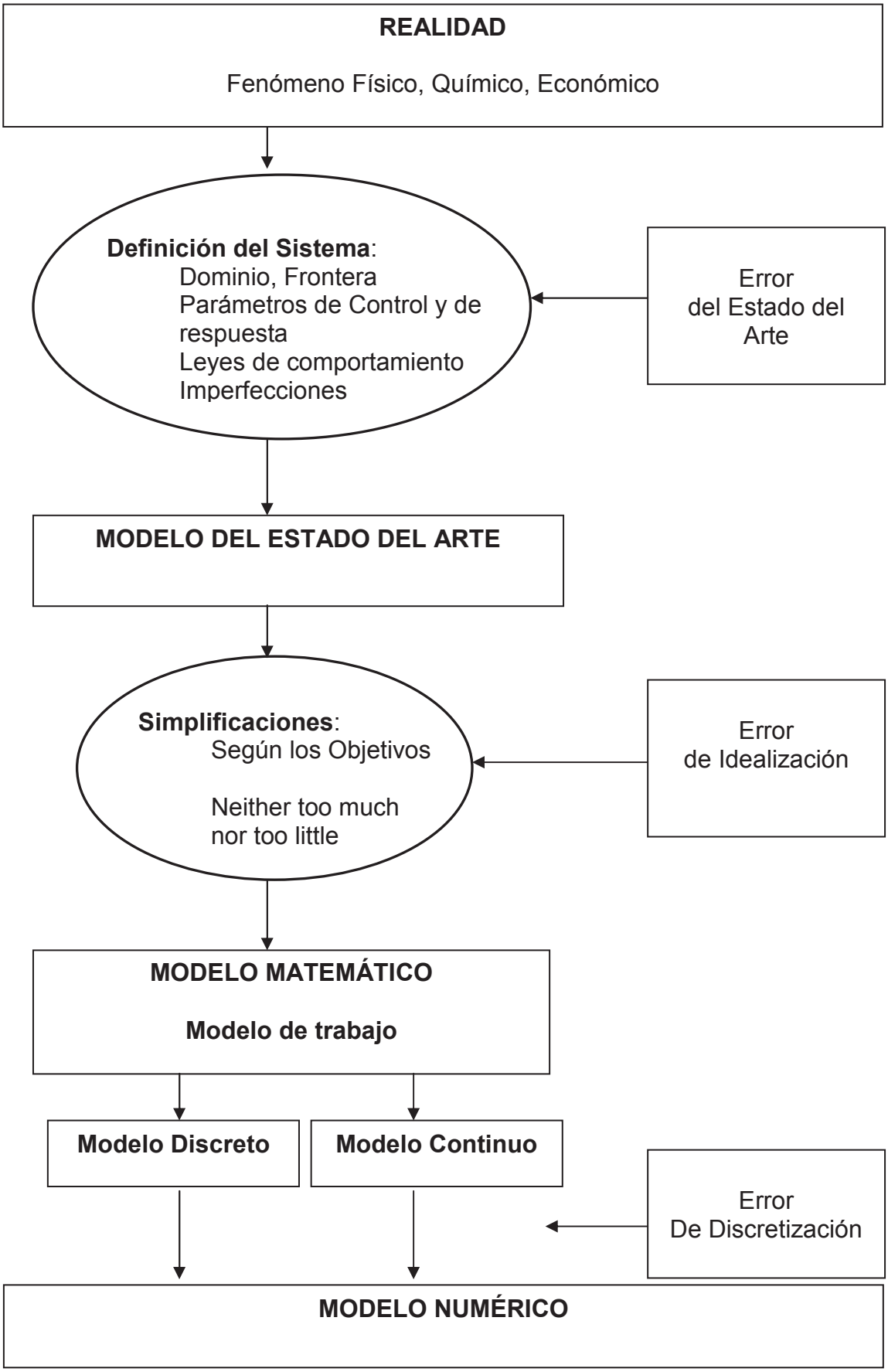
1.3 Monitoreo y control de datos

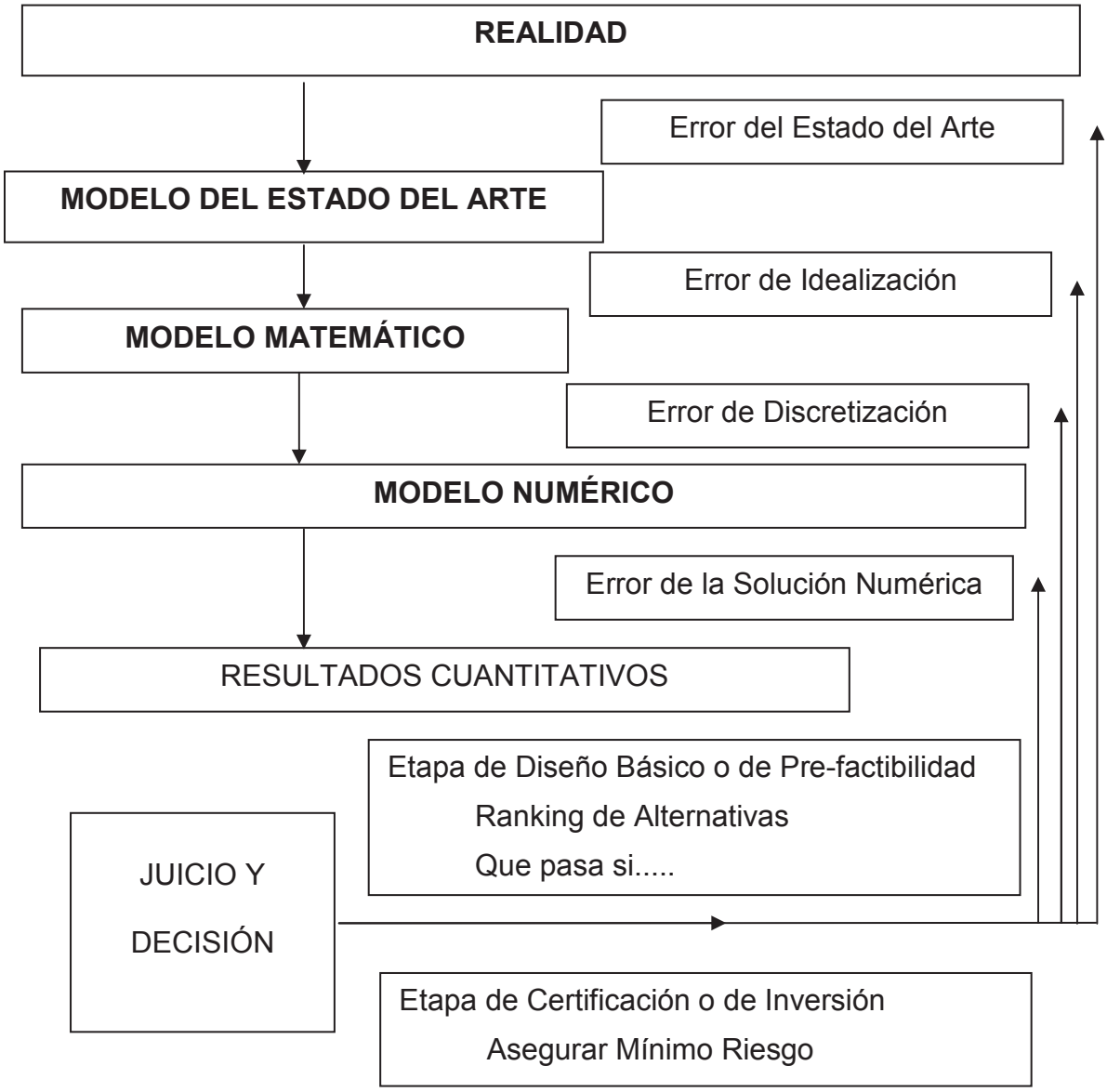
- **Vibraciones** de componentes estructurales: Amplitud y frecuencias.
- **Pulsos cardíacos**: presión y frecuencia en reposo, actividad, durante intervenciones quirúrgicas, etc.
- **Temperaturas en Reactores**: Nucleares o de Procesos.

1.4 Entretenimientos

- Juegos Virtuales
  DOS: Ping Pong, Laberintos, etc
  Windows: Flight Simulator, Golf, etc.

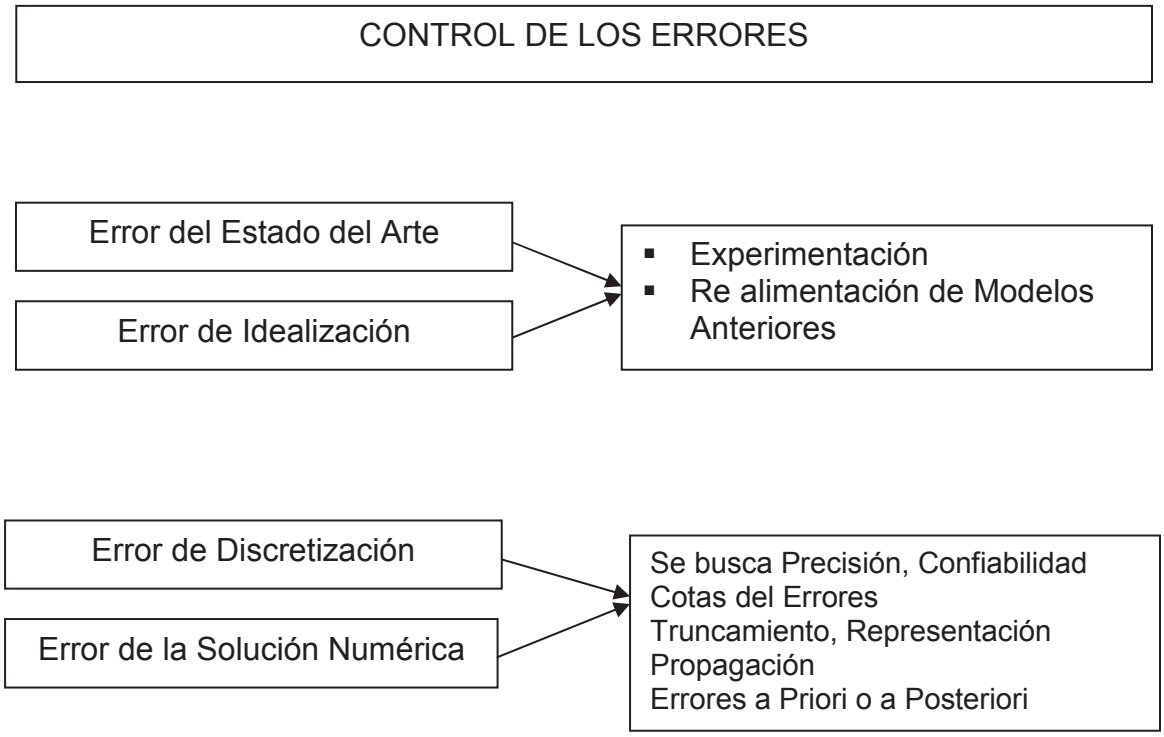


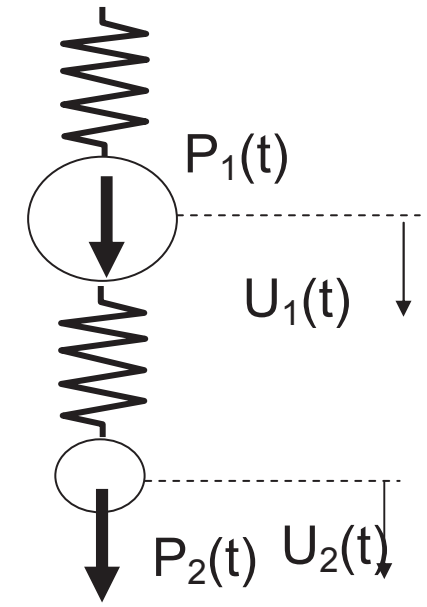

REALIDAD
Fenómeno Físico, Químico, Económico
Definición del Sistema:
Dominio, Frontera
Parámetros de Control y de respuesta
Leyes de comportamiento
Imperfecciones
Error del Estado del Arte
MODELO DEL ESTADO DEL ARTE
Simplificaciones:
Según los Objetivos
Neither too much nor too little
Error de Idealización
MODELO MATEMÁTICO
Modelo de trabajo
Modelo Discreto
Modelo Continuo
Error De Discretización
MODELO NUMÉRICO



Ejemplos:

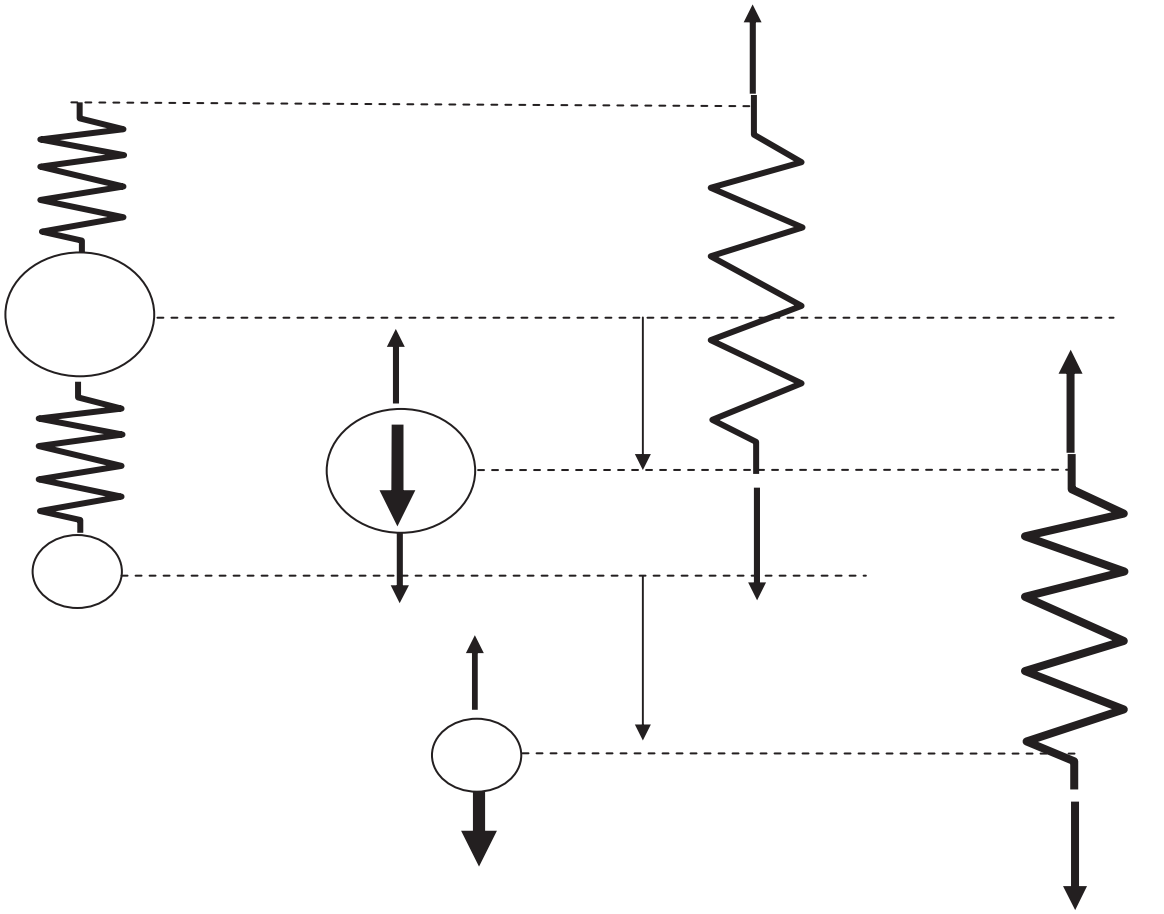
- Diseño de Turbinas, CFD, CAD, FEA, CAM
- Experimentos: Solo luego de optimizar modelo numérico.
- Inversiones de Fondos de Pensión
- Bioingeniería, Cirugía ocular.


CONTROL DE LOS ERRORES
Error del Estado del Arte
Error de Idealización
Experimentación
Re alimentación de Modelos Anteriores
Error de Discretización
Error de la Solución Numérica
Se busca Precisión, Confiabilidad
Cotas del Errores
Truncamiento, Representación
Propagación
Errores a Priori o a Posteriori



PARÁMETROS
- DE CONTROL
- DE RESPUESTA

PARÁMETROS
GEOMÉTRICOS Y FISICOS



CONDICION DE EQUILIBRIO

CONDICIONES CINEMATICAS

Algoritmia. Conceptos básicos

Introducción

Se presentan algunas estructuras algorítmicas de gran uso en procesos de cálculo y decisión en ingeniería. Se persigue presentarlas siguiendo ejemplo propios de las operaciones matriciales y los métodos de cálculo numérico. Si bien no se pretende presentar la sintaxis de lenguajes de programación, si se busca que los algoritmos se expresen en la forma de los denominados pseudocódigos

Definiciones

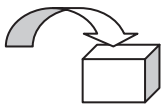
Algoritmo:
Es una forma ordenada de describir un procedimiento. La forma más elemental de mostrar un algoritmo es la de pseudo código, que constituye la expresión en palabras y ecuaciones de los pasos a seguir para desarrollar el procedimiento que se busca describir. Existen otras formas, como a de diagrama de flujo, de bloques, etc.

Para describir un proceso mediante la definición de un algoritmo se utilizan elementos tales como variables, constante, operadores algebraicos y lógicas; y estructuras típicas como las secuenciales y las iterativas. Entre éstas últimas es de destacar las llamadas "repetir", "mientras"

Todo algoritmo debe tener los siguientes elementos

- Declaración de variables
- Ingreso de datos o asignaciones primarias
- Proceso propiamente dicho
- Entrega de resultados

Variables:
Podríamos decir que una variable es una especie de recipiente (con una dirección, alojada en la memoria de la computadora, para la identificación de la computadora), en cuyo interior podemos colocar cierta información.
Para que la podamos identificar nosotros le daremos un nombre, que debe comenzar con una letra, y no debe poseer más de 10 caracteres.
Se sugiere que el nombre que le demos a la variable debe ser representativo de la información que alojemos en su interior.



Nombre de la variable: altura

A esta variable le podemos :

- **asignarle valores:** a través de constantes, otra/s variable/s, mediante expresiones algebraicas o lógicas. Con el símbolo ← se interpretará la asignación. Por ejemplo:

$$altura \leftarrow 10$$

se debe interpretar que en la variable denominada *altura* se asignará el valor 10; lo que es equivalente a decir que se almacenará el valor 10.

altura ← (base * 2) / 10

se debe interpretar que en la variable denominada *altura* se asignará el resultado de la operación (*base* * 2) / 10; lo que es equivalente a decir que se almacenarà el resultado de la operación (*base* * 2) / 10. Debiendo estar previamente asignada la variable *base*

- **modificarla**: a través de la asignación de un valor resultante de la operación que involucra a la propia variable a modificar. Por ejemplo:

  altura ← *altura* + 2

  se debe interpretar que en la variable denominada *altura* se asignará el resultado de la operación (*altura* + 2); lo que es equivalente a decir que se almacenarà el resultado de la operación (*altura* + 2).
- **borrarla**. por ejemplo: altura←∅
- **mostrar** (escribir en pantalla), por ejemplo:

  Escribir *altura*

  Se debe interpretar como la sentencia u orden que permite que el contenido de la variable aparezca visible (por pantalla de la computadora o por medio de un archivo) para quien ejecuta el algoritmo.
- **leer** (ingresar la información a la variable a través del teclado), por ejemplo:

  Leer *altura*

  Se debe interpretar como la sentencia u orden que permite que quien ejecuta el algoritmo asignarle un valor deseado a la variable.

Clasificación de las variables según su dimensión

Variables simples

Son las variables que ya hemos vistos: *altura*; *base*

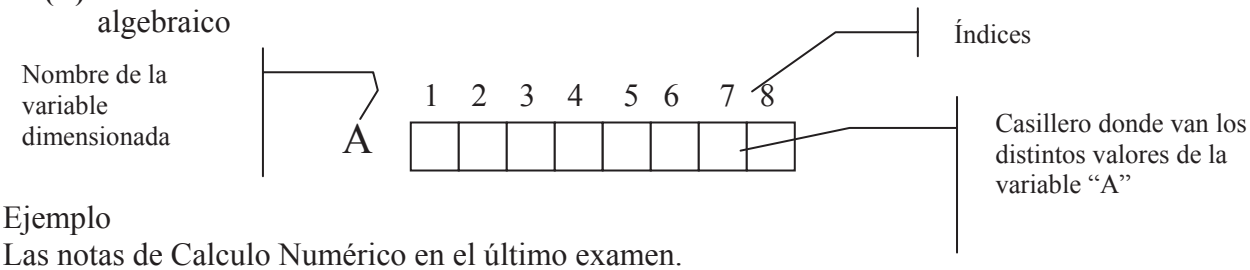
Es una variable que solo puede guardar un solo valor referida a la información que le deseamos almacenar.

Variables Dimensionadas

Son las variables que representan (o guardan información) referido a un mismo dato y que por su magnitud (cantidad de datos) es necesario dimensionarlas:

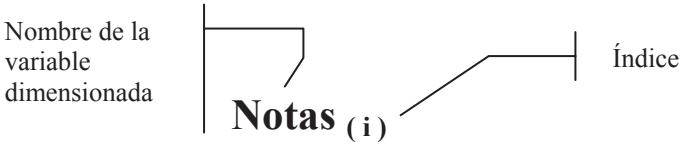
Variables de una dimensión

A (i) Posee una dimensión y son llamadas "vectores" por su semejanza al concepto algebraico



Ejemplo

Las notas de Calculo Numérico en el último examen.



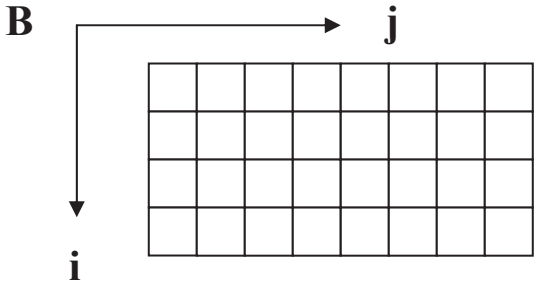
Variables de más de una dimensión

$\mathbf{B_{(i;j)}}$ Posee dos dimensiones y son llamadas "matrices (plano)" por su semejanza al concepto algebraico

Donde:

B ← nombre de la variable
i ← Representa la fila que identifica el casillero correspondiente
j ← Representa la columna que identifica el casillero correspondiente

Ejemplo: B (4 ;8) (en la declaración de variable)



Aclaración:

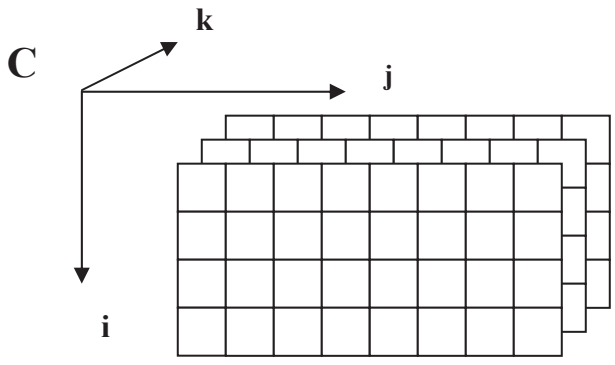
- **Si se escribe B (4;8) en la declaración de variable se está expresando una matriz de dos dimensiones llamada "B" que posee 4 filas y 8 columnas.**
- **Pero si en el desarrollo del algoritmo se escribe B (4;8) se esta haciendo referencia al casillero ubicado en la intersección de la fila 4 y la columna 8 de la variable dimensionada "B".**
- **Esta aclaración es valida para TODAS las variables dimensionadas**

$\mathbf{C_{(i;j;k)}}$ Posee tres dimensiones y son llamadas "matrices (espacio)" por su semejanza al concepto algebraico

Donde:

C ← nombre de la variable
i ← Representa la fila que identifica el casillero correspondiente
j ← Representa la columna que identifica el casillero correspondiente
k ← Representa la profundidad que identifica el casillero correspondiente

Ejemplo: C (4;8;3) (en la declaración de variable)



Clasificación de las variables según su contenido

La información que podemos colocar en su interior depende del tipo de variable. Los tipos de variables dependen del lenguaje de programación que se vaya a utilizar. En este curso utilizaremos solamente 4 tipos ya que trabajaremos en pseudocódigo. Asì se tienen los siguientes tipos de variables

- **Numérico-Enteros** : por ejemplo en su interior pueden contener:4 , 5 , -8 , -4587 , 4875
- **Numérico-Reales** : por ejemplo en su interior pueden contener:4.2 , 5.0 , -5.88 (los reales engloban a los enteros)
- **Lógicas**: por ejemplo en su interior solamente pueden contener: [V] o [F]
- **Carácter**: en su interior pueden contener cadenas de caracteres por ejemplo:"mamá", "casa", etc. Para mostrar el contenido de este tipo de variables y para no confundirnos con el nombre de las variables siempre al contenido lo colocaremos entre comillas

Se recuerda que lo anterior sólo son ejemplos de lo que puede contener una variable de un determinado tipo. Su nombre debe respetar lo ya establecido:

- Comenzar con una letra.
- El nombre no debe tener más de 10 caracteres.
- Ser representativo de lo que posee en su interior.

Siempre las variables que se van a utilizar en un algoritmo deben ser declaradas al iniciarse el mismo, estableciendo en esta declaración el nombre y el tipo de variable de que se trata.
Pseudocódigo
Var (Var1:Entero; Var2, Var3:Real; Var4:Carácter)

por ejemplo
Var(altura:Real; base:Entero)

Constantes:

Pueden ser del mismo tipo que las variables pero tienen la particularidad que no cambian su valor y por lo tanto tampoco se les debe asignar un nombre, no se declaran.

Operadores:

Son los elementos a través de los cuales puedo realizar operaciones algebraicas o establecer relaciones entre variables o constantes.
Los operadores tienen una relación establecida por el orden de prioridad establecida:

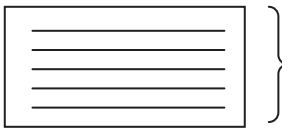
Prioridad	Operador	*Nombre*	Resultado
1	^	Potencia	Numérico
2	* /	Producto - Cociente	Numérico
3	+ - +	Suma – Resta – Concatenación de caracteres	Suma y resta el resultado es numérico. Concatenación el resultado es Carácter
4	= ≠ < ≤ > ≥	Relación	Lógico
5	**.NOT.**	Negación	Lógico
6	**.AND.**	Conjunción [Y] lógico	Lógico
7	**.OR.**	Disyunción [O] lógico	Lógico

Cuando en una expresión se tiene dos operadores de la misma prioridad se resuelve de izquierda a derecha.
La prioridad establecida deja de tener su efecto cuando se está ante en la presencia (de apertura y cierre) de paréntesis, corchetes o llaves.
Se debe resolver primero siempre lo que está encerrado por tales signos

ALGORITMO TIPO Secuencia

La secuencia es la relación más simple de un algoritmo, la misma establece que una línea de código no se ejecuta hasta que no se haya terminado de ejecutar la anterior y por consecuencia la línea de código siguiente no puede ejecutarse hasta que no se termine de ejecutar la línea de código actual.

Esquema:



EJEMPLO 1 ALGORITMO para calcular el valor medio de dos valores dados:
Datos: **Valores a,b**
Formula de calculo: **xm=(a+b)/2**

El Pseudocódigo debe contener todos los pasos necesarios para definir el proceso (declaración de variables, Ingreso de datos o asignaciones primarias, proceso propiamente dicho, entrega de los resultados), es decir:

- Declaración de variables
  Var (ExtrA, ExtrB, PtoMedC: Real)
- Ingreso de datos o asignaciones primarias
  Escribir "Ingrese extremo inferior del intervalo"
  Escribir "Ingrese extremo inferior del intervalo"
  Leer ExtrA
  Escribir "Ingrese extremo superior del intervalo"
  Leer ExtrB
- Proceso propiamente dicho
  PtoMedC ← (ExtrA + ExtrB) / 2
- Entrega de resultados
  Escribir "El valor medio es", PtoMedC

Cuerpo de líneas de código

Así el algoritmo queda de la siguiente manera:

```
Var (ExtrA, ExtrB, PtoMedC: Real)

Escribir "Ingrese extremo inferior del intervalo"
Escribir "Ingrese extremo inferior del intervalo"
Leer ExtrA
Escribir "Ingrese extremo superior del intervalo"
Leer ExtrB

PtoMedC ← (ExtrA + ExtrB) / 2

Escribir "El valor medio es", PtoMedC
```

Cuerpo de líneas de código

EJEMPLO 2 ALGORITMO para encontrar las raíces de una ecuación de segundo grado

Datos: **Coeficientes a, b, c**

Formula de cálculo: $Raiz1,2 = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$

```
Var (a, b, c, Raiz1, Raiz2: Real)

Escribir "Ingrese coeficiente a"
Leer a
Escribir "Ingrese coeficiente b"
Leer b
Escribir "Ingrese coeficiente c"
Leer c

Raiz1← (-b+(b^2-4*a*c) ^0,5)/(2*a)
Raiz2← (-b-(b^2-4*a*c) ^0,5)/(2*a)

Escribir "La raiz 1 es", Raiz1
Escribir "La raiz 1 es", Raiz1
```

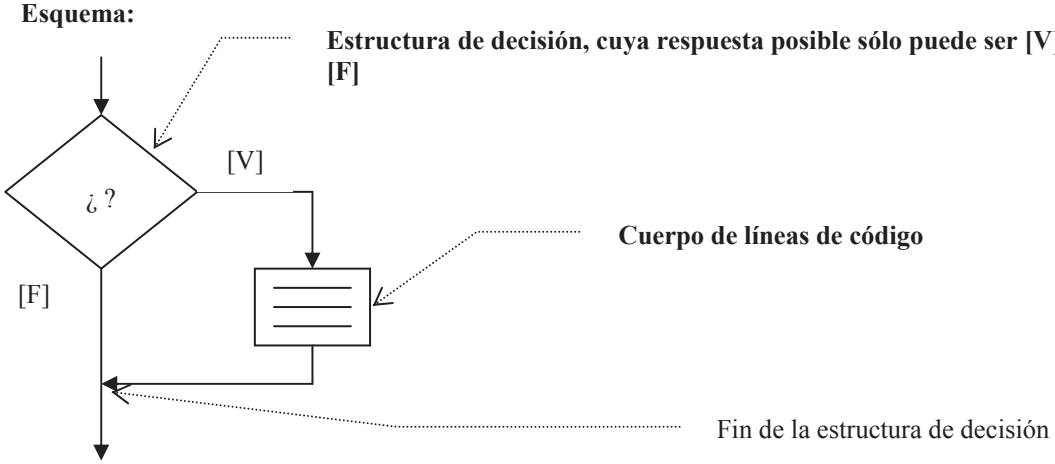
Cuerpo de líneas de código

Ejercicio:
Escribir un algoritmo que calcule la suma de los numeros enteros de 1 hasta 10, almacene el resultado en una variable llamada suma y muestre el resultado

ALGORITMO TIPO Decisión simple

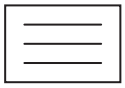
La decisión simple es cuando a través de una pregunta cuyo resultado sólo puede ser lógico (Verdadero [V] o Falso [F]), se realiza una determinada acción pre-establecida (cuerpo de línea de código) si la respuesta es Verdadera [V] para luego continuar con la ejecución del programa.
Pero si la respuesta es Falsa [F] se continúa con la ejecución del programa **sin** realizar ninguna acción pre-establecida.

Esquema:



Pseudocódigo

IF (¿Pregunta ?) **THEN**



ENDIF

Por ejemplo

En un proceso se debe controlar si la variable ExtrA es o no negativa; y si lo es, se la debe incrementar en 1.

Si (ExtrA < 0) ***Entonces***
ExtrA ← ExtrA + 1 ⟵ Cuerpo de líneas de código
Finsi ⟵ Fin de la estructura de decisión

O bien,

IF (ExtrA < 0) ***THEN***
ExtrA ← ExtrA + 1 ⟵ Cuerpo de líneas de código
ENDIF ⟵ Fin de la estructura de decisión

EJEMPLO
ALGORITMO para encontrar las raíces de una ecuación de segundo grado, sòlo si el discriminante es positivo

Datos: **Coeficientes a, b, c**

Formula de cálculo: $Raiz1,2 = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$

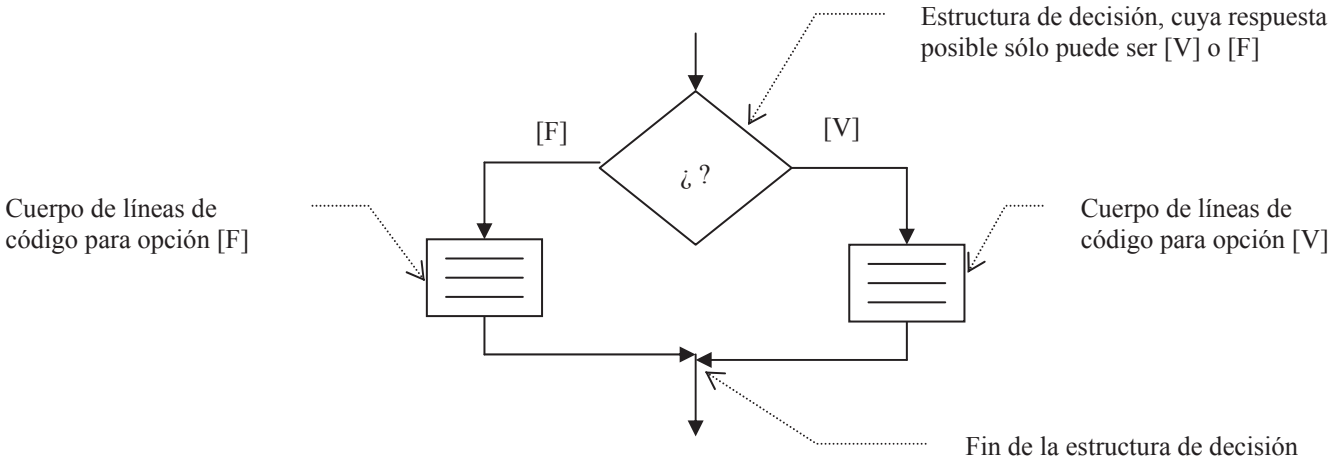
```
Var (a, b, c, Discrim, Raiz1, Raiz2: Real)

Escribir "Ingrese coeficiente a"
Leer a
Escribir "Ingrese coeficiente b"
Leer b
Escribir "Ingrese coeficiente c"
Leer c

Discrim ← (b^2-4*a*c)
IF ((Discrim>0).OR.( Discrim=0))THEN
      Raiz1← (-b+(Discrim) ^0,5)/(2*a)
      Raiz2← (-b-(Discrim) ^0,5)/(2*a)
ENDIF

Escribir "La raiz 1 es", Raiz1
Escribir "La raiz 1 es", Raiz1
```

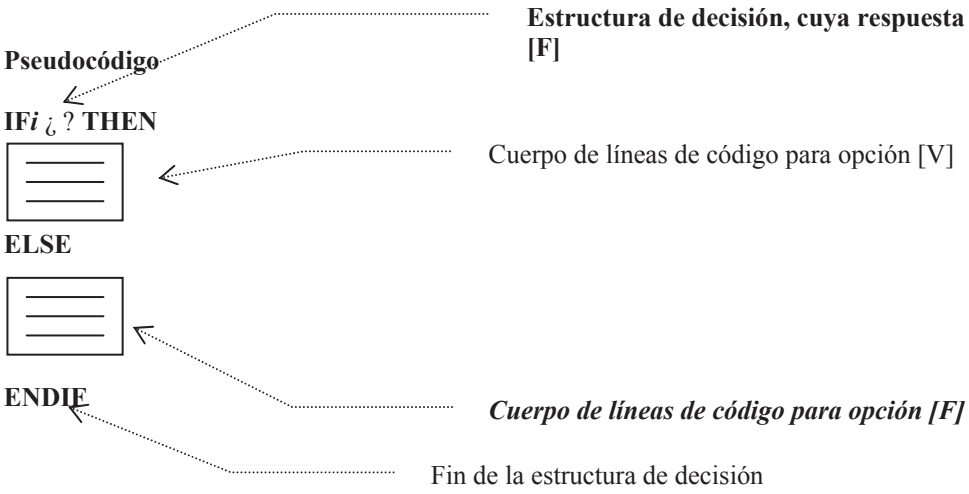
Cuerpo de líneas de código

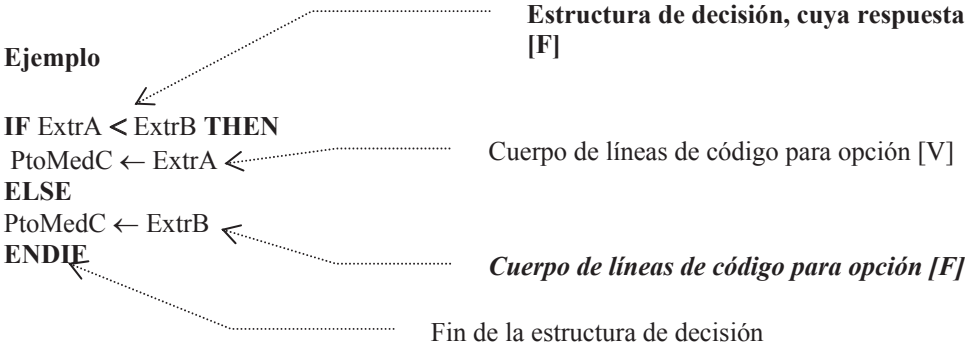
ALGORITMO TIPO Decisión Compuesta

La decisión compuesta al igual que la decisión simple es cuando a través de una pregunta cuyo resultado sólo puede ser lógico (Verdadero [V] o Falso [F]), se realiza una determinada acción.
En este caso siempre se ejecuta un cuerpo de líneas de código, pero si la respuesta es Verdadera [V] se ejecuta un cuerpo de línea de código distinto al que si la respuesta es Falsa [F]
Es decir tenemos 2 cuerpos de líneas de código distintos uno para la opción Verdadera [V] y otro para la opción Falsa [F].

Esquema







ALGORITMO TIPO Estructuras Iterativas

Cuando un cuerpo de líneas de código debe repetirse en un mismo algoritmo, puede resultar muy engorroso y dar mucho trabajo hacerlo. Para solucionar esto se utilizan las estructuras iterativas. Veremos cuatro posibilidades.

- Mientras
- Variar
- Repetir
- Iterar

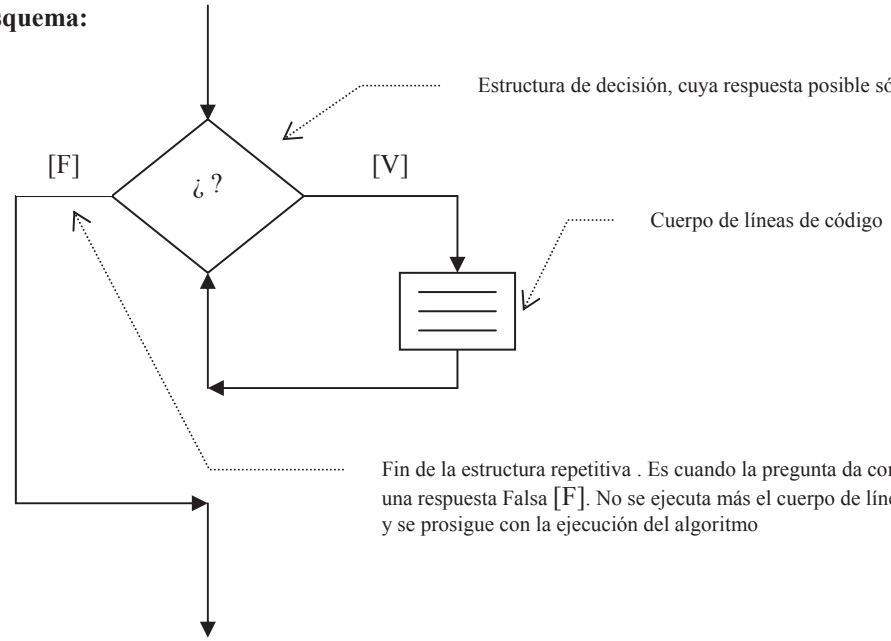
- **Mientras (DO WHILE)**

Esta estructura iterativa realiza la repetición del cuerpo de líneas de código que encierra mientras una pregunta cuyo resultado sólo puede ser lógico (Verdadero [V] o Falso [F]) sea Verdadero [V], en cuanto la respuesta sea Falsa [F] no ejecuta el cuerpo de líneas de código y prosigue con la ejecución del algoritmo. Como comentario podemos agregar que en esta estructura iterativa el cuerpo de líneas de código que encierra puede llegar a no ejecutarse nunca si la pregunta da en primera instancia un Falso [F]).

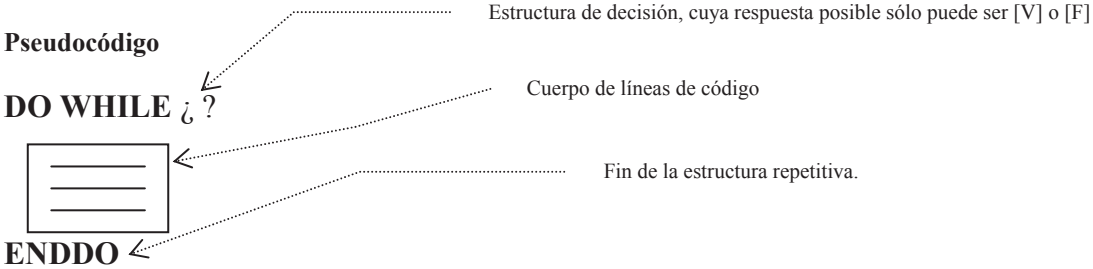
En esta estructura el análisis se hace antes de la ejecución del cuerpo de líneas de código.

Para no caer en un bucle infinito que provocaría un desbordamiento de memoria (se cuelga la PC), siempre se debe tratar de que en el cuerpo de líneas de código la respuesta a la pregunta tienda a ser Falsa [F], de tal manera que permita la continuación de la ejecución del algoritmo.

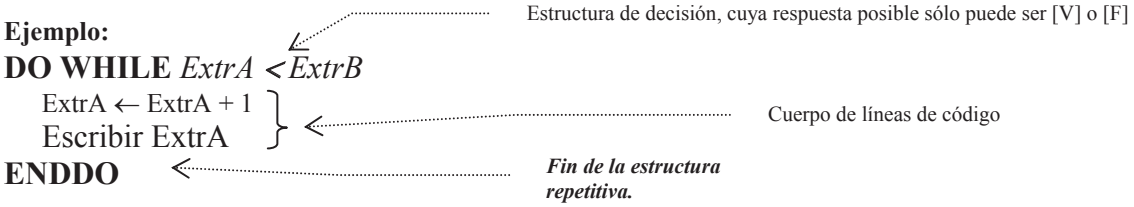
Esquema:



Pseudocódigo



Ejemplo:



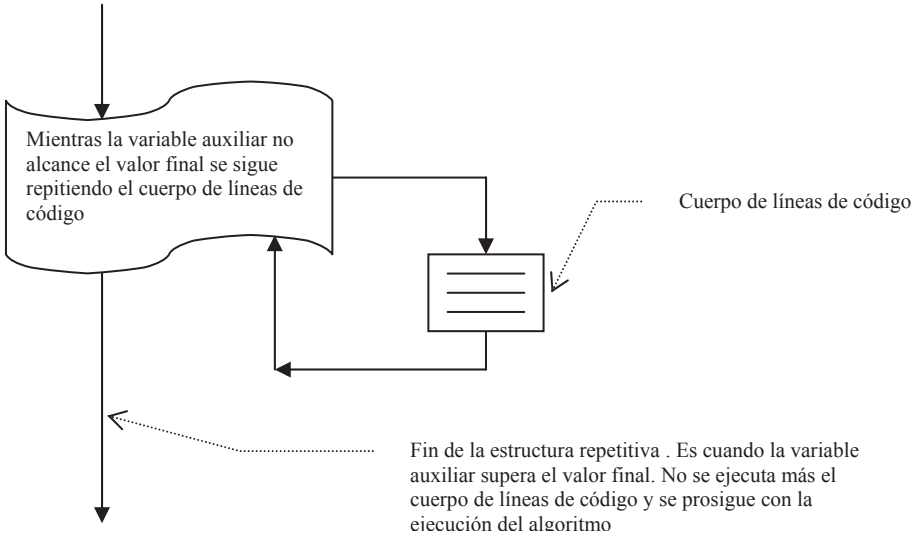
Ejemplo: buscar en un vector de numeros la primer componente del vector que sea mayor a 100.

Ejemplo: a un numero ingresado como dato dividirlo sucesivamente por 2 hasta que su resultado sea menor a la unidad. Trabajar sòlo con nùmeros positivos

- **Variar (DO FOR)**

Es un caso particular de Mientras.
Esta estructura repetitiva es para cuando sabemos exactamente el número de veces que debe repetirse un cuerpo de líneas de código.
Para ello se utiliza una variable auxiliar (que debe cumplir todas las condiciones de variables ya expuestas anteriormente, generalmente es una variable de tipo entera).
La variable auxiliar varía desde un valor inicial a un valor final a través de un determinado paso.

Esquema:



Pseudocódigo

DO FOR *VarAux* **OF** *VI* **TO** *VF* **STEP** *P*

Variable auxiliar = VarAux
Valor Inicial = VI
Valor Final = VF
Paso = P

Cuerpo de líneas de código

ENDDO

Fin de la estructura repetitiva

DO FOR *Aux* **OF** *1* **TO** *20* **STEP** *1*

Variable auxiliar = Aux
Valor Inicial = 1
Valor Final = 20
Paso = 1

Escribe PtoMedC
PtoMedC ← PtoMedC / 2

Cuerpo de líneas de código

ENDDO

Fin de la estructura repetitiv

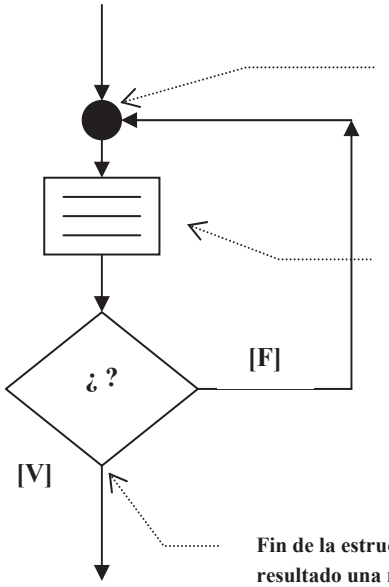
Ejemplo: Sumar los elementos de un vector
Ejemplo: Restar dos vectores de igual numero de componentes
Ejemplo Obtener el producto escaar entre dos vectores

Ejemplo: Obtener la traza de una matriz

- **Repetir**

En esta estructura el análisis se hace después de la ejecución del cuerpo de líneas de código.
Esta estructura repetitiva realiza la repetición del cuerpo de líneas de código que encierra mientras una pregunta (que se encuentra al final) cuyo resultado sólo puede ser lógico (Verdadero [V] o Falso [F]) sea Falsa [F], en cuanto la respuesta sea Verdadera [V] no ejecuta el cuerpo de líneas de código y prosigue con la ejecución del algoritmo. Como comentario podemos agregar que en esta estructura iterativa el cuerpo de líneas de código que encierra siempre se ejecuta como mínimo una vez.
Para no caer en un bucle infinito que provocaría un desbordamiento de memoria (se cuelga la PC), siempre se debe tratar que en el cuerpo de líneas de código la respuesta a la pregunta tienda a ser Verdadera [V], de tal manera que permita la continuación de la ejecución del algoritmo.

Esquema



Comienzo de la estructura

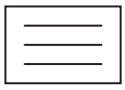
Cuerpo de líneas de código

Fin de la estructura repetitiva . Es cuando la pregunta da como resultado una respuesta Verdadera [V]. No se ejecuta más el cuerpo de líneas de código y se prosigue con la ejecución del algoritmo

Pseudocódigo

Comienzo de la estructura repetitiva

DO



Cuerpo de líneas de código

Fin de la estructura repetitiva.

IF ¿ ? **EXIT**

Estructura de decisión, cuya respuesta posible sólo puede ser [V] o [F]

ENDDO

Ejemplo

Comienzo de la estructura repetitiva

DO

ExtrA ← ExtrA - 1
Escribir ExtrA

Cuerpo de líneas de código

IF *ExtrA < 10* **EXIT**

Fin de la estructura repetitiva

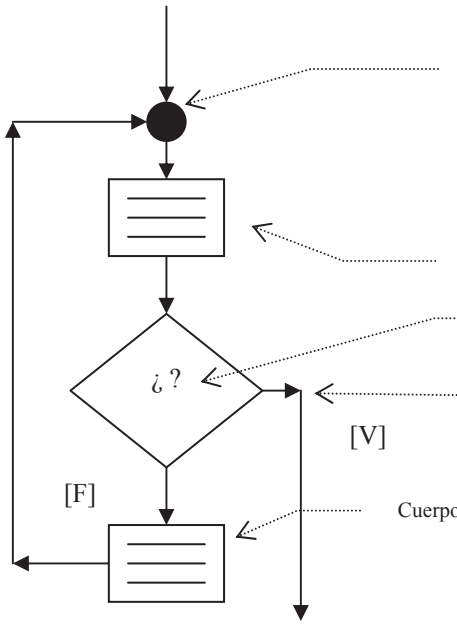
ENDDO

Estructura de decisión, cuya respuesta posible sólo puede ser [V] o [F]

- **Iterar**

En esta estructura vamos a tener 2 cuerpos de líneas de código (que llamaremos Cp1 y Cp2) y el análisis se hace entre los 2 cuerpos de líneas de código.
Esta estructura repetitiva realiza la repetición de los 2 cuerpos de líneas de código que encierra mientras una pregunta (que se encuentra entre ellas) cuyo resultado sólo puede ser lógico (Verdadero [V] o Falso [F]) sea Falsa [F]. En cuanto la respuesta sea Verdadera [V] no ejecuta el cuerpo de líneas de código Cp2 y prosigue con la ejecución del algoritmo. Como comentario podemos agregar que en esta estructura iterativa el cuerpo de líneas de código Cp1 se ejecuta al menos una vez y siempre una vez más que el Cp2.
Para no caer en un bucle infinito que provocaría un desbordamiento de memoria (se cuelga la PC), siempre se debe tratar de que en el cuerpo de líneas de código la respuesta a la pregunta tienda a ser Verdadera [V], de tal manera que permita la continuación de la ejecución del algoritmo.

Esquema



Comienzo de la estructura repetitiva

Cuerpo de líneas de código Cp1

Estructura de decisión, cuya respuesta posible sólo puede ser [V] o [F]

Fin de la estructura repetitiva . Es cuando la pregunta da como resultado una respuesta Verdadera [V]. No se ejecuta el cuerpo de líneas de código Cp2 y se prosigue con la ejecución del algoritmo

Cuerpo de líneas de código Cp2

Pseudocódigo

Comienzo de la estructura repetitiva

DO

Cuerpo de líneas de código Cp1

IF ¿ ? **EXIT**

Estructura de decisión, cuya respuesta posible sólo puede ser [V] o [F]

Cuerpo de líneas de código Cp2

ENDDO

Fin de la estructura repetitiva.

Ejemplo

Comienzo de la estructura repetitiva

DO

Cuerpo de líneas de código Cp1

ExtrA ← ExtrA * 10

IF *100 < ExtrA* EXIT

Estructura de decisión, cuya respuesta posible sólo puede ser [V] o [F]

Escribir ExtrA

Cuerpo de líneas de código Cp2

ENDDO

Fin de la estructura repetitiva.

EJEMPLO. Algoritmo del Método de Bisección.

El método de bisección se utiliza para encontrar la abscisa solución (x_s) tal que $f(x_s)=0$, de una cierta función $f(x)$ conocida; que debe ser continua en un intervalo $[a\ ,\ b]$, donde se debe cumplir: $sig[f(a)] \neq sig[f(b)]$

La función $f(x)$ se debe programar en cada nueva oportunidad.

El algoritmo debe tener como Entradas: a , b , Error: E
Y debe tener como Salida: solución aproximada

ALTERNATIVA 1.
Un posible algoritmo para este problema es el siguiente:

Algoritmo Bisección para función *f(x)*
Var(a, b: entero; p, E: real)

Nota: Aquí se declaran la totalidad de las variables a utilizar.
Las sentencias que siguen son las "ordenes" para ingresar datos, que deben cumplir las condiciones iniciales del método

Escribir ("ingrese el valor de a")
Leer (a) \ ingresa el valor de "a"
Escribir ("ingrese el valor de b")
Leer (b) \ ingresa el valor de "b"
Escribir ("ingrese el valor del error admisible")
Leer (E) \ ingresa el valor de "E"

Nota: Aquí se calcula la primera aproximación de la raíz.

$$p \longleftarrow a + \frac{b-a}{2}$$

DO WHILE ((*f(p)* ≠ 0) **[AND]** ($\frac{b-a}{2} > E$))

Nota: El ciclo iterativo controlado por el DO WHILE, se detiene cuando una de las dos condiciones se deja de cumplir.

IF ($f(a) * f(p) > 0$) **THEN**

$$a \longleftarrow p$$

Nota: Se reasigna "a" y se mantiene "b"

ELSE

$$b \longleftarrow p$$

Nota: Se mantiene "a" y se reasigna "b"

ENDIF

$$p \longleftarrow a + \frac{b-a}{2}$$

Nota: Se calcula la nueva arpoximación

ENDDO
Escribir ("la solución aproximada es:" *p*)
END

Se debe destacar que:

- Los controles de detención (o medidas del error) se calcula en la misma sentencia DO WHILE.
- Las raíces aproximadas en iteraciones anteriores se pierden.
- Los límites del intervalo donde está la raíz en cada iteración se pierden.

- El control del número de iteraciones que se va realizando no se realiza.

NOTESE BIEN: Realice un diagrama de flujo de este algoritmo
ALTERNATIVA 2.
Un posible algoritmo para este problema es el siguiente:

```
Algoritmo Bisección para función f(x)
Var(a, b, N, i: entero; p: real)                    declaración de variables

Escribir ("ingrese el valor de a")
Leer (a)                                            \ ingresa el valor de "a"
Escribir ("ingrese el valor de b")
Leer (b)                                            \ ingresa el valor de "b"
Escribir ("ingrese el Nº máx de itaraciones")
Leer (N)                                            \ ingresa el valor de "N"

p <---- a + (b - a)/2                               \ Calcula la primera aproximación

DO FOR  i = 1   TO   i = N   STEP.  1

    IF  ( f(a) * f(p) > 0 )  THEN

        a <---- p                                   \ Se reasigna "a" y se mantiene "b"

    ELSE
        b <---- p                                   \ Se mantiene  "a" y se reasigna "b"
    ENDIF
    p <---- a + (b - a)/2                           \ Se calcula la nueva aproximación
ENDDO

Escribir ("la solución aproximada es:" p)           \ Se tiene una aproximación luego de N iteraciones
END
```

Se debe destacar que:
- Los valores iniciales de a y b deben ser tales que cumplan la condición de inicialización del método de bisección
- Los controles de detención no existen, ya que es una "tarea" que se realizará N veces.
- La raíz aproximada que se retiene es la última.
- Los límites del intervalo donde está la raíz en cada iteración se pierden.

NOTESE BIEN: Realice un diagrama de flujo de este algoritmo

ALTERNATIVA 3.
Un posible algoritmo para este problema es el siguiente, que combina los dos algoritmos anteriores:

Algoritmo Bisección para función *f(x)*
Var(a, b, N: entero; p, E: real) \ decalración de la variables a utilizar

Escribir ("ingrese el valor de a")
Leer (a) \ ingresa el valor de "a"
Escribir ("ingrese el valor de b")
Leer (b) \ ingresa el valor de "b"
Escribir ("ingrese el valor del error admisible")
Leer (E) \ ingresa el valor de "E"
Escribir ("ingrese el Nº máx de itaraciones")
Leer (N) \ ingresa el valor de "N"

$p \longleftarrow a + \frac{b-a}{2}$ \ se calcula la primera aproximación de la raíz

$i \longleftarrow 1$ \ se inicia el control de número de iteraciones

DO WHILE ($(f(p) \neq 0)$ [AND] $(\frac{b-a}{2} > E)$ [AND] $(i \leq N)$)
\ se detiene cuando una de las tres condiciones se cumple

IF $f(a) * f(p) > 0$ **THEN**

$a \longleftarrow p$ \ Se reasigna "a" y se mantiene "b"

ELSE

$b \longleftarrow p$ \ Se mantiene "a" y se reasigna "b"

ENDIF

$p \longleftarrow a + \frac{b-a}{2}$ \ Se calcula la nueva aproximación de la raíz.

$i \longleftarrow i + 1$ \ Se calcula el número de iteración a realizar

ENDDO
Escribir ("la solución aproximada es:" *p*)
END

Se debe destacar que:

- Los valores iniciales de a y b deben ser tales que cumplan la condición de inicialización del método de bisección
- La raíz aproximada que se retiene es la última.
- Los límites del intervalo donde está la raíz en cada iteración se pierden.
- Los controles de detención (o medidas del error) se calcula en la misma sentencia DO WHILE.
- Como control de detención se tiene se debe cumplir alguno de los siguientes:
  - el "valor de la función es cero"
  - el valor de la longitud del intervalo es menor que E,
  - El número de iteración superó el valor "máximo de iteraciones"

NOTESE BIEN: Realice un diagrama de flujo de este algoritmo

SOLUCION NUMÉRICA DE

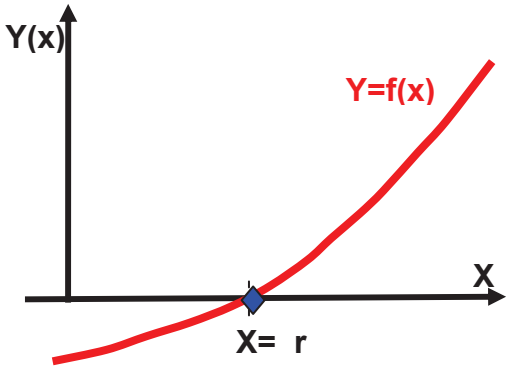
RAÍCES DE ECUACIONES NO LINEALES.

1 INTRODUCCIÓN

En diversos problemas de la Ingeniería resulta necesario obtener los valores de las variables que hacen cero un determinada función conocida. Así ocurre cuando se buscan los valores que anulan el polinomio característico de una matriz para determinar sus autovalores. Algo similar ocurre cuando se buscan los valores que anulan las denominadas funciones trascendentes en la determinación de estados inestables de sistemas conservativos o frecuencias naturales de sistemas dinámicos.

En todos los casos se puede formular el problema matemáticamente de la siguiente forma:

Dada una función continua y=f(x) de R → R, se busca x=r tal que f(r)=0



Geométricamente se trata de buscar el punto de abscisa r y ordenada 0, que verifican la relación funcional 0=f(r), siendo y=f(x) la función dada. En la gráfica adjunta el punto solución se identifica con un rombo. El punto solución se lo denomina *raíz de la ecuación no lineal*. *La ecuación no lineal es la función f(x) igualada a cero*.

Para ecuaciones polinómicas de grado 2 o 3 existen fórmulas explícitas para calcular las raíces. Pero en general para polinomios mayores a 3 no es frecuente encontrar dichas expresiones. Lo mismo ocurre cuando se trata de ecuaciones trascendentes que tiene expresiones trigonométricas.

En estas notas se presentan ideas básicas para resolver el problema mediante métodos iterativos. En principio se platea el esquema genérico a seguir, para luego describir las características de un proceso iterativo general. Se presenta a continuación sólo una clasificación de los métodos iterativos más frecuentes.

Se debe señalar que una descripción detallada de los distintos métodos, sus bases, algoritmos y ejemplos se puede encontrar en el texto *Métodos Numéricos para Ingenieros* de S. Chapra, R. Canale; Mc Graw Hill, que se recomienda consultar.

2 PROCEDIMIENTO GENERAL

Para encontrar las raíces de una ecuación no lineal es conveniente seguir los siguientes pasos:

- **Paso Inicial**

  Es conveniente realizar un análisis de la función a los efectos de determinar las singularidades, posibles discontinuidades, asíntotas y toda la información posible a los efectos de elegir adecuadamente las variables iniciales de los procesos iterativos.

- **Paso de Acercamiento**

  Se trata de encontrar un intervalo en el eje X donde exista al menos una raíz de la ecuación no lineal.

  Para **funciones continuas** en un intervalo $[a_k ; b_k]$ perteneciente al eje X de las abscisas una condición que debe cumplir la función es que cambie de signo al menos

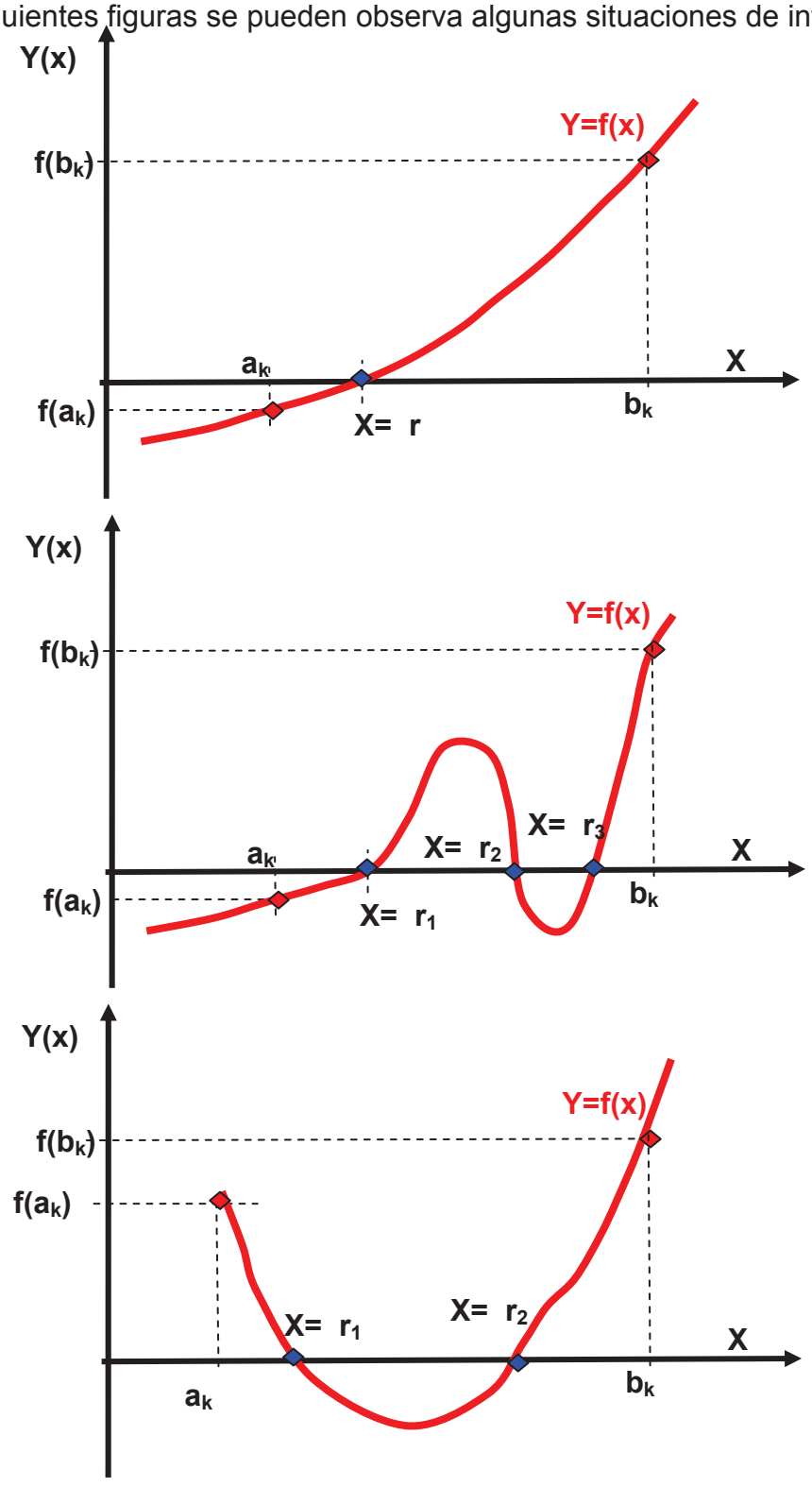
una vez. Entonces, dados los valores de abscisas a_k ; b_k que definen el intervalo $[a_k ; b_k]$ de los números reales,

Si $f(a_k)*f(b_k) < 0$

$\Rightarrow X= r \varepsilon [a_k ; b_k]$

Siendo X= r al menos una de las raíces buscadas.

En las siguientes figuras se pueden observa algunas situaciones de interés.



- **Paso de Aproximación**

  En general se trata de métodos iterativos. Entre los más difundidos se puede destacar los siguientes:

  Métodos de Intervalos

  Método de Bisección

  Método de Regula Falsi

  Métodos Abiertos

  Método de la Secante o de Newton Lagrange

  Método de Newton

  Métodos de Puntos Fijos

3 MÉTODOS ITERATIVOS EN GENERAL

Los Métodos Iterativos son procedimientos que generan elementos de una *sucesión de soluciones aproximadas*. Bajo ciertos requisitos dichos elementos se aproximan cada vez más a la solución exacta. Es decir que el error en cada iteración es cada vez menor.

En general los métodos iterativos tienen las siguientes características:

- Condición de Inicialización
- Formula de recurrencia
- Control de detención
- Actualización de variables

3.1 Condición de Inicialización

Son las condiciones que deben cumplirse para que la sucesión de soluciones aproximadas converja a la solución exacta.

Así por ejemplo en todos los métodos se debe cumplir que la función sea continua en el entrono de trabajo.

En los métodos de intervalos se debe conocer un intervalo en el cuál la función sea continua y tenga signo contrario en sus extremos.

3.2 Fórmula de Recurrencia

Son las fórmulas con las que se generan los elementos de la sucesión de soluciones aproximadas.

3.3 Controles de Detención

Son las condiciones que permiten detener el procedimiento. En general se expresan de forma que su evaluación de un resultados lógico (Verdadero o Falso).

Suele ser las *Medidas del Error* que se está cometiendo; o bien, *Medidas del proceso de convergencia*. En todos los casos pueden ser medidas absolutas o relativas.

En general son comparaciones respecto de valores admisibles de error. Estos valores admisibles son definidos en particular en cada problema. De esta manera los procesos iterativos se hacen tan "precisos" se desee.

3.4 Actualización de Variables

Son las reasignaciones de las variables de trabajo a los efectos de cumplir con las condiciones de inicialización y poder realizar un nuevo ciclo o iteración.

3.5 Síntesis algoritmia

Se puede sintetizar algorítmicamente usando "bloques" que deben existir en el algoritmo o definición del método iterativo

INICIALIZACIÓN

Se deben definir los contenidos de las variables de modo que se cumplan las condiciones de Inicialización del método

HACER MIENTRAS No Hay Solución es Verdadero
DOWHILE (NHS)

RECURRENCIA
Se debe evaluar la nueva solución aproximada correspondiente a la nueva iteración o ciclo.
Se obtiene r_{k+1}

CONTROL DE DETENCIÓN
Si alguna Medida de Error es adecuada entonces se ha obtenido la solución buscada y debe asignarse Falso a NHS.

SI (Valor Absoluto de $f(r_{k+1})$ < Tolerancia)
NHS es FALSO
FINSI

ACTUALIZACIÓN DE VARIABLES
Se reasignan las variable de modo que se cumplan con las condiciones de Inicalización

ENDDO o Fin del HACER MIENTRAS

4 SINTESIS DE LOS DISTINTOS MÉTODOS

Para un análisis detallado de los distintos métodos es recomendable consultar el texto "Métodos Numéricos para Ingenieros" de S. Chapra, R. Canale, u otros.
En los distintos métodos iterativos considerados en el curso se deben destacar los elementos de los procesos iterativos en general; es decir, inicialización, recurrencia, control de detención, y actualización.

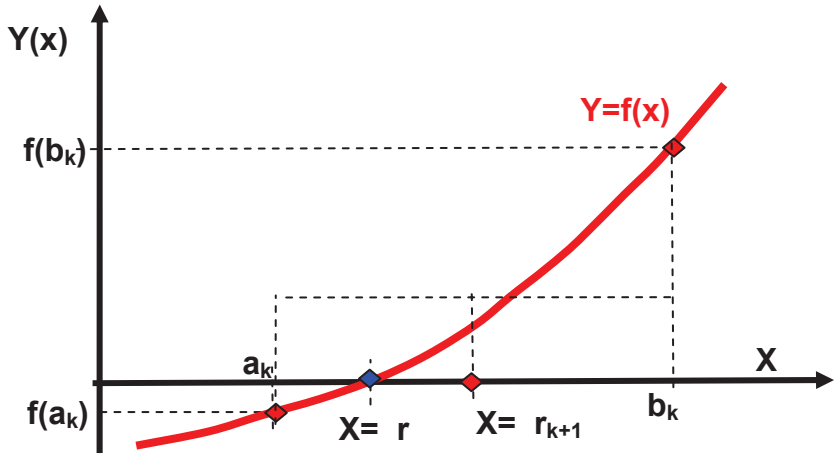
4.1 Método de Bisección

- **Inicialización**

Es necesario tener como datos dos valores de abscisas $x=a_k$; $x=b_k$ que definan un intervalo $[a_k ; b_k]$ en el eje de las abscisas X en el cuál la función no lineal tenga al menos una raíz. Esto es abscisas tales que

$$f(a_k).f(b_k) < 0$$

siendo f(x) la función no lineal cuyas raíces se buscan. Gráficamente esta condición se puede ilustrar de la siguiente forma



- **Recurrencia**

Dadas las abscisas $x=a_k$; $x=b_k$ que definan un intervalo $[a_k ; b_k]$ en el eje de las abscisas X en el cuál la función no lineal tiene al menos una raíz, la aproximación de la raíz se calcula como la abscisa media de dicho intervalo. Esto es,

$$r_{k+1} = (a_k + b_k)/2$$

- **Control de Detención**

Calculada la nueva aproximación de la raíz r_{k+1}, se debe controla si dicha abscisa es efectivamente la raíz. Es decir se debe controlar ***si se cumple que***

$$\left| f(r_{k+1}) \right| \leq \varepsilon_f$$

Donde la barras indican valor absoluto y ε_f es una magnitud tan pequeña y cercana a cero como la precisión del problema a resolver lo requiera.

En algunas situaciones, por ejemplo cuando la función no lineal *f(x)* intersecta al eje de las abscisas X en forma "muy vertical", la condición anterior es de "difícil" cumplimiento. Resulta así conveniente controlar ***si se cumple que***

$$\left| r_{k+1} - r_k \right| \leq \varepsilon_{ra}$$

O en términos relativos, ***si se cumple que***

$$\left| \frac{r_{k+1} - r_k}{r_{k+1}} \right| \leq \varepsilon_r$$

Siendo ε_{ra} y ε_r magnitudes tan pequeñas como la precisión del problema a resolver lo requiera.

Es útil fijar que el proceso iterativo no supere un número máximo de iteraciones. Es decir que se debe controlar ***si se cumple que***

$$k \leq MaxIter$$

Siendo *k* la iteración considerada, y *MaxIter* el número de iteraciones máximo fijado para el problema en consideración.

- **Actualización de Variables**

Dadas las abscisas $x=a_k$; $x=b_k$ y la nueva aproximación de la raíz r_{k+1} quedan definidos dos nuevos subintervalos en el eje de las abscisas X: $[a_k ; r_{k+1}]$ y $[r_{k+1} ; b_k]$.

Es necesario establecer en que intervalo se cumple con la condición de inicialización del método para poder así comenzar una nueva iteración.

Se puede plantear el siguiente algoritmo o proceso:

Si $[f(\boldsymbol{a_k}).f(\boldsymbol{r_{k+1}}) < 0$]es verdadero entonces
a_{k+1} *es igual a* a_k
b_{k+1} *es igual a* r_{k+1}
Fin de lo que debe realizarse Si $[f(\boldsymbol{a_k}).f(\boldsymbol{r_{k+1}}) < 0$]es verdadero

Si $[f(\boldsymbol{b_k}).f(\boldsymbol{r_{k+1}}) < 0$]es verdadero entonces
a_{k+1} *es igual a* r_{k+1}
b_{k+1} *es igual a* b_k
Fin de lo que debe realizarse Si $[f(\boldsymbol{b_k}).f(\boldsymbol{r_{k+1}}) < 0$]es verdadero

Así se selecciona un nuevo intervalo que cumple con la condición de inicialización del método.

4.2 Método de Regula Falsi

- **Inicialización**

Es la misma condición que en el método de Bisección

- **Recurrencia**

La aproximación de la raíz se obtiene en la abscisa donde la Recta Lk que une los puntos $[a_k ; f(a_k)]$ y $[b_k ; f(b_k)]$ intersecta al eje de las abscisa X. Es decir,

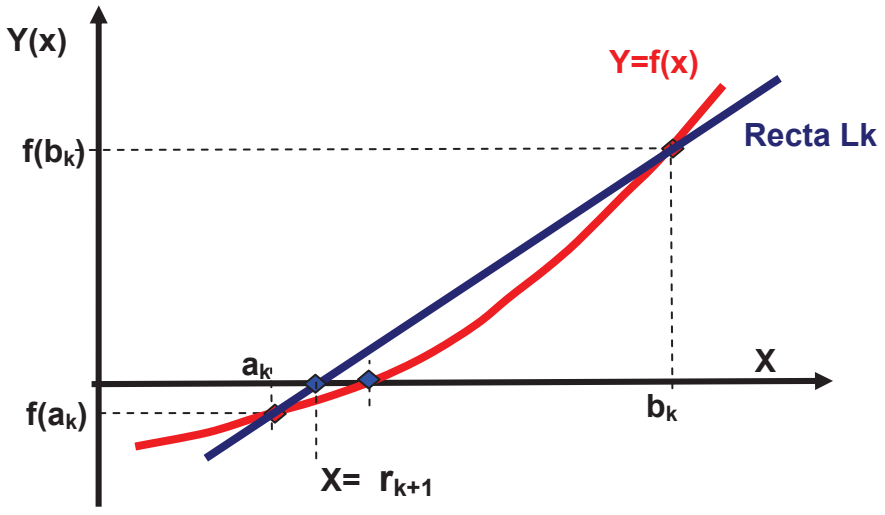
$$r_{k+1} = a_k - \frac{f(a_k)}{\left(\frac{f(a_k) - f(b_k)}{(a_k) - (b_k)}\right)}$$

O bien,

$$r_{k+1} = b_k - \frac{f(b_k)}{\left(\frac{f(a_k) - f(b_k)}{(a_k) - (b_k)}\right)}$$

Se debe destacar que:

- $m_k = \left(\frac{f(a_k) - f(b_k)}{(a_k) - (b_k)}\right)$ es la pendiente de la recta considerada, y su valor es independiente del punto que se toma como referencia para calcular el incremento de ordenada y el incremento de abscisa.
- r_{k+1} es invariante de que punto se tomo como referencia para expresar al ecuación de la recta que pasa por ese punto con pendiente conocida



- **Control de Detención**

Se debe realizar igual que en el método de Bisección

- **Actualización de Variables**

Se debe realizar igual que en el método de Bisección.

4.3 Método de la Secante

- **Inicialización**

Es necesario tener DOS aproximaciones anteriores. Es decir dos valores de abscisas r_{k-1}, r_k cercanas a las raíz que se busca.

- **Recurrencia**

La aproximación de la raíz se obtiene en la abscisa donde la Recta Lk que une los puntos $[r_{k-1} ; f(r_{k-1})]$ y $[r_k ; f(r_k)]$ intersecta al eje de las abscisa X. Es decir,

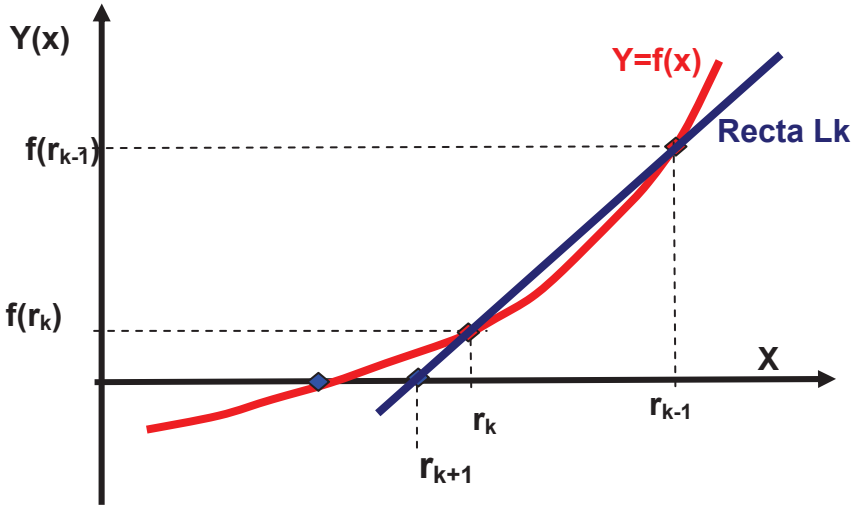
$$r_{k+1} = r_k - \frac{f(r_k)}{\left(\frac{f(r_k) - f(r_{k-1})}{(r_k) - (r_{k-1})}\right)}$$

O bien,

$$r_{k+1} = r_{k-1} - \frac{f(r_{k-1})}{\left(\frac{f(r_k) - f(r_{k-1})}{(r_k) - (r_{k-1})}\right)}$$

Se debe destacar que:

- $m_k = \left(\frac{f(r_k) - f(r_{k-1})}{(r_k) - (r_{k-1})}\right)$ es la pendiente de la recta considerada, y su valor es independiente del punto que se toma como referencia para calcular el incremento de ordenada y el incremento de abscisa.
- r_{k+1} es invariante de que punto se tomo como referencia para expresar al ecuación de la recta que pasa por ese punto con pendiente conocida



Se puede decir que los puntos $[r_{k-1} ; f(r_{k-1})]$ y $[r_k ; f(r_k)]$ son equivalentes a los $[a_k ; f(a_k)]$ y $[b_k ; f(b_k)]$ del método de Regula Falsi, a los efectos de la fórmula de recurrencia.

- **Control de Detención**

Se debe realizar igual que en el método de Bisección

- **Actualización de Variables**

Se deben retener las dos últimas aproximaciones. Esto es una ventaja respecto de los métodos de intervalo (Bisección y Regula Falsi) ya que no se requiere analizar los datos. Sólo basta tener dos aproximaciones.

4.4 Método de Newton Raphson

- **Inicialización**

Es necesario tener UNA aproximación de la raíz.

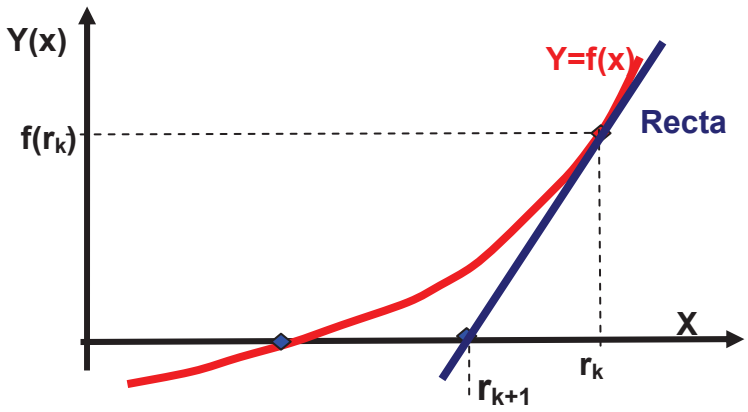
- **Recurrencia**

La aproximación de la raíz se obtiene en la abscisa donde la Recta Tangente Tk a la f(x) en el punto $[r_k ; f(r_k)]$ de abscisa r_k . intersecta al eje de las abscisa X. Es decir,

$$r_{k+1} = r_k - \frac{f(r_k)}{\left(\frac{df(x)}{dx}\right)\Big|_{r_k}} = r_k - \frac{f(r_k)}{(m_k)}$$

Se debe destacar que:

- $m_k = \left(\frac{df(x)}{dx}\right)\Big|_{r_k}$ es la pendiente de la recta tangente, evaluada en la aproximación de la raíz rk conocida.



Se debe considerar que la dirección de la Recta Tangente a una curva en un punto dado es las dirección de máximo cambio de dicha curva. Así el método de Newton Raphson es el método de mayor velocidad de acercamiento a la raíz buscada.

- **Control de Detención**

Se debe realizar igual que en el método de Bisección

- **Actualización de Variables**

Se deben retener la última aproximación obtenida.

4.5 Planteo Alternativo para el Método de Newton Raphson

Dada una función no lineal $y = F(x)$ se busca el valor de abscisa x_r tal que

$$F(x_r) = C$$

Con C una constante arbitraria y conocida. En la siguiente Figura pueden observar estas definiciones.

Esta ecuación es no lineal ya que F(x) es una función no lineal. Es posible escribir la ecuación no lineal en la forma:

$$\psi(x) = F(x) - C = 0$$

Dada una aproximación inicial x_k al evaluar $\psi(x_k)$ resulta que se tiene *un residuo* r_k dado por

$$r_k = \psi(x_k) \neq 0$$

Si se considera un expansión en Serie de Taylor de $\psi(x)$ alrededor de la abscisa x_k se tiene que

$$\psi(x_k + \Delta x) = \psi(x_k) + \Delta x \cdot \left(\frac{d\psi(x)}{dx} \right)_{Xk} + O(\Delta x^2)$$

Si se truncan los términos $O(\Delta x^2)$, y se busca la abscisa $x_{k+1} = x_k + \Delta x$ tal que la

$$\psi(x_{k+1}) = 0$$

es posible plantear

$$0 = \psi(x_k) + \Delta x \cdot \left(\frac{d\psi(x)}{dx} \right)_{Xk}$$

De donde se obtiene

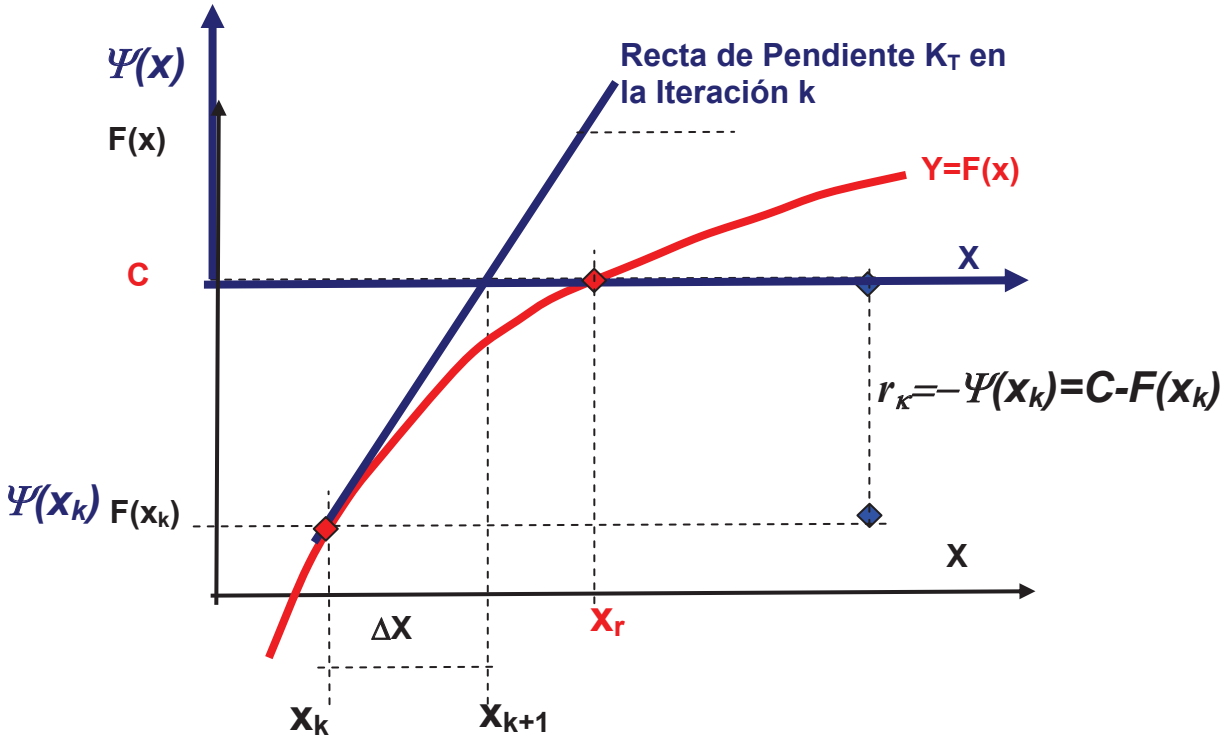
$$\Delta x = K_T^{-1} \cdot r_k = \left(\frac{d\psi(x)}{dx} \right)_{Xk}^{-1} \cdot (-\psi(x_k))$$

$$x_{k+1} = x_k + \Delta x$$

Con

- $r_k = -\psi(x_k) \neq 0$ denominado residuo en la iteración k.
- $K_T = \left(\frac{d\psi(x)}{dx} \right)_{Xk}$, denominado Tangente en la iteración k.

En la siguiente gráfica se pueden visualizar las variables y ecuaciones planteadas anteriormente.



Así la síntesis de la alternativa del Método de Newton Raphson resulta

- **Inicialización**

Es necesario tener UNA aproximación de la raíz.

- **Recurrencia**

La aproximación de la raíz se obtiene mediante,

$$\Delta x = K_T^{-1} \cdot r_k = \left(\frac{d\psi(x)}{dx} \right)^{-1}_{Xk} \cdot \psi(x_k)$$

$$x_{k+1} = x_k + \Delta x$$

Con

- $r_k = \psi(x_k) \neq 0$ denominado residuo en la iteración k.
- $K_T = \left(\frac{d\psi(x)}{dx} \right)_{Xk}$, denominado Tangente en la iteración k.

- **Control de Detención**

Se debe realizar igual que en el método de Bisección

- **Actualización de Variables**

Se deben retener la última aproximación obtenida.

4.6 Método de Punto Fijo

Dada una función no lineal $y = F(x)$ se busca el valor de abscisa x_r tal que

$$F(x_r) = C$$

Con C una constante arbitraria y conocida.

Esta ecuación es no lineal ya que F(x) es una función no lineal. Es posible escribir la ecuación no lineal en la forma:

$$\psi(x) = F(x) - C = 0$$

Si se multiplica esta igualdad por un número no nulo α arbitrario, se tiene

$$\alpha \cdot \psi(x) = \alpha \cdot (F(x) - C) = 0$$

Y si a esa igualdad se suma en ambos miembros x, se tiene

$$x = x + \alpha \cdot \psi(x)$$

O bien,

$$x = x + \alpha \cdot (F(x) - C)$$

Estas igualdades se pueden escribir en la forma

$$x = g(x)$$

donde

$$g(x) = x + \alpha \cdot (F(x) - C) = x + \alpha \cdot \psi(x)$$

La igualdad $x = g(x)$ se puede interpretar como la intersección de las siguientes funciones

$$\begin{cases} y = x \\ y = g(x) \end{cases}$$

Es decir de la recta que bisecta el primer cuadrante (y=x) con la curva y=g(x). Se debe destacar que el punto solución es tal que tiene abscisa y ordenadas de igual valor, y por lo tanto se lo denomina *Punto Fijo de a curva g(x).*

Es posible resolver la ecuación $x = g(x)$ mediante un esquema iterativo en la forma

- **Inicialización**

Es necesario tener UNA aproximación de la raíz.

- **Recurrencia**

La aproximación de la raíz se obtiene mediante,

$$x_k = g(x_k)$$

- **Actualización de Variables**

Se deben retener la última aproximación obtenida.

- **Control de Detención**

Se debe realizar igual que en el método de Bisección: Alternativamente calculada la nueva aproximación de la raíz x_{k+1}, se debe controlar si dicha abscisa es efectivamente igual a la ordenada $g(x_{k+1})$. Es decir se debe controlar ***si se cumple que***

$$\left| x_{k+1} - g(x_{k+1}) \right| \leq \varepsilon_f$$

Donde la barras indican valor absoluto y ε_f es una magnitud tan pequeña y cercana a cero como la precisión del problema a resolver lo requiera.

Es posible destacar que si se compara la definición de *g(x)* dada por

$$g(x) = x + \alpha \cdot \psi(x)$$

Y la fórmula de recurrrencia del método de punto fijo con la fórmula de recurrencia del método de Newton Raphson, se puede establecer que

$$\alpha = -\cdot \left(\frac{d\psi(x)}{dx} \right)^{-1}_{Xk} = -\frac{1}{\left(\frac{d\psi(x)}{dx} \right)_{Xk}}$$

Es decir que el método de Newton Raphson se puede interpretar como un método de Punto Fijo con el coeficiente α variable en cada iteración.

4.7 Condición de Convergencia del Método de Punto Fijo

Sea X_S el punto fijo de *g(x)*; es decir, la solución de la igualdad $x = g(x)$. Por lo tanto se tiene que

$$x_s = g(x_s)$$

Al considerar la fórmula de recurrencia del método de punto fijo se tiene que

$$x_{k+1} = g(x_k)$$

Al restar miembro a miembro estas dos igualdades, se tiene

$$x_{k+1} - x_s = g(x_k) - g(x_s)$$

En el segundo miembro, es posible aplicar el Teorema del Valor Medio, con lo que resulta

$$(x_{k+1} - x_s) = \left.\frac{dg(x)}{dx}\right|_{x=\xi} \cdot (x_k - x_s)$$

Siendo ξ una abscisa entre X_k y X_s. En el primer miembro se tiene ε_{k+1} el Error de la iteración k+1; mientras que en el segundo miembro se tiene ε_k el Error de la iteración k. Es decir,

$$\varepsilon_{k+1} = (x_{k+1} - x_s)$$

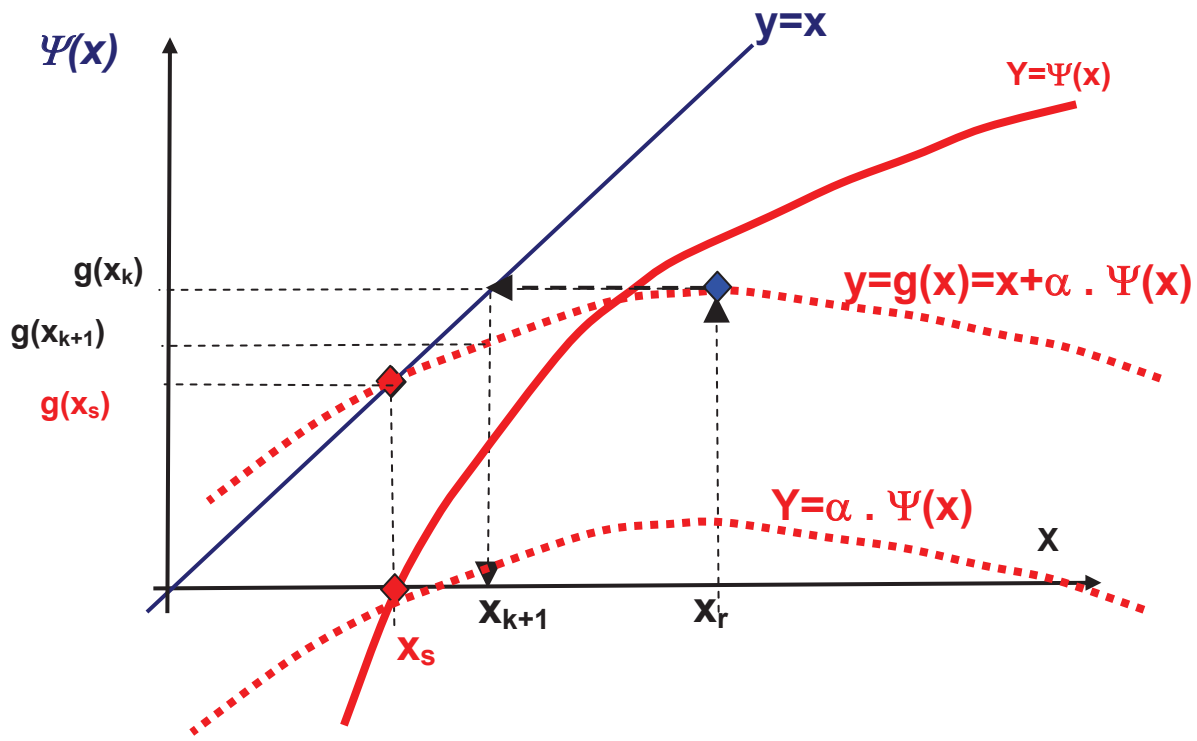
$$\varepsilon_k = (x_k - x_s)$$

Para que el ε_{k+1} Error de la iteración k+1 sea menor que el ε_k Error de la iteración k, se debe cumplir que el valor absoluto de la pendiente de la función *g(x)* en el entorno al punto fijo solución debe ser menor a uno. Así resulta que la condición de convergencia es:

$$\left|\left.\frac{dg(x)}{dx}\right|_{x=\xi}\right| < 1$$

El proceso iterativo del método de punto fijo convergerá si el error de una iteración es menor que el error de la iteración anterior .

En la siguiente Figura se ilustra el Método de Punto Fijo.



Sistemas de Ecuaciones Lineales

1 Introducción

En muchos problemas de ingeniería se requiere resolver sistemas de ecuaciones lineales. En cada ecuación del sistema, las incógnitas están combinadas linealmente con coeficientes constantes; y dicha combinación lineal está igualada a una constante conocida.
En el contexto del Algebra lineal, se puede interpretar un sistema de ecuaciones lineales como la obtención de los coeficientes (incógnitas del sistema) que combinan linealmente vectores de una base (columnas de la matriz de coeficientes) para generar un vector conocido (término independiente del sistema de ecuaciones).
Los distintos métodos computacionales para resolver ecuaciones diferenciales en forma discreta, en general conducen a sistemas de ecuaciones lineales de N ecuaciones con N incógnitas. Dichos métodos son de creciente aplicación en problemas de ingeniería y entre ellos se puede citar a los métodos de diferencias finitas, elementos finitos o volúmenes finitos. La particularidad de éstos métodos es que el orden N del sistema de ecuaciones suele ser muy grande (algunas centenas de miles o hasta millones). Es por ello que se debe recurrir a métodos eficientes para resolver los sistemas de ecuaciones lineales.
Básicamente se puede dividir a los métodos para resolver sistemas de ecuaciones lineales en dos grandes grupos: métodos de factorización y métodos iterativos.

2 Métodos de Factorización

Los métodos de factorización son particularmente útiles cuando la matriz de coeficientes del sistema de ecuaciones lineales tiene poco ceros (que se suele denominar matriz "llena"), o cuando con la misma matriz de coeficientes hay que resolver varios sistemas en los que cambia el término independiente.

Existen numerosos métodos de factorización que se basan en distintas propiedades de de la matriz de coeficientes del sistema de ecuaciones lineales o de sus matrices equivalentes. Entre los distintos métodos se puede citar el de Doolitlle, el de Crout, el de Cholesky, etc.

2.1 Método de Factorización LU

En el estudio de las matrices se demuestra que una matriz **A** se puede factorizar en términos de una matriz triangular inferior **L** (lower) y una triangular superior **U** (upper) si y sólo si se puede resolver de manera única el sistema lineal **A** $\underline{x}$ = $\underline{b}$ por eliminación de Gauss. Podemos utilizar esta propiedad entonces para resolver un SEL (sistema de ecuaciones lineales), a partir de lo siguiente:

Sea

$$\mathbf{A}\,\underline{\boldsymbol{x}} = \underline{b}$$

Pero

$$\mathbf{A} = \mathbf{L}\,\mathbf{U}$$

Entonces

$$(\mathbf{L}\,\mathbf{U})\,\underline{\boldsymbol{x}} = \underline{b}$$

Por propiedad asociativa podemos decir lo siguiente:

$$\mathbf{L}\,(\mathbf{U}\,\underline{\boldsymbol{x}}) = \underline{b}$$

Si se define

$$(\mathbf{U}\ \underline{\boldsymbol{x}}) = \underline{\boldsymbol{z}}_{\ nx1}$$

Resulta

$$\mathbf{L}\ \underline{\boldsymbol{z}} = \underline{b}$$

De donde se obtiene por **sustitución progresiva** el vector $\underline{\boldsymbol{z}}$.

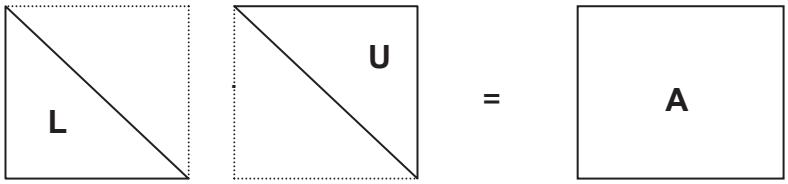
Y luego con

$$(\mathbf{U}\ \underline{\boldsymbol{x}}) = \underline{\boldsymbol{z}}$$

Por **sustitución regresiva** se obtiene el vector $\underline{\boldsymbol{x}}$.

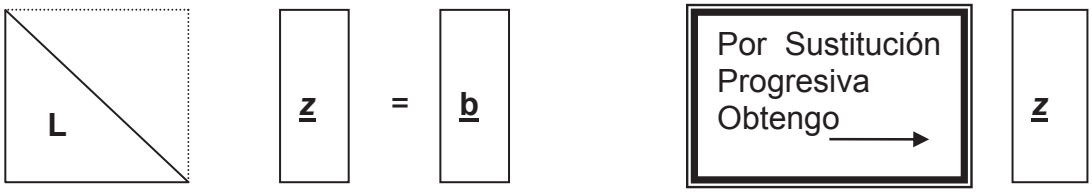
Gráficamente se puede sintetizar la idea general del método LU de la siguiente forma:

Primera Fase: Descomposición en LU



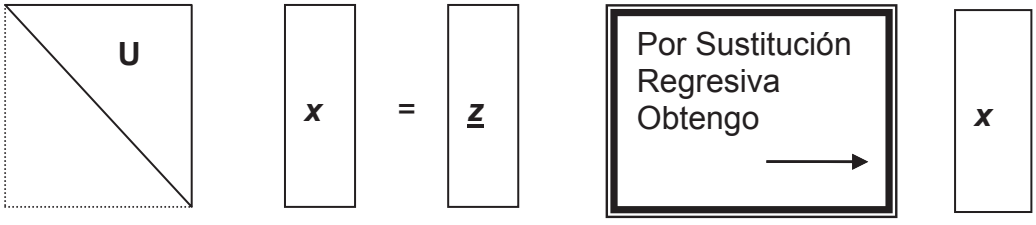
L U = A

Segunda Fase: Sustitución Progresiva



L $\underline{\boldsymbol{z}} = \underline{b}$

Tercer Fase: Sustitución Regresiva



U $\boldsymbol{x} = \underline{\boldsymbol{z}}$

La obtención de los coeficientes de las matrices triangulares **L** y **U** se basa en el método de eliminación de Gauss. La sistematización de la obtención de Ly U da origen a dos métodos de factorización: el de Crout y el de Doolittle.
En lo que sigue se adopta el método de Doolittle como método de factorización para

este curso.
Se debe destacar que cuando se cambia el término independiente del sistema de ecuaciones lineales, sólo se deben realizar las fases 2 y 3 para obtener la solución.
Así resulta de particular facilidad cuando se pretende calcular la matriz inversa de la matriz de coeficientes del sistema.

2.2 Método de Doolittle

Los NxN elementos de la matriz **A** son conocidos y se pueden usar para determinar aunque sea en forma parcial los elementos de **L** y de **U**, pero como debemos obtener una solución única se necesitan condiciones adicionales para los elementos de **L** y de **U**; las condiciones que utilizaremos son arbitrariamente las siguientes:

l_{ii} = 1 A la resolución con este condicionamiento se la conoce como "Método De Descomposición De Factores LU De Doolittle".

Se desea factorizar **A = L U** donde debido a las condiciones impuestas resulta que

$$\mathbf{L} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ l_{21} & 1 & 0 & 0 \\ l_{31} & l_{32} & 1 & 0 \\ l_{n1} & l_{n2} & \dots & 1 \end{pmatrix} \quad U = \begin{pmatrix} u_{11} & u_{12} & u_{13} & u_{1n} \\ 0 & u_{22} & u_{23} & u_{2n} \\ 0 & 0 & u_{33} & u_{3n} \\ 0 & 0 & 0 & u_{nn} \end{pmatrix}$$

Un elemento cualquiera de la matriz **A**, como el elemento a_{ij}, es igual al producto escalar de los vectores dados por la i-ésima fila de L y por la j-ésima columna de U. Es decir,

$$a_{ij} = (\mathbf{LU})_{ij} = [l_{i1}, l_{i2}, l_{i3}, \dots, l_{ii-1}, 1, 0, 0, \dots, 0] \cdot \begin{bmatrix} u_{1j} \\ u_{2j} \\ \dots \\ \dots \\ u_{jj} \\ 0 \\ \dots \\ \dots \\ 0 \end{bmatrix}$$

Así se tiene por cada elemento de la matriz A, una ecuación de donde calcular los coeficientes de L y de U.

Es muy conveniente seguir los siguientes pasos de solución.

Pasos de resolución:

1. Multiplicamos la fila i = 1 de **L** por todas las columnas de **U** para obtener:

$$u_{ij} = a_{ij} \text{ para } j = 1,2,\dots,n$$

2. Luego multiplico las filas de **L** (sin tener en cuenta la primera) por la 1º columna de **U**

$$l_{i1}\, u_{11} = a_{i1}$$

$$l_{ij} = \frac{a_{i1}}{u_{11}} \quad ; \text{para } i = 2,3,...n$$

3. Se sigue con la fila 2 de **L** por las columnas de **U** omitiendo la primera obteniendo así la 2º fila de **U**

$$u_{2j} = a_{2j} - l_{21}\, u_{1j} \; ; \text{ para } j=2,3,...,n$$

4. Multiplicando las filas de L (omitiendo la 1º y la 2º) por la 2º columna de U

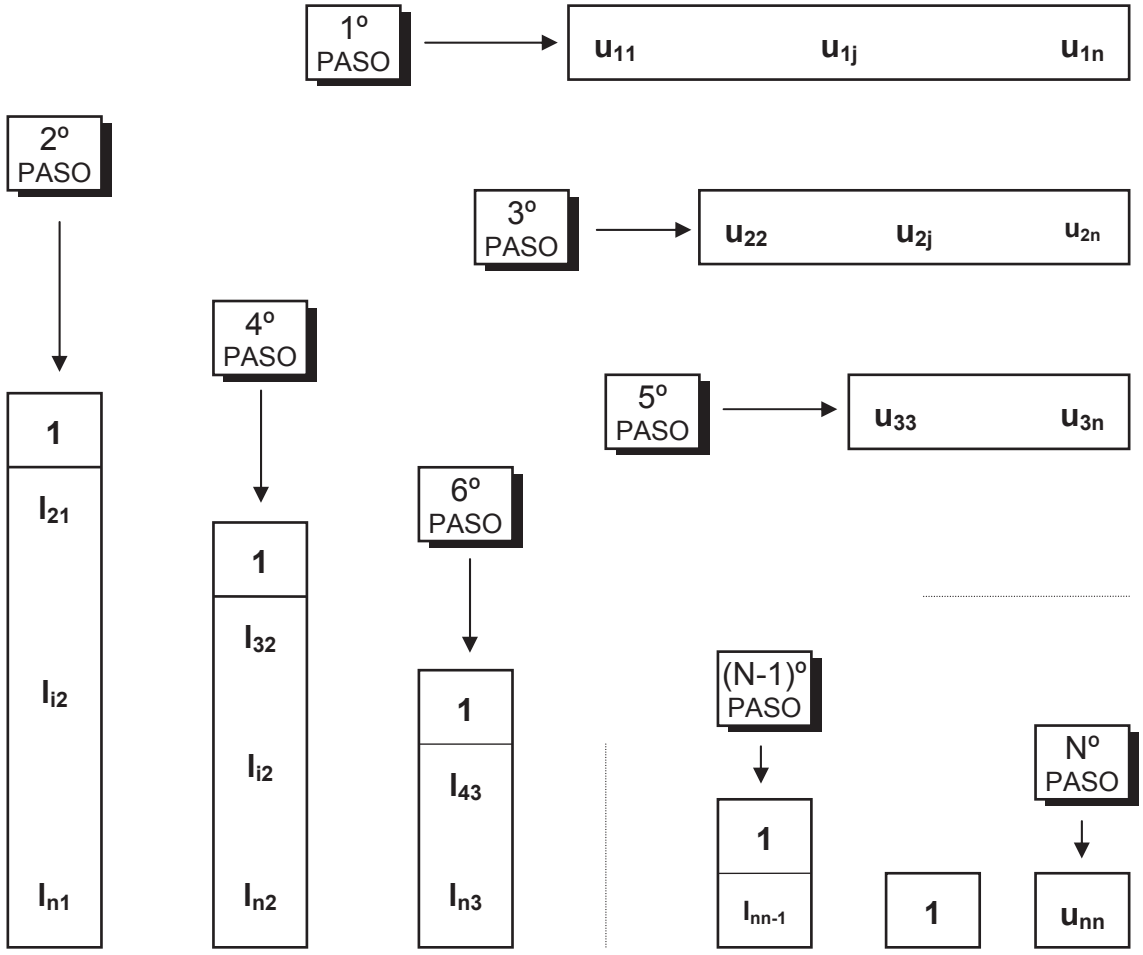
$$l_{i2} = \frac{1}{u_{22}}(a_{ij} - l_{i1}u_{12}) \quad \text{para } i = 3,4,...,n$$

En forma general podemos escribir que:

$$u_{rj} = a_{rj} - \sum_{k=1}^{r-1} l_{rk}\, u_{kj} \quad , \text{para } j = r, r+1,..., n$$

$$l_{ir} = \frac{1}{u_{rr}}(a_{ir} - \sum_{k=1}^{r-1} l_{ik}\, u_{kr} \quad , \text{para } i = r+1, r+2,..., n$$

Podríamos graficar los pasos de la siguiente manera:



Una manera de sintetizar el procedimiento es la siguiente:

1) Determinar los coeficientes de la primera fila de **U**, desde el elemento diagonal hacia la derecha. Luego los coeficientes de la primera columna de **L**, desde el elemento diagonal hacia abajo.

2) Determinar los coeficientes de la segunda fila de **U**, desde el elemento diagonal hacia la derecha. Luego los coeficientes de la segunda columna de **L**, desde el elemento diagonal hacia abajo.

3) Determinar los coeficientes de la tercera fila de **U**, desde el elemento diagonal hacia la derecha. Luego los coeficientes de la tercera columna de **L**, desde el elemento diagonal hacia abajo.

Y así seguir hasta terminar todas las filas y columnas.

2.3 Síntesis del Método de Doolittle

Dada la matriz **A** de NxN, se buscan los elementos de **L** y de **U**, de la siguiente forma

A = L U

Se deben realizar N Pasos, que se distinguen con l**a variación de r desde 1 hasta N**. Para un **Paso r** cualquiera se debe calcular los elementos de **U** con,

$$u_{rj} = a_{rj} - \sum_{k=1}^{r-1} l_{rk} \cdot u_{kj} \quad para \;\; j = r,(r+1),(r+2).....,N$$

Y los elementos de **L** con,

$$l_{ir} = \frac{(a_{ir} - \sum_{k=1}^{r-1} l_{ik} \cdot u_{kr})}{u_{rr}} \quad para \;\; i = r,(r+1),(r+2).....,N$$

Una vez determinadas L y U, se procede con:

Sustitución progresiva para obtener ***z*** mediante:

$$z_i = b_i - \sum_{k=1}^{i-1} l_{ik} \cdot z_k \quad con \; i = 1,2,.....,N$$

Y con la **Sustitución regresiva** para obtener ***x*** mediante:

$$x_i = \frac{(z_i - \sum_{k=i+1}^{N} u_{ik} \cdot x_k)}{u_{ii}} \quad con \; i = N,(N-1),(N-2).....2,1$$

2.4 Planteo del Método de Doolittle a partir de Matrices Elementales

Dada una matriz **A** es posible obtener una matriz equivalente de la forma triangular superior **U** (upper), mediante operaciones elementales de filas aplicadas en la matriz **A.** Dichas operaciones elementales de filas, consisten en reemplazar la i-ésima fila de la matriz por otra que resulta de combinación lineal de la i-ésima fila y otra fila de la matriz. Para no alterar el valor del determinante de la matriz original, es conveniente que el coeficiente de la combinación lineal de la i-ésima fila sea igual a 1, mientras que el coeficiente de la otra fila es tal que en la fila resultante aparezca algún elemento nulo.

Sea por ejemplo la siguiente matriz

$$\mathbf{A} = \begin{pmatrix} 4 & 8 & 4 \\ 2 & 2 & 3 \\ -1 & -3 & 0 \end{pmatrix}$$

Para obtener una matriz equivalente **Ar1,** en la que el elemento $Ar1_{21}$ sea nulo, es conveniente reemplazar la segunda fila de **A** (f_2) por la fila que resulta de la siguiente combinación lineal

(-2/4) fila1 de **A** + (1) fila2 de **A**→ fila2 de **Ar1**

esto es,

$$\begin{array}{r} (-2/4)\cdot(4 \quad 8 \quad 4) \\ (\ 1\ )\ \cdot\ (2 \quad 2 \quad 3) \\ \hline 0 \quad -2 \quad 1 \end{array}$$

Y dejando las restantes filas iguales. Así se obtiene la matriz

$$\mathbf{Ar1} = \begin{pmatrix} 4 & 8 & 4 \\ 0 & -2 & 1 \\ -1 & -3 & 0 \end{pmatrix}$$

Es posible relacionar las matrices **A** y su equivalente **Ar1,** mediante la denominada matriz elemental asociada a la combinación lineal de filas. Esta matriz elemental se obtiene de la matriz identidad en la que se reemplaza la fila 2 por una correspondiente a los coeficientes de la combinación lineal utilizada. Así la matriz elemental para este caso es

$$\mathbf{E1} = \begin{pmatrix} 1 & 0 & 0 \\ (-2/4) & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Se puede escribir la siguiente relación:

$$\mathbf{E1}\cdot\mathbf{A} = \mathbf{Ar1}$$

$$\begin{pmatrix} 1 & 0 & 0 \\ (-2/4) & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 4 & 8 & 4 \\ 2 & 2 & 3 \\ -1 & -3 & 0 \end{pmatrix} = \begin{pmatrix} 4 & 8 & 4 \\ 0 & -2 & 1 \\ -1 & -3 & 0 \end{pmatrix}$$

Análogamente es posible obtener una matriz equivalente **Ar2** de la matriz **Ar1**, tal que

su elemento $Ar2_{31}$ sea nulo. Se obtiene reemplazando la tercer fila de **Ar1** (f_3) por la fila que resulta de la siguiente combinación lineal

(1/4) fila1 de **Ar1** + (1) fila3 de **Ar1**→ fila3 de **Ar1**

esto es,

$$\begin{array}{r} (1/4)\cdot(\ 4 \quad 8 \quad 4) \\ (\ 1\ )\cdot(-1 \quad -3 \quad 0) \\ \hline 0 \quad -1 \quad 1 \end{array}$$

Obteniendose la matriz

$$\mathbf{Ar2} = \begin{pmatrix} 4 & 8 & 4 \\ 0 & -2 & 1 \\ 0 & -1 & 1 \end{pmatrix}$$

Así la matriz elemental para este caso es

$$\mathbf{E2} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ (1/4) & 0 & 1 \end{pmatrix}$$

Se puede escribir la siguiente relación:

$$\mathbf{E2}\cdot\mathbf{Ar1} = \mathbf{Ar2}$$

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ (1/4) & 0 & 1 \end{pmatrix}\cdot\begin{pmatrix} 4 & 8 & 4 \\ 0 & -2 & 1 \\ -1 & -3 & 0 \end{pmatrix} = \begin{pmatrix} 4 & 8 & 4 \\ 0 & -2 & 1 \\ 0 & -1 & 1 \end{pmatrix}$$

Al considerar la relación anterior entre la matriz **A** y su matriz equivalente **Ar1**, es posible asegurar que

$$\mathbf{E2}\cdot\mathbf{E1}\cdot\mathbf{A} = \mathbf{Ar2}$$

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ (1/4) & 0 & 1 \end{pmatrix}\cdot\begin{pmatrix} 1 & 0 & 0 \\ (-2/4) & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}\cdot\begin{pmatrix} 4 & 8 & 4 \\ 2 & 2 & 3 \\ -1 & -3 & 0 \end{pmatrix} = \begin{pmatrix} 4 & 8 & 4 \\ 0 & -2 & 1 \\ 0 & -1 & 1 \end{pmatrix}$$

Así se ha logrado "escalonar" la primera columna de la matriz **A** mediante la pre-multiplicación de la matriz **A** por matrices elementales.

Para escalonar la siguiente columna; es decir, para obtener una matriz **Ar3** equivalente **Ar2**, en la que el elemento $\mathbf{Ar3}_{32}$ sea nulo, es posible plantear la siguiente combinación lineal

(-1/2) fila2 de **Ar2** + (1) fila3 de **Ar2**→ fila3 de **Ar3**

esto es,

$$\begin{array}{r} (-1/2)\cdot(0 \quad -2 \quad 1) \\ (\ 1\ )\ \cdot\ (0 \quad -1 \quad 1) \\ \hline 0 \quad 0 \quad +1/2 \end{array}$$

Obteniendose la matriz

$$\mathbf{Ar3} = \begin{pmatrix} 4 & 8 & 4 \\ 0 & -2 & 1 \\ 0 & 0 & 1/2 \end{pmatrix}$$

La matriz elemental para este caso es

$$\mathbf{E3} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -1/2 & 1 \end{pmatrix}$$

Se puede escribir la siguiente relación:

$$\mathbf{E3 \cdot Ar2 = Ar3}$$

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -1/2 & 1 \end{pmatrix} \cdot \begin{pmatrix} 4 & 8 & 4 \\ 0 & -2 & 1 \\ 0 & -1 & 1 \end{pmatrix} = \begin{pmatrix} 4 & 8 & 4 \\ 0 & -2 & 1 \\ 0 & 0 & 1/2 \end{pmatrix}$$

Al considerar la relación anterior entre la matriz **A** y su matriz equivalente **Ar2**, es posible asegurar que

$$\mathbf{E3 \cdot E2 \cdot E1 \cdot A = Ar3}$$

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & (-1/2) & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ (1/4) & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 \\ (-2/4) & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 4 & 8 & 4 \\ 2 & 2 & 3 \\ -1 & -3 & 0 \end{pmatrix} = \begin{pmatrix} 4 & 8 & 4 \\ 0 & -2 & 1 \\ 0 & 0 & 1/2 \end{pmatrix}$$

Se obtiene así una matriz equivalente a la matriz **A**, que tiene la forma Triangular Superior, mediante la pre-multiplicación de la matriz **A** por matrices elementales, correspondientes a las operaciones elementales de filas realizadas.

Es posible agrupar en una única matiz **P** al producto de todas las matrices elementales utilizadas. Así se tiene que

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & (-1/2) & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ (1/4) & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 \\ (-2/4) & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot = \mathbf{P}$$

Y es posible escribir que

$$\mathbf{P \cdot A = U}$$

Siendo

$$\mathbf{U} = \mathbf{Ar3} = \begin{pmatrix} 4 & 8 & 4 \\ 0 & -2 & 1 \\ 0 & 0 & 1/2 \end{pmatrix}$$

Si se premultiplica la igualdad anterior por la inversa de la matriz **P**, resulta

$$\mathbf{P}^{-1} \cdot \mathbf{P} \cdot \mathbf{A} = \mathbf{P}^{-1} \cdot \mathbf{U}$$

O bien,

$$\mathbf{A} = \mathbf{P}^{-1} \cdot \mathbf{U}$$

La matriz $\mathbf{P^{-1}}$, resulta de hacer el producto de las inversas de las matrices elementales, que son muy simples de calcular. Para este caso,

$$\mathbf{P}^{-1} = \begin{pmatrix} 1 & 0 & 0 \\ (-2/4) & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}^{-1} \cdot \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ (1/4) & 0 & 1 \end{pmatrix}^{-1} \cdot \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & (-1/2) & 1 \end{pmatrix}^{-1}$$

$$\mathbf{P}^{-1} = \begin{pmatrix} 1 & 0 & 0 \\ (+2/4) & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ (-1/4) & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & (+1/2) & 1 \end{pmatrix}$$

$$\mathbf{P}^{-1} = \begin{pmatrix} 1 & 0 & 0 \\ (+2/4) & 1 & 0 \\ (-1/4) & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & (+1/2) & 1 \end{pmatrix}$$

$$\mathbf{P}^{-1} = \begin{pmatrix} 1 & 0 & 0 \\ (+2/4) & 1 & 0 \\ (-1/4) & (+1/2) & 1 \end{pmatrix}$$

La matriz $\mathbf{P^{-1}}$, resulta ser una matriz Triangular Inferior **L**, obtenida a partir de la inversa de matrices elementales.

$$\begin{pmatrix} 4 & 8 & 4 \\ 2 & 2 & 3 \\ -1 & -3 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ (+2/4) & 1 & 0 \\ (-1/4) & (+1/2) & 1 \end{pmatrix} \cdot \begin{pmatrix} 4 & 8 & 4 \\ 0 & -2 & 1 \\ 0 & 0 & 1/2 \end{pmatrix}$$

$$\mathbf{A} = \mathbf{L} \cdot \mathbf{U}$$

Se debe destacar que siempre es posible representar las operacioes elementales mediante matrices elementales que son triangulares inferiores., cuyas inversas también los son. Así la matriz $\mathbf{P^{-1}}$ resulta ser el producto de matrices triangulares inferiores y por lo tanto también será triangular inferior, y será la matriz **L** buscada en el método de Doolitle.

En síntesis: dada una matriz **A**, es posible obtener mediante pre multiplicación de la matriz **A** por matrices elementales **Ek,** una matriz equivalente de la matriz **A** que tiene la forma de escalonada reducida o triangular superior **U**. Donde cada matriz elemental **Ek**, representa la combinación lineal de filas realizada.

$$(\mathbf{Ek}\cdot\cdot.....\mathbf{E3}\cdot\mathbf{E2}\cdot\mathbf{E1})\cdot\mathbf{A}=\mathbf{U}$$

Siempre las inversas de las matrices elementales planteadas anteriormente, resultan simples de calcular cambiando el signo del coeficiente no nulo fuera de la diagonal.

Así la matriz original **A** se puede representar por

$$\mathbf{A}=(\mathbf{Ek}\cdot\cdot.....\mathbf{E3}\cdot\mathbf{E2}\cdot\mathbf{E1})^{-1}\cdot\mathbf{U}$$

$$\mathbf{A}=\mathbf{L}\cdot\mathbf{U}$$

$$\mathbf{L}=(\mathbf{Ek}\cdot\cdot.....\mathbf{E3}\cdot\mathbf{E2}\cdot\mathbf{E1})^{-1}$$

$$\mathbf{L}=\mathbf{E1}^{-1}\cdot\mathbf{E2}^{-1}\cdot\mathbf{E3}^{-1}\cdot.....\mathbf{Ek}^{-1}\cdot$$

Se debe destacar que esta forma de obtener las matrices L y U no es una forma suficientemente práctica para implementar computacionalmente. Es por ello, que la implementación computacional es conveniente realizarla mediante el procedimiento o algoritmo presentado en el punto anterior.

2.5 Cálculo de la Matriz Inversa Aplicando Doolittle

Dada la matriz **A** de NxN, se buscan la matriz inversa $\mathbf{A^{-1}}$, y se pretende utilizar la factorización **L** y **U**.

2.5.1 Alternativa Directa

Dado que

$$\mathbf{A^*\ A^{-1}{=}I}$$

con **I** matriz identidad de orden N. Entonces cada columna de la matriz inversa $\mathbf{A^{-1}}$ se obtiene de resolver un sistema de ecuaciones de la forma

$$\mathbf{A^*}\ \underline{\boldsymbol{a_k}} = \underline{\delta_k}$$

siendo $\underline{\boldsymbol{a_k}}$ la k-ésima columna de $\mathbf{A^{-1}}$; y $\underline{\delta_k}$ la k-ésima columna de la matriz identidad de orden N.

Al considerar la factorización LU, se puede obtener $\underline{\boldsymbol{z}}$ por **sustitución progresiva** mediante:

$$z_i=\delta_{k_i}-\sum_{k=1}^{i-1}l_{ik}\cdot z_k \qquad con\ i=1,2,.....,N$$

Siendo $\underline{\delta_k}$ la k-ésima columna de la matriz identidad de orden N, es decir un vector de

ceros, salvo el elemento k que es igual a uno.
Y con la **sustitución regresiva** para obtener $\underline{a_k}$, k-ésima columna de $\mathbf{A^{-1}}$; mediante:

$$a_{k\,i} = \frac{(z_i - \sum_{l=i+1}^{N} u_{il} \cdot a_{k\,l})}{u_{ii}} \qquad con\ i = N,(N-1),(N-2).....2,1$$

Es decir que con N sustituciones progresivas y regresivas se obtienen las columnas de la matriz inversa.

2.5.2 Alternativa Indirecta

Al conocer la factorización LU, y considerando que:

$$A^{-1} = U^{-1}\ L^{-1}$$

Se pude definir $L^{-1} = C$ y $U^{-1} = D$.
Es decir que L*C = I ; o sea, que cada columna de la matriz $L^{-1} = C$ se obtiene de la **sustitución progresiva** tomando como termino independiente cada vector columna de la matriz identidad.

$$C_{ii} = 1 \qquad para \quad i = 1,2,3.....,N$$

$$C_{ij} = -\sum_{k=j}^{i-1} l_{rk} \cdot C_{kj} \qquad con\ i = 2,....., N \quad j = 1,2,3,...(i-1)$$

Cada columna de la matriz $U^{-1} = D$ se obtiene de la **sustitución regresiva** tomando como termino independiente cada vector columna de la matriz identidad.

$$D_{ii} = 1/u_{ii} \qquad para\ \ i = N,(N-1),(N-2),.....2,1$$

$$D_{ij} = \frac{(-\sum_{l=i+1}^{j} u_{il} \cdot D_{lj})}{u_{ii}} \qquad con\ i = N,(N-1),(N-2).....2,1$$

$$j = N,(N-1),(N-2).....(i+1)$$

Así, las inversas de L y de U se obtienen por simples sustituciones hacia adelante y hacia atrás respectivamente.

3 Métodos Iterativos

Aunque casi siempre la eliminación de Gauss es la mejor técnica para resolver un SEL; cuando el número de ecuaciones crece y cuando la matriz tiene muchos ceros (matriz rala) hay otros métodos más simples, eficientes (en términos de computación, ya que insumen menor tiempo de proceso), y que logran alcanzar los objetivos en menor tiempo y con menos complejidad de cálculo.

3.1 Método de Jacobi

La primera de las técnicas que mostraremos se conoce como ***Iteración de Jacobi o Método de los Desplazamientos Simultáneos.***

Una técnica Iterativa para resolver un SEL de n x n

$$A\,\boldsymbol{x} = b$$

Empieza con una aproximación lineal $\boldsymbol{x}^0$ al vector solución ***x,*** y va generando una serie de vectores $\boldsymbol{x}^n$ con n desde 0 a infinito que converge hacia ***x.***

El ***Método de Jacobi*** (y la mayoría de este tipo de métodos) transforman al sistema A ***x*** =b en una forma equivalente

$$x = T\,x + c$$

que permite el siguiente proceso iterativo para obtener una solución aproximada desde un vector propuesto x^0, y para todo k mayor o igual a 1, con

$$\mathbf{x}_{(k)} = \mathbf{T}\,\mathbf{x}_{(k-1)} + \mathbf{c}$$

hasta tanto algún criterio de detención se cumpla.

Como criterios de detención se pueden adoptar medidas del error de la solución aproximada o bien algún número máximo de iteraciones.

Entonces, dado el siguiente SEL

$$a_{11}x_1 + a_{12}x_2 + a_{13}x_3 = b_1$$
$$a_{21}x_1 + a_{22}x_2 + a_{23}x_3 = b_2$$
$$a_{31}x_1 + a_{32}x_2 + a_{33}x_3 = b_3$$

Lo planteamos de la siguiente manera:

$$x_1 = \frac{1}{a_{11}}(b_1 - a_{11}\cdot 0 - a_{12}\cdot x_2 - a_{13}\cdot x_3)$$

$$x_2 = \frac{1}{a_{22}}(b_2 - a_{21}\cdot x_1 - a_{22}\cdot 0 - a_{23}\cdot x_3)$$

$$x_3 = \frac{1}{a_{33}}(b_3 - a_{31}\cdot x_1 - a_{32}\cdot x_2 - a_{33}\cdot 0)$$

Si conocemos una aproximación de la solución y la introducimos en los segundos miembros del SEL obtenemos un nuevo juego de valores para x_1, x_2 y x_3 que serán una mejor aproximación a la ***solución x***. Luego de una cantidad de sustituciones ***r*** nos habremos aproximado a la solución en función del máximo error que pretendíamos.

Hay que entender aquí que el método solo nos aproxima a la solución pero solo

la alcanzará si realizamos infinitas iteraciones.

Para el orden n:

Dado un SEL de n x n de la forma

$$a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \ldots + a_{1n}x_n = b_1$$
$$a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + \ldots + a_{2n}x_n = b_2$$
$$\vdots \qquad \vdots \qquad \vdots \qquad \vdots \qquad \vdots$$
$$a_{n1}x_1 + a_{n2}x_2 + a_{n3}x_3 + \ldots + a_{nn}x_n = b_n$$

El método consiste en transformar el sistema A $\boldsymbol{x}$ = b en uno de la forma $x_k = T\, x^{(k-1)} + c$ para luego resolver la i-ésima ecuación del SEL para x siempre y cuando a_{ii} sea distinto de cero, para obtener que

$$x_i = \sum_{j=1,\, j\neq i}^{n} \left(-\frac{a_{ij}.x_j}{a_{ii}} \right) + \frac{b_i}{a_{ii}}$$

Para i = 1, 2, ... ,n
Para $a_{ii} \neq 0$

Y generar cada $x_i^{(k)}$ de las componentes de $x_i^{(k-1)}$ para cada k ≥1 con

$$x_i^k = \sum_{j=1,\, j\neq i}^{n} \frac{(-a_{ij}.x_j^{k-1}) + b_i}{a_{ii}}$$

Con una tolerancia máxima ξ de:

$$\xi > \frac{\left\| x^k - x^{(k-1)} \right\|}{\left\| x^k \right\|} > 0$$

Consideraciones:

1) a_{ii} debe ser siempre distinto de cero; si no se deberá reorganizar el sistema de manera que esta condición se cumpla. Se recomienda que las ecuaciones se acomoden de manera que los valores de la diagonal sean lo mas grande posible, logrando de esta manera una convergencia mas veloz
2) Tolerancia o Error : el criterio para detener las iteraciones es:

$$\xi > \frac{\left\| x^k - x^{(k-1)} \right\|}{\left\| x^k \right\|} > 0$$

Siendo ξ el máximo error aceptado. Se puede utilizar cualquier norma y solo la conveniencia dicta cual debe utilizare. Por lo general se utiliza la norma infinita.

3) En general los métodos iterativos se utilizan para resolver SEL cuando el número de ecuaciones es grande (generalmente > 50) y/o cuando las matrices de coeficientes son ralas (muchos elementos nulos), ya que el tiempo necesario para alcanzar una solución suficientemente precisa requiere de muchas iteraciones.

4) Método de Gauss Seidel

Analicemos ahora el ***Método de Jacobi***, partiendo de una solución inicial $\boldsymbol{x^0}$ arbitraria o no, vamos obteniendo una serie de soluciones parciales que se aproximan al vector solución ***x***, sin embargo, es claramente visible que al resolver la primera ecuación con el vector solución parcial obtenemos un valor para x_1 (primera componente del vector solución siguiente) y es razonable suponer que este valor de la componente es mas cercano al valor final que el valor que utilizamos para calcularlo por lo cual podríamos introducirlo en las ecuaciones siguientes y de esta manera utilizar una mejor aproximación en la resolución de cada ecuación logrando una convergencia más veloz.

El método de ***Gauss Seidel*** parece ser mejor que el de ***Jacobi***; en muchos casos lo es pero **NO siempre**. Por otra parte hay sistemas que pueden ser resueltos por uno y no por el otro.

Definición:

El **Método de iterativo de GAUSS - SEIDEL** es entonces una mejora del **Método de Jacobi** en el que para calcular x_i^k en vez de utilizar los valores del vector x^{k-1} utilizamos los siguientes valores:

X_r^k desde r = 0 hasta r = i - 1

X_r^{k-1} desde r = i+1 hasta n

De manera que

$$x_i^k = \frac{-\sum_{j=1}^{i-1}(a_{ij}.x_j^{(k)}) - \sum_{j=i+1}^{n}(a_{ij}.x_j^{(k-1)}) + b_i}{a_{ii}}$$

Con $a_{ii} \neq 0$

I = 1,2,...,n

k = 1,2,...,n

Con una tolerancia máxima ξ de:

$$\xi > \frac{\left\| x^k - x^{(k-1)} \right\|}{\left\| x^k \right\|} > 0$$

Estudio de la convergencia

Las soluciones convergerán hacia el valor real de la solución del SEL bajo ciertas circunstancias. Haremos un estudio a modo introductorio de la convergencia de los métodos iterativos, utilizando un sistema de 2x2.

Para n = 2 (Ecuaciones). El sistema de ecuaciones a resolver es

$$a_{11}\,x + a_{12}\,y = b_1$$
$$a_{21}\,x + a_{22}\,y = b_2$$

La solución exacta (x,y) del sistema debe verificar las siguientes igualdades obtenidas

despejando de cada ecuación una de las variables

$$x = \frac{1}{a_{11}}(b_1 - a_{12}\, y) \qquad E1$$

$$y = \frac{1}{a_{22}}(b_2 - a_{21}\, x) \qquad E2$$

La forma algorítmica de calcular en cada iteración las aproximaciones de las incógnitas es:

$$x^{(k)} = \frac{1}{a_{11}}\left[b_1 - a_{12} y^{(k-1)}\right] \quad E3$$

$$y^{(k)} = \frac{1}{a_{22}}\left[b_2 - a_{21} x^{(k)}\right] \quad E4$$

Si definimos el error en la iteración k como el valor exacto menos el valor aproximado:

$$\Delta x^{(k)} = x - x^{(k)}$$

$$\Delta y^{(k)} = y - y^{(k)}$$

$$\Delta y^{(k-1)} = y - y^{(k-1)}$$

Restando $E1 - E3$

$$\Delta x^{(k)} = -\frac{a_{12}}{a_{11}} \Delta y^{(k-1)}$$

Restando $E2 - E4$

$$\Delta y^{(k)} = -\frac{a_{21}}{a_{22}} \Delta x^{(k)}$$

Si reeplazamos

$$\Delta x^{(k)} = \frac{a_{12} a_{21}}{a_{11} a_{22}} \Delta x^{(k-1)} \quad E5$$

$$\Delta x^{(k-1)} = \frac{a_{12} a_{21}}{a_{11} a_{22}} \Delta x^{(k-2)} \quad E6$$

Poniendo E6 en E5

$$\Delta x^{(k)} = \left[\frac{a_{12} a_{21}}{a_{11} a_{22}}\right]^2 \Delta x^{(k-2)}$$

En general

$$\Delta x^{(k)} = \left[\frac{a_{12} a_{21}}{a_{11} a_{22}}\right]^k \Delta x^{(0)}$$

$$\Delta y^{(k)} = \left[\frac{a_{12} a_{21}}{a_{11} a_{22}}\right]^k \Delta y^{(0)}$$

Si

$$\left|\frac{a_{12}a_{21}}{a_{11}a_{22}}\right| < 1$$

El proceso iterativo converge.

Esto se verifica si:

$|a_{11}| > |a_{12}|$

$|a_{22}| \geq |a_{21}|$

O sea que **los términos de la diagonal principal son dominantes.**

Si lo analizamos en forma gráfica por medio de un ejemplo con n = 2:

$2x + y = 2$

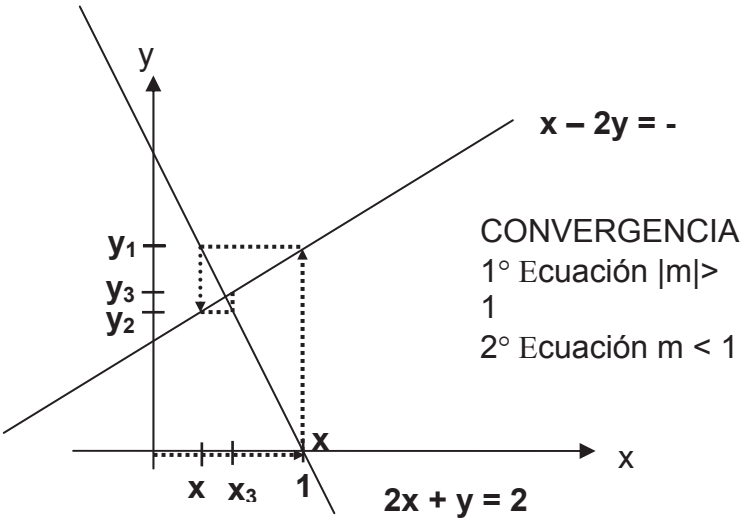
$x - 2y = -2$

Solución real (2/5,6/5)

Si iteramos empezando arbitrariamente por él (0,0) obtendríamos:

Iteración	***X***	***Y***
0	0	0
1	1	3/2
2	1/4	9/8
3	7/16	39/32

Ejercicio: Encontrar la matriz T y el vector c para este proceso iterativo



Invirtiendo las ecuaciones (su posición relativa).

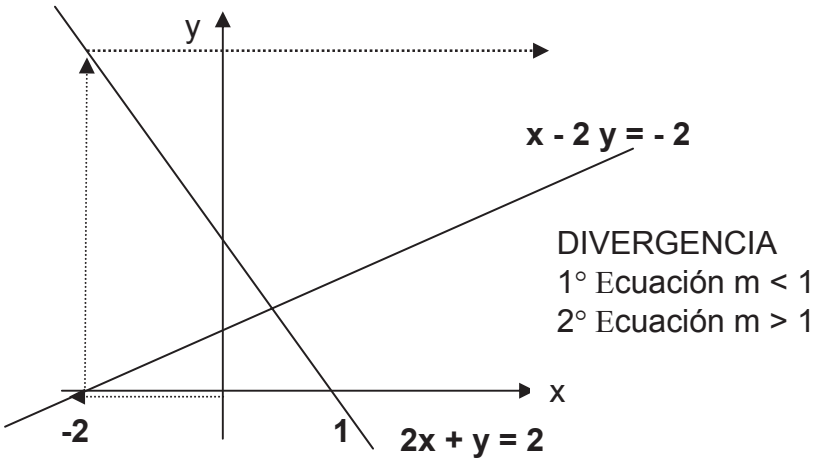
$x - 2y = -2$

$2x + y = 2$

Solución real (2/5,6/5)

Iteración	***X***	***Y***
0	0	0
1	-2	6
2	10	-18
3	-38	78

Ejercicio: Encontrar la matriz T y el vector c para este proceso iterativo



Entonces:

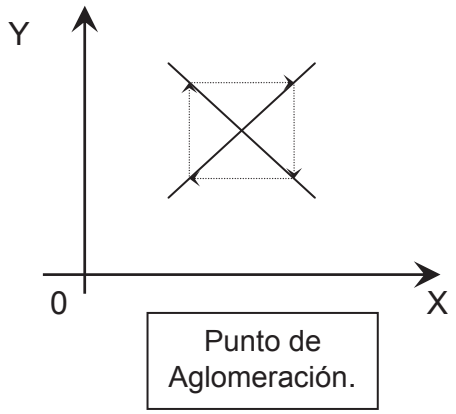
Se verifica que si $|m_1| \geq 1$, $|m_2| \leq 1$ y al menos una es una desigualdad estricta, entonces se cumple:

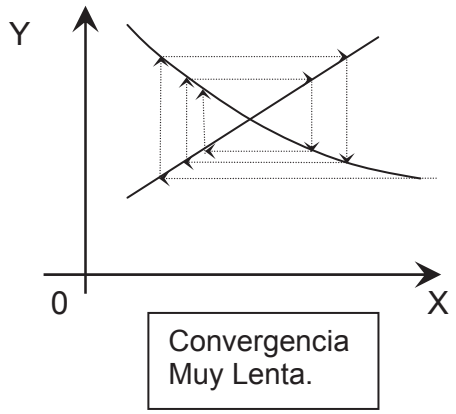
(a) Si m_1 y m_2 tienen igual signo, hay convergencia hacia un solo lado (como en el ejemplo).

(b) Si m_1 y m_2 tienen distinto signo, hay convergencia oscilante.

(c) Para n ecuaciones: si son irreductibles para cualquier i
$|a_{ii}| \geq |a_{i1}| + ... + |a_{ii-1}| + |a_{i+1}| + ... + |a_{in}|$
y si para al menos una i
$|a_{ii}| > |a_{i1}| + ... + |a_{ii-1}| + |a_{i+1}| + ... + |a_{in}|$ (sea desigualdad estricta)
Entonces el método converge.

(d) Grafiquemos ahora dos casos en que el sistema converge lentamente o cuando el sistema tiene una forma en que no converge pero no diverge. (Punto de aglomeración).





Es obvio que este tipo de problemas se puede dar con cualquier tipo de curvas.

4 Planteo Alternativo para el Método Iterativo de Jacobi

Dada una Matriz **A** ε R^{NxN}, un vector **b** ε R^N, el sistema de ecuaciones lineales asociado es

$$\mathbf{A} \cdot \underline{x} = \underline{b}$$

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} & .... & a_{1N} \\ a_{21} & a_{22} & a_{23} & .... & a_{2N} \\ a_{31} & a_{32} & a_{33} & .... & a_{3N} \\ .... & .... & .... & .... & .... \\ a_{N1} & a_{N2} & a_{N3} & .... & a_{NN} \end{pmatrix} \begin{Bmatrix} x_1 \\ x_2 \\ x_3 \\ ... \\ x_N \end{Bmatrix} = \begin{Bmatrix} b_1 \\ b_2 \\ b_3 \\ ... \\ b_N \end{Bmatrix}$$

Es posible escribirlo de la forma:

$$\mathbf{D} \cdot \underline{x} + \mathbf{B} \cdot \underline{x} = \underline{b}$$

Con

$$\begin{pmatrix} a_{11} & 0 & 0 & .... & 0 \\ 0 & a_{22} & 0 & .... & 0 \\ 0 & 0 & a_{33} & .... & 0 \\ .... & .... & .... & .... & .... \\ 0 & 0 & 0 & .... & a_{NN} \end{pmatrix} = \mathbf{D} \qquad \mathbf{B} = \begin{pmatrix} 0 & a_{12} & a_{13} & .... & a_{1N} \\ a_{21} & 0 & a_{23} & .... & a_{2N} \\ a_{31} & a_{32} & 0 & .... & a_{3N} \\ .... & .... & .... & 0 & .... \\ a_{N1} & a_{N2} & a_{N3} & .... & 0 \end{pmatrix}$$

$$\mathbf{D} \cdot \underline{x} = -\mathbf{B} \cdot \underline{x} + \underline{b}$$

$$\underline{x} = -\mathbf{D}^{-1} \cdot \mathbf{B} \cdot \underline{x} + \mathbf{D}^{-1} \cdot \underline{b}$$

$$\underline{x} = \mathbf{T} \cdot \underline{x} + \underline{c}$$

$$\mathbf{T} = -\mathbf{D}^{-1} \cdot \mathbf{B} \qquad \underline{c} = \mathbf{D}^{-1} \cdot \underline{b}$$

$$\mathbf{T} = \begin{pmatrix} 0 & -a_{12}/a_{11} & -a_{13}/a_{11} & .... & -a_{1N}/a_{11} \\ -a_{21}/a_{22} & 0 & -a_{23}/a_{22} & .... & -a_{2N}/a_{22} \\ -a_{31}/a_{33} & -a_{32}/a_{33} & 0 & .... & -a_{3N}/a_{33} \\ .... & .... & .... & 0 & .... \\ -a_{N1}/a_{NN} & -a_{N2}/a_{NN} & -a_{N3}/a_{NN} & .... & 0 \end{pmatrix} \quad \underline{c} = \begin{Bmatrix} b_1/a_{11} \\ b_2/a_{22} \\ b_3/a_{33} \\ ... \\ b_N/a_{NN} \end{Bmatrix}$$ Se

puede iterar con

$$\underline{x}^{(k+1)} = \mathbf{T} \cdot \underline{x}^{(k)} + \underline{c}$$

Hasta encontrar que el ERROR es tan pequeño como se quiera

5 Planteo Alternativo para el Método Iterativo de Gauss Seidel

A partir del método iterativo de Jacobi, cuyo fórmula de recurrencia está dada por

$$\underline{x}^{(k+1)} = \mathbf{T} \cdot \underline{x}^{(k)} + \underline{c}$$

$$\mathbf{T} = \begin{pmatrix} 0 & -a_{12}/a_{11} & -a_{13}/a_{11} & .... & -a_{1N}/a_{11} \\ -a_{21}/a_{22} & 0 & -a_{23}/a_{22} & .... & -a_{2N}/a_{22} \\ -a_{31}/a_{33} & -a_{32}/a_{33} & 0 & .... & -a_{3N}/a_{33} \\ .... & .... & .... & 0 & .... \\ -a_{N1}/a_{NN} & -a_{N2}/a_{NN} & -a_{N3}/a_{NN} & .... & 0 \end{pmatrix} \quad \underline{c} = \begin{Bmatrix} b_1/a_{11} \\ b_2/a_{22} \\ b_3/a_{33} \\ ... \\ b_N/a_{NN} \end{Bmatrix}$$

Se puede iterar con

$$\underline{x}^{(k+1)} = \mathbf{Tl} \cdot \underline{x}^{(k+1)} + \mathbf{Ts} \cdot \underline{x}^{(k)} + \underline{c}$$

Hasta encontrar que el ERROR es tan pequeño como se quiera.
Siendo

$$\mathbf{Tl} = \begin{pmatrix} 0 & 0 & 0 & .... & 0 \\ -a_{21}/a_{22} & 0 & 0 & .... & 0 \\ -a_{31}/a_{33} & -a_{32}/a_{33} & 0 & .... & 0 \\ .... & .... & .... & 0 & .... \\ -a_{N1}/a_{NN} & -a_{N2}/a_{NN} & -a_{N3}/a_{NN} & .... & 0 \end{pmatrix} \quad \underline{x}^{(k+1)} = \begin{Bmatrix} x_1^{(k+1)} \\ x_2^{(k+1)} \\ x_3^{(k+1)} \\ ... \\ x_N^{(k+1)} \end{Bmatrix}$$

$$\mathbf{Ts} = \begin{pmatrix} 0 & -a_{12}/a_{11} & -a_{13}/a_{11} & .... & -a_{1N}/a_{11} \\ 0 & 0 & -a_{23}/a_{22} & .... & -a_{2N}/a_{22} \\ 0 & 0 & 0 & .... & -a_{3N}/a_{33} \\ .... & .... & .... & 0 & .... \\ 0 & 0 & 0 & .... & 0 \end{pmatrix} \quad \underline{x}^{(k)} = \begin{Bmatrix} x_1^{(k)} \\ x_2^{(k)} \\ x_3^{(k)} \\ ... \\ x_N^{(k)} \end{Bmatrix}$$ Se

debe destacar que:

- para calcular x_1, participa todo el vector **x** de la iteración anterior.
- Para calcular x_2, participa x_1 de la iteración actual (recién calculado)y todas las demás componentes del vector **x** de la iteración anterior.
- Para calcular x_3, participa x_1 y x_2 de la iteración actual (recién calculadas)y todas las demás componentes del vector **x** de la iteración anterior.
- Para calcular x_4, participa x_1 ,x_2 y x_3 de la iteración actual (recién calculadas)y todas las demás componentes del vector **x** de la iteración anterior.
- Así se sigue hasta calcular x_N con todas las componentes de la iteración actual del vector **x** (recién calculadas), desde la 1 hasta la N-1.

VALORES Y VECTORES PROPIOS

1 INTRODUCCIÓN

Un problema de Valores y Vectores Propios consiste en encontrar los vectores $\underset{\sim}{v} \in \Re^N$ tales que son direcciones invariantes de la transformación lineal dada por la matriz

$$\mathbf{A} \in \Re^{NxN}$$

esto es

$$\mathbf{A}\,\underset{\sim}{v} = \lambda\,\underset{\sim}{v}$$

Siendo λ el escalar que cambia el módulo del vector $\underset{\sim}{v}$ cuya dirección permanece invariante. Se denomina autovalor λ y autovector $\underset{\sim}{v}$.

El sistema de ecuaciones puede escribirse de la forma:

$$(\mathbf{A} - \lambda \cdot \mathbf{I})\,\underset{\sim}{v} = \underset{\sim}{0}$$

e interesan las soluciones $\underset{\sim}{v} \neq \underset{\sim}{0}$ (distintas de la trivial ,v = 0). Esto está garantizado sí y sólo sí el $\det(\mathbf{A} - \lambda \cdot \mathbf{I}) = 0$. El determinante constituye un polinomio de grado N en el autovalor λ, y se denomina polinomio característico. Las raíces de dicho polinomio son los autovalores λ de la matriz **A** para los cuales existen los autovectores o direcciones invariantes $\underset{\sim}{v}$. Se los denomina también valores propios λ y vectores propios $\underset{\sim}{v}$. Por cada valor propio existe al menos una dirección invariante dada por el autovector $\underset{\sim}{v}$.

Existen diversos métodos para la determinación de los valores y vectores característicos que se los puede agrupar en las siguientes categorías:

- **Métodos de resolución del Polinomio Característico**
  Consisten en encontrar las raíces del polinomio característico y posteriormente resolver el sistema homogéneo para cada valor propio obtenido como raíz del polinomio característico. Este método es de utilidad práctica para sistemas de bajo orden (cuando N es pequeño. A medida que el orden N del sistema crece, también crece el orden del polinomio y con ello la dificultad para encontrar raíces.

- **Métodos de Trasformación**
  Consisten en transformar la matriz de coeficientes **A** en una matriz diagonal mediante procesos de rotaciones y/o traslaciones. Entre los métodos más eficaces de este tipo se encuentran los métodos de Jacobi, Givens y el de Householder. En estos métodos se encuentran la totalidad de los valores y vectores propios del sistema.

- **Métodos Iterativos**
  Consisten en aproximar sucesivamente los valores y vectores propios. En general los métodos convergen a un valor y vector propio. Para encontrar más de un vector y valor propio se debe recurrir a eliminar de los procesos iterativos los valores y vectores propios que ya se conocen

mediante técnicas de deflación. El método iterativo que se tratará en el curso es el **Método de las Potencias**.

2 MÉTODO DE LAS POTENCIAS

Es un método iterativo, también se lo conoce como Método de Iteración Matricial (Penzien, Clough), Método de Iteración Vectorial (Bathe) o Método de Stodola (quién lo aplicó en problemas de vibraciones) o de Vianello (en inestabilidad).

Como método de las potencias se lo encuentra en los textos de álgebra lineal de Strang, Grossman y en los de análisis numérico de Kincaid y Burden.

Dada una matriz $\mathbf{A} \in \Re^{NxN}$ se asume que

- $\lambda_i, \underset{\sim}{v_i}$ son los autovalores y autovectores de **A**. El índice i varía de 1 a N.

Se verifica que $\mathbf{A} \cdot \underset{\sim}{v_i} = \lambda_i \underset{\sim}{v_i} \quad i = 1, N$.

- λ_1 es el autovalor dominante, dado que

$$|\lambda_1| > |\lambda_2| \geq |\lambda_3| \cdots \geq |\lambda_N|.$$

- **A** es diagonalizable, es decir que los $\underset{\sim}{v_i}$ son **linealmente independientes** y forman una base de $\Re^N$,

$$\text{base de } \Re^N = \left\{ \underset{\sim}{v_1}, \underset{\sim}{v_2}, \underset{\sim}{v_3}, \ldots \underset{\sim}{v_N} \right\}.$$

Si $\left\{ \underset{\sim}{v_1}, \underset{\sim}{v_2}, \underset{\sim}{v_3}, \ldots \underset{\sim}{v_N} \right\}$ es una base de $\Re^N$ se puede escribir cualquier vector $\underset{\sim}{y_0} \in \Re^N$ como la combinación lineal

$$\underset{\sim}{y_0} = a_1 \underset{\sim}{v_1} + a_2 \underset{\sim}{v_2} + a_3 \underset{\sim}{v_3} + \ldots + a_N \underset{\sim}{v_N} = \sum_{i=1}^{N} a_i \underset{\sim}{v_i}$$

Si a $\underset{\sim}{y_0}$ se lo premultiplica por la matriz **A**, se obtiene un vector $\underset{\sim}{y_1} \in \Re^N$

$$\underset{\sim}{y_1} = \mathbf{A}\, \underset{\sim}{y_0}$$

o bien,

$$\underset{\sim}{y_1} = \mathbf{A}\, (a_1 \underset{\sim}{v_1} + a_2 \underset{\sim}{v_2} + a_3 \underset{\sim}{v_3} + \ldots + a_N \underset{\sim}{v_N})$$

$$\underset{\sim}{y_1} = (a_1\, \mathbf{A} \underset{\sim}{v_1} + a_2 \quad \mathbf{A} \underset{\sim}{v_2} + a_3 \quad \mathbf{A} \underset{\sim}{v_3} + \ldots + a_N \quad \mathbf{A} \underset{\sim}{v_N})$$

Considerando la definición de λ_i y $\underset{\sim}{v_i}$ se obtiene

$$\underset{\sim}{y_1} = a_1 \lambda_1 \underset{\sim}{v_1} + a_2 \lambda_2 \underset{\sim}{v_2} + a_3 \lambda_3 \underset{\sim}{v_3} + \ldots + a_N \lambda_N \underset{\sim}{v_N}$$

Que se puede expresar como

$$\underset{\sim}{y_1} = \lambda_1 \left[a_1 \underset{\sim}{v_1} + \frac{\lambda_2}{\lambda_1} a_2 \underset{\sim}{v_2} + \frac{\lambda_3}{\lambda_1} a_3 \underset{\sim}{v_3} + \ldots + \frac{\lambda_N}{\lambda_1} a_N \underset{\sim}{v_N} \right]$$

Si a $\underset{\sim}{y_1}$ se lo premultiplica por la matriz **A**, se obtiene $\underset{\sim}{y_2} \in \Re^N$

$$\underset{\sim}{y_2} = \mathbf{A}\, \underset{\sim}{y_1}$$

$$\underset{\sim}{y_2} = \mathbf{A}(a_1\lambda_1 \underset{\sim}{v_1} + a_2\lambda_2 \underset{\sim}{v_2} + a_3\lambda_3 \underset{\sim}{v_3} + \ldots + a_N\lambda_N \underset{\sim}{v_N})$$

o bien,

$$\underset{\sim}{y_2} = (a_1\lambda_1 \mathbf{A}\underset{\sim}{v_1} + a_2\lambda_2\mathbf{A} \underset{\sim}{v_2} + a_3\lambda_3\mathbf{A} \underset{\sim}{v_3} + \ldots + a_N\lambda_N\mathbf{A} \underset{\sim}{v_N})$$

Considerando la definición de λ_i y $\underset{\sim}{v_i}$ se obtiene

$$\underset{\sim}{y_2} = a_1\lambda_1^{\,2} \underset{\sim}{v_1} + a_2\lambda_2^{\,2} \underset{\sim}{v_2} + a_3\lambda_3^{\,2} \underset{\sim}{v_3} + \ldots + a_N\lambda_N^{\,2} \underset{\sim}{v_N}$$

$$\underset{\sim}{y_2} = \lambda_1^{\,2} \left[a_1 \underset{\sim}{v_1} + \left(\frac{\lambda_2}{\lambda_1}\right)^2 a_2 \underset{\sim}{v_2} + \left(\frac{\lambda_3}{\lambda_1}\right)^2 a_3 \underset{\sim}{v_3} + \ldots + \left(\frac{\lambda_N}{\lambda_1}\right)^2 a_N \underset{\sim}{v_N} \right]$$

Si este proceso se repite 'k' veces, se obtiene

$$\underset{\sim}{y_k} = \mathbf{A}\ \underset{\sim}{y_{k-1}}$$

$$\underset{\sim}{y_k} = \lambda_1^{\,k} \left[a_1 \underset{\sim}{v_1} + \underbrace{\left(\frac{\lambda_2}{\lambda_1}\right)^k a_2 \underset{\sim}{v_2} + \left(\frac{\lambda_3}{\lambda_1}\right)^k a_3 \underset{\sim}{v_3} + \ldots + \left(\frac{\lambda_N}{\lambda_1}\right)^k a_N \underset{\sim}{v_N}}_{\underset{\sim}{\varepsilon_k}} \right]$$

$$\underset{\sim}{y_k} = \lambda_1^{\,k} \left[a_1 \underset{\sim}{v_1} + \underset{\sim}{\varepsilon_k} \right]$$

siendo

$$\underset{\sim}{\varepsilon_k} = \left(\frac{\lambda_2}{\lambda_1}\right)^k a_2 \underset{\sim}{v_2} + \left(\frac{\lambda_3}{\lambda_1}\right)^k a_3 \underset{\sim}{v_3} + \ldots + \left(\frac{\lambda_N}{\lambda_1}\right)^k a_N \underset{\sim}{v_N}$$

Al considerar que

$$|\lambda_1| > |\lambda_2| > |\lambda_3| \cdots > |\lambda_N|$$

se cumple que

$$\lim_{k\to\infty} \underset{\sim}{\varepsilon_k} = \underset{\sim}{0}$$

es decir que, a partir de **cualquier** $\underset{\sim}{y_0} \in \Re^N$ por sucesivas premultiplicaciones por $\mathbf{A} \in \Re^{NxN}$ los vectores que se obtienen **se van alineando** cada vez más con el **autovector dominante** $\underset{\sim}{v_1}$.

Así

$$\underset{\sim}{y}_k \cong \lambda_1^k a_1 \underset{\sim}{v}_1$$
$$\underset{\sim}{y}_{k+1} \cong \lambda_1^{k+1} a_1 \underset{\sim}{v}_1$$

Si se hace el cociente entre componentes

$$\alpha_j(k+1) \cong \frac{\lambda_1^{k+1} a_1 \left(\underset{\sim}{v_1} \right)_j}{\lambda_1^k a_1 \left(\underset{\sim}{v_1} \right)_j}$$
$$\alpha_j(k+1) \cong \lambda_1$$

Es decir que para 'k' suficientemente grande el cociente de las componentes entre los vectores de dos iteraciones sucesivas, resulta el autovalor dominante λ_1.
Esta afirmación es natural si se entiende que el vector de cada iteración se va alineando con $\underset{\sim}{v_1}$, así por la definición misma de $\left(\lambda_1, \underset{\sim}{v_1} \right)$ los sucesivos productos por **A** mantienen la dirección de $\underset{\sim}{v_1}$ cambiando su módulo λ_1 veces.

Se debe notar que:

- Hay convergencia al autovalor λ_1 de mayor valor absoluto.
- Si $a_1 = 0$ la convergencia no se produce o bien es muy lenta.
- Así planteado el proceso, para 'k' grandes los valores de las componentes de $\underset{\sim}{y_k}$ pueden ser muy grandes si $|\lambda_1| > 1$ o muy chicas si $|\lambda_1| < 1$ ya que son proporcionales a λ_1^k.
  Para evitar operar con números cada vez más grandes (o más chicos) se debe utilizar un escalamiento en cada iteración.

2.1 PROCESO ITERATIVO CON ESCALAMIENTO

Cualquier vector $\underset{\sim}{y_0} \in \Re^N$ se puede escribir como una combinación lineal de los autovectores $\underset{\sim}{v_i} \in \Re^N$, asociados a los autovalores λ_i de la matriz $\mathbf{A} \in \Re^{NxN}$.

$$\underset{\sim}{y_0} = a_1 \underset{\sim}{v_1} + a_2 \underset{\sim}{v_2} + ... + a_N \underset{\sim}{v_N}$$

Es posible introducir una Norma $\left\| \underset{\sim}{y_0} \right\|$ de modo que se pueda obtener un vector

$$\underset{\sim}{x_0} = \frac{1}{\left\| \underset{\sim}{y_0} \right\|} \underset{\sim}{y_0}$$

de manera que

$$\left\| \underset{\sim}{x_0} \right\| = 1$$

Puede considerarse como **norma** alguna de las siguientes:

1. $\left\| \underset{\sim}{y_0} \right\|_\infty = \textit{Máxima componente de } \underset{\sim}{y_0}$, tomadas en valor absoluto
2. $\left\| \underset{\sim}{y_0} \right\|_2 = \left\langle \underset{\sim}{y_0}, \underset{\sim}{y_0} \right\rangle^{1/2} = \sqrt{\underset{\sim}{y_0}^T . \underset{\sim}{y_0}}$
3. $\left\| \underset{\sim}{y_0} \right\|_{\mathbf{B}} = \left(\underset{\sim}{y_0}^T \ \mathbf{B} \ \underset{\sim}{y_0} \right)^{1/2}$ con $\mathbf{B} \in \Re^{NxN}$

Observación: si bien la correspondencia que asigna a cada vector su primera componente no es una norma, en la práctica se puede utilizar la primera componente de cada vector en lugar de una norma determinada, en el proceso de escalamiento que se presentará a continuación.

El proceso iterativo consiste en premultiplicar a los vectores $\underset{\sim}{x}$ por la matriz **A** y obtener

$$\underset{\sim}{y_1} = A \ \underset{\sim}{x_0}$$

$$\underset{\sim}{y_1} = \frac{\mathbf{A}}{\left\| \underset{\sim}{y_0} \right\|} \left[a_1 \underset{\sim}{v_1} + a_2 \underset{\sim}{v_2} + ... + a_N \underset{\sim}{v_N} \right]$$

$$\underset{\sim}{y}_1 = \frac{1}{\left\| \underset{\sim}{y}_0 \right\|} \left[a_1 \ \mathbf{A} \underset{\sim}{v}_1 + a_2 \ \mathbf{A} \underset{\sim}{v}_2 + \ldots + a_N \ \mathbf{A} \underset{\sim}{v}_N \right]$$

Que considerando la definición de valores y vectores propios de la matriz **A,** resulta:

$$\underset{\sim}{y}_1 = \frac{1}{\left\| \underset{\sim}{y}_0 \right\|} \left[\lambda_1 a_1 \underset{\sim}{v}_1 + \lambda_2 a_2 \underset{\sim}{v}_2 + \ldots + \lambda_N a_N \underset{\sim}{v}_N \right]$$

Es posible sacar factor común el mayor autovalor,

$$\underset{\sim}{y}_1 = \frac{\lambda_1}{\left\| \underset{\sim}{y}_0 \right\|} \left[a_1 \underset{\sim}{v}_1 + \left(\frac{\lambda_2}{\lambda_1} \right) a_2 \underset{\sim}{v}_2 + \ldots + \left(\frac{\lambda_N}{\lambda_1} \right) a_N \underset{\sim}{v}_N \right]$$

Es posible evaluar el cociente entre las componentes de $\underset{\sim}{x}_0$ y el vector $\underset{\sim}{y}_1$,

$$\left(\alpha_1 \right)_i = \left(\underset{\sim}{y}_1 \right)_i \Big/ \left(\underset{\sim}{x}_0 \right)_i$$

$$\left(\alpha_1 \right)_i = \frac{\lambda_1}{\left\| \underset{\sim}{y}_0 \right\|} \frac{\left[a_1 \left(\underset{\sim}{v}_1 \right)_i + \left(\frac{\lambda_2}{\lambda_1} \right) a_2 \left(\underset{\sim}{v}_2 \right)_i + \ldots + \left(\frac{\lambda_N}{\lambda_1} \right) a_N \left(\underset{\sim}{v}_N \right)_i \right]}{\frac{1}{\left\| \underset{\sim}{y}_0 \right\|} \left[a_1 \left(\underset{\sim}{v}_1 \right)_i + a_2 \left(\underset{\sim}{v}_2 \right)_i + \ldots + a_N \left(\underset{\sim}{v}_N \right)_i \right]}$$

Se evalúa $\left\| \underset{\sim}{y}_1 \right\|$ y se obtiene el vector normalizado

$$\underset{\sim}{x}_1 = \underset{\sim}{y}_1 \frac{1}{\left\| \underset{\sim}{y}_1 \right\|}$$

con el que se continua el ciclo iterativo. Es decir, se obtiene:

$$\underset{\sim}{y}_2 = \quad \mathbf{A} \ \underset{\sim}{x}_1$$

Que resulta equivalente a

$$\underset{\sim}{y}_2 = \frac{1}{\left\| \underset{\sim}{y}_1 \right\|} \quad \mathbf{A} \ \underset{\sim}{y}_1$$

$$\underset{\sim}{y}_2 = \frac{1}{\left\| \underset{\sim}{y}_1 \right\|} \frac{\lambda_1}{\left\| \underset{\sim}{y}_0 \right\|} \left[a_1 \quad \mathbf{A}\underset{\sim}{v}_1 + \left(\frac{\lambda_2}{\lambda_1}\right) a_2 \quad \mathbf{A}\underset{\sim}{v}_2 + \ldots + \left(\frac{\lambda_N}{\lambda_1}\right) a_N \quad \mathbf{A}\underset{\sim}{v}_N \right]$$

$$\underset{\sim}{y}_2 = \frac{1}{\left\| \underset{\sim}{y}_1 \right\|} \frac{\lambda_1^{\,2}}{\left\| \underset{\sim}{y}_0 \right\|} \left[a_1 \underset{\sim}{v}_1 + \left(\frac{\lambda_2}{\lambda_1}\right)^2 a_2 \underset{\sim}{v}_2 + \ldots + \left(\frac{\lambda_N}{\lambda_1}\right)^2 a_N \underset{\sim}{v}_N \right]$$

El cociente entre componentes es

$$(\alpha_2)_i = \left(\underset{\sim}{y}_2\right)_i \Big/ \left(\underset{\sim}{x}_1\right)_i$$

$$(\alpha_2)_i = \frac{\dfrac{1}{\left\| \underset{\sim}{y}_1 \right\|} \dfrac{\lambda_1^{\,2}}{\left\| \underset{\sim}{y}_0 \right\|} \left[a_1 \left(\underset{\sim}{v}_1\right)_i + \left(\frac{\lambda_2}{\lambda_1}\right)^2 a_2 \left(\underset{\sim}{v}_2\right)_i + \ldots + \left(\frac{\lambda_N}{\lambda_1}\right)^2 a_N \left(\underset{\sim}{v}_N\right)_i \right]}{\dfrac{1}{\left\| \underset{\sim}{y}_1 \right\|} \dfrac{\lambda_1}{\left\| \underset{\sim}{y}_0 \right\|} \left[a_1 \left(\underset{\sim}{v}_1\right)_i + \left(\frac{\lambda_2}{\lambda_1}\right) a_2 \left(\underset{\sim}{v}_2\right)_i + \ldots + \left(\frac{\lambda_N}{\lambda_1}\right) a_N \left(\underset{\sim}{v}_N\right)_i \right]}$$

$$(\alpha_2)_i = \frac{\lambda_1^{\,2}}{\lambda_1} \frac{\left[a_1 \left(\underset{\sim}{v}_1\right)_i + \left(\frac{\lambda_2}{\lambda_1}\right)^2 a_2 \left(\underset{\sim}{v}_2\right)_i + \ldots + \left(\frac{\lambda_N}{\lambda_1}\right)^2 a_N \left(\underset{\sim}{v}_N\right)_i \right]}{\left[a_1 \left(\underset{\sim}{v}_1\right)_i + \left(\frac{\lambda_2}{\lambda_1}\right) a_2 \left(\underset{\sim}{v}_2\right)_i + \ldots + \left(\frac{\lambda_N}{\lambda_1}\right) a_N \left(\underset{\sim}{v}_N\right)_i \right]}$$

Luego de 'k' iteraciones se tendrá:

- $\underset{\sim}{x}_k = \dfrac{1}{\left\| \underset{\sim}{y}_k \right\|} \underset{\sim}{y}_k$
- $\underset{\sim}{y}_{k+1} = \mathbf{A}\ \underset{\sim}{x}_k$

El cociente entre componentes será

- $(\alpha_{k+1})_i = \left(\underset{\sim}{y}_{k+1}\right)_i \dfrac{1}{\left(\underset{\sim}{x}_k\right)_i}$

o bien,

$$\left(\alpha_{k+1}\right)_i = \frac{\lambda_1^{k+1}}{\lambda_1^k}\left[\frac{a_1\left(\underset{\sim}{v}_1\right)_i + \left(\varepsilon_{k+1}\right)_i}{a_1\left(\underset{\sim}{v}_1\right)_i + \left(\varepsilon_k\right)_i}\right]$$

siendo

$$\lim_{k\to\infty}\left(\varepsilon_k\right)_i = \lim_{k\to\infty}\left[\left(\frac{\lambda_2}{\lambda_1}\right)^k a_2\left(\underset{\sim}{v}_2\right)_i + \ldots + \left(\frac{\lambda_N}{\lambda_1}\right)^k a_N\left(\underset{\sim}{v}_N\right)_i\right]$$

y así el cociente entre componentes resulta:

$$\left(\alpha_{k+1}\right)_i = \frac{\lambda_1^{k+1}}{\lambda_1^k}\frac{a_1\left(\underset{\sim}{v}_1\right)_i}{a_1\left(\underset{\sim}{v}_1\right)_i} \cong \lambda_1$$

Es decir que el cociente entre cualquier componente i tiende a λ_1 cuando el número de iteraciones k tiende a infinito.

Así luego de 'k' iteraciones los vectores $\underset{\sim}{y}_k$ se alinean con el autovector $\underset{\sim}{v}_1$ y el cociente entre componentes tiende al autovalor λ_1. El escalamiento en cada iteración logra que las componentes de los vectores no aumenten (disminuyan) excesivamente, asegurando la convergencia.

Se debe destacar que:

1. el escalamiento puede hacerse con cualquiera de las normas de $\underset{\sim}{y}$ definidas;
2. el cociente entre las normas de $\underset{\sim}{y}$ de dos iteraciones sucesivas, converge al $|\lambda_1|$ cuando k es suficientemente grande $(k\to\infty)$. Para demostrarlo se debe seguir el mismo análisis que el cociente entre las componentes de $\underset{\sim}{y}$ y $\underset{\sim}{x}$ realizado.

$$\underset{\sim}{y}_k = \frac{\lambda_1^k}{\left\|\underset{\sim}{y}_{k-1}\right\|\left\|\underset{\sim}{y}_1\right\|\left\|\underset{\sim}{y}_0\right\|}\left[a_1\,\underset{\sim}{v}_1 + \underset{\sim}{\varepsilon}_k\right]$$

$$\left\| y_k \right\|_2^2 = \left(y_k \right)_i \left(y_k \right)_i = \left[\frac{\lambda_1^k}{\left\| y_{k-1} \right\| \cdots \left\| y_1 \right\| \left\| y_0 \right\|} \right]^2 \left[\left(a_1 \left(v_1 \right)_i + \left(\varepsilon_k \right)_i \right) \left(a_1 \left(v_1 \right)_i + \left(\varepsilon_k \right)_i \right) \right]$$

$$\left\| y_k \right\|_2 = \frac{\sqrt{\left(\lambda_1^k\right)^2}}{\left\| y_{k-1} \right\| \cdots \left\| y_1 \right\| \left\| y_0 \right\|} \left[a_1 \left(v_1 \right)_i \; a_1 \left(v_1 \right)_i + 2 a_1 \left(v_1 \right)_i \left(\varepsilon_k \right)_i + \left(\varepsilon_k \right)_i \left(\varepsilon_k \right)_i \right]^{1/2}$$

recordar que $\lim_{k \to \infty} \left(\varepsilon_k \right)_i = \underline{0}$ y sólo serán significativas las componentes de v_1 .

2.2 SINTESIS DEL MÉTODO DE LA POTENCIA

1) Lectura de datos:
   Leer la matriz **A**
   Asumir e vector inicial y_0
2) Inicialización del proceso iterativo
   Cálculo de la Norma de y_0
   Cálculo del vector normalizado x_0
   Inicio del contador de iteraciones k en Cero.
   Inicio de la variable lógica "NHS"en falso
3) **Hacer Mientras** "NHS" sea falso
   Incrementar el contador de iteraciones k en 1
   Calcular y_{k+1}=**A** x_k
   Calcular los cocientes entre componentes de alfa(i)= $y_{k+1}(i)/ x_k(i)$
   Si las componentes alfa(i) son "suficientemente parecidas" entonces
   "NHS" se asigna Verdadero
   Fin SI
   Actualización de variables
   Cálculo de la Norma de y_{k+1}
   Cálculo del vector normalizado x_k como el y_{k+1} dividido la Norma de y_{k+1}
   Fin de Actualización
   Fin del Hacer mientras
4) Entregar los resultados
   El número de iteraciones realizadas fue k
   El autovector obtenido es y_{k+1}
   El autovector normalizado obtenido es x_k
   El mayor autovalor obtenido es alfa(i)

3 PROCESO DE ITERACIÓN INVERSA

La matriz $\mathbf{A} \in \Re^{NxN}$ es diagonizable y sus valores propios $\lambda_i \in \Re$ y $\underset{\sim}{v_i} \in \Re^N$ verifican

$$\mathbf{A}\,\underset{\sim}{v_i} = \lambda_i\,\underset{\sim}{v_i}$$

Si la ecuación anterior se multiplica por la matriz inversa de la matriz **A**, que denominaremos $\mathbf{A}^{-1}$, resulta

$$\mathbf{I}\,\underset{\sim}{v_i} = \lambda_i \mathbf{A}^{-1}\,\underset{\sim}{v_i}$$

o bien

$$\mathbf{A}^{-1}\,\underset{\sim}{v_i} = \left(\frac{1}{\lambda_i}\right)\underset{\sim}{v_i}$$

es decir que $\eta_i = \dfrac{1}{\lambda_i}$ es autovalor dominante de $\mathbf{A}^{-1}$ asociado al mismo autovector $\underset{\sim}{v_i}$.

El Método de la Potencia aplicado sobre la matriz inversa $\mathbf{A}^{-1}$ converge al autovalor dominante de $\mathbf{A}^{-1}$ esto es el mayor η_i tomado en valor absoluto. Pero según la relación entre los η_i y los λ_i, el mayor $|\eta_i|$ está asociado con el menor $|\lambda_i|$ de la matriz **A**.

El autovalor dominante de $\mathbf{A}^{-1}$, es tal que

$$|\eta_1| > |\eta_2| \geq |\eta_3| \geq \ldots \geq |\eta_N|$$

y se cumple la siguiente relación:

$$\eta_1 = \frac{1}{\lambda_N}.$$

El proceso iterativo del Método de la Potencia aplicado sobre $\mathbf{A}^{-1}$, se puede resumir para la iteración k-ésima de la siguiente manera:

dado $\underset{\sim}{y_i}$,

- $\underset{\sim}{x_k} = \underset{\sim}{y_k}\dfrac{1}{\left\|\underset{\sim}{y_k}\right\|}$
- $\underset{\sim}{y_{k+1}} = \mathbf{A}^{-1}\,\underset{\sim}{x_k}$
- $(\alpha_{k+1})_i = \left(\underset{\sim}{y_{k+1}}\right)_i \dfrac{1}{\left(\underset{\sim}{x_k}\right)_i}$

Luego de un número suficiente de iteraciones el cociente entre cualesquiera de las componentes $(\alpha_k)_i$ tiende a η_1 autovalor dominante de $\mathbf{A}^{-1}$, que es igual a $1/\lambda_N$, con λ_N el menor de los autovalores de **A** (tomados en valor absoluto). Los vectores $\underset{\sim}{y}_k$ se alinearán con el autovector $\underset{\sim}{v}$ asociado a dichos autovalores.

Se debe destacar que

$$\underset{\sim}{y}_{k+1} = \mathbf{A}^{-1}\ \underset{\sim}{x}_k$$

exige conocer $\mathbf{A}^{-1}$. Pero, en general lo que se conoce es **A**. Para evitar el cálculo de $\mathbf{A}^{-1}$, el proceso iterativo se modifica

$$\mathbf{A}\cdot\underset{\sim}{y}_{k+1} = \mathbf{A}\cdot\mathbf{A}^{-1}\ \underset{\sim}{x}_k$$

o bien

$$\mathbf{A}\,\underset{\sim}{y}_{k+1} = \underset{\sim}{x}_k$$

que es un sistema de ecuaciones lineales que sólo exige factorizar la matriz **A** una sola vez.

Observación: puede ser preferible en algunos casos realizar una factorización de la matriz **A**, **A** =**LU**, al principio y por única vez. Luego, en **cada paso** del proceso iterativo del método de Potencias inversas se resuelve el sistema de ecuaciones $\mathbf{A}\cdot\underset{\sim}{y}_{k+1} = \underset{\sim}{x}_k$ (donde **A** =**LU** y $\underset{\sim}{x}_k$ son conocidos e $\underset{\sim}{y}_{k+1}$ es el vector de incógnitas) mediante una sustitución progresiva (**L**z = $\underset{\sim}{x}_k$) seguida de una sustitución regresiva (**U** $\underset{\sim}{y}_{k+1}$ =z).

4 CONVERGENCIA A MODOS INTERMEDIOS

Al analizar el proceso de convergencia del Método de las Potencias resulta claro observar que si los vectores $\underset{\sim}{x}_k$ no tienen componente en el autovector $\underset{\sim}{v}_1$ (es decir $a_1 = 0$), la convergencia será al segundo autovector $\underset{\sim}{v}_2$.

Esto se logra mediante una "deflación".
Dado un vector cualquiera $\underset{\sim}{x}_k$, se lo puede escribir como

$$\underset{\sim}{x}_k = a_1 \underset{\sim}{v}_1 + a_2 \underset{\sim}{v}_2 + ... + a_N \underset{\sim}{v}_N$$

ya que los $\underset{\sim}{v}_i$ son una base de $\Re^N$ por ser $\mathbf{A} \in \Re^{NxN}$ diagonizable.

Para obtener a_1 se puede proyectar $\underset{\sim}{x}_k$ en la dirección de $\underset{\sim}{v}_1$ o sea,

$$\left\langle \underset{\sim}{v}_1^T \bullet \underset{\sim}{x}_k \right\rangle = a_1 \left\langle \underset{\sim}{v}_1^T \bullet \underset{\sim}{v}_1 \right\rangle + a_2 \left\langle \underset{\sim}{v}_1^T \bullet \underset{\sim}{v}_2 \right\rangle + ... + a_N \left\langle \underset{\sim}{v}_1^T \bullet \underset{\sim}{v}_N \right\rangle$$

Si se acepta que la matriz **A** *es simétrica, se sabe que sus autovectores constituyen una base de* $\Re^N$ ***ortogonal,*** *es decir*

$$\left\langle \underset{\sim}{v}_1^T \bullet \underset{\sim}{v}_1 \right\rangle = \left\| \underset{\sim}{v}_1 \right\|_2^2$$

$$\left\langle \underset{\sim}{v}_1^T \bullet \underset{\sim}{v}_j \right\rangle = 0 \qquad \forall \quad j \in [2, N]$$

así para **A** simétrica

$$a_1 = \left\langle \underset{\sim}{v}_1^T \bullet \underset{\sim}{x}_k \right\rangle \frac{1}{\left\| \underset{\sim}{v}_1 \right\|_2^2}$$

Al reemplazar en la combinación lineal resulta:

$$\underset{\sim}{x}_k = \underset{\sim}{v}_1 \underset{\sim}{v}_1^T \underset{\sim}{x}_k \frac{1}{\left\| \underset{\sim}{v}_1 \right\|_2^2} + a_2 \underset{\sim}{v}_2 + ... + a_N \underset{\sim}{v}_N$$

o bien

$$\underset{\sim}{x}_k - \underset{\sim}{v}_1 \underset{\sim}{v}_1^T \underset{\sim}{x}_k \frac{1}{\left\| \underset{\sim}{v}_1 \right\|_2^2} = a_2 \underset{\sim}{v}_2 + ... + a_N \underset{\sim}{v}_N$$

$$\left[\mathbf{I} - \underset{\sim}{v_1} \bullet \underset{\sim}{v_1}^T \frac{1}{\left\| \underset{\sim}{v_1} \right\|_2^2} \right] \underset{\sim}{x_k} = a_2 \underset{\sim}{v_2} + \ldots + a_N \underset{\sim}{v_N}$$

Al multiplicar la igualdad anterior por la matriz **A** se tiene:

$$\mathbf{A} \left[\mathbf{I} - \underset{\sim}{v_1} \bullet \underset{\sim}{v_1}^T \frac{1}{\left\| \underset{\sim}{v_1} \right\|_2^2} \right] \underset{\sim}{x_k} = a_2 \ \mathbf{A} \ \underset{\sim}{v_2} + \ldots + a_N \quad \mathbf{A} \underset{\sim}{v_N}$$

$$= \lambda_2 a_2 \underset{\sim}{v_2} + \ldots + \lambda_N a_N \underset{\sim}{v_N}$$

$$= \lambda_2 \left(a_2 \underset{\sim}{v_2} + \ldots + \left(\frac{\lambda_N}{\lambda_2} \right) a_N \underset{\sim}{v_N} \right)$$

En cada iteración se elimina la componente en $\underset{\sim}{v_1}$ si se trabaja con la matriz $\mathbf{B}_1$ dada por,

$$\mathbf{B_1} = \ \mathbf{A} \left[\mathbf{I} - \underset{\sim}{v_1} \underset{\sim}{v_1}^T \frac{1}{\left\| \underset{\sim}{v_1} \right\|_2^2} \right]$$

Que se puede escribir como:

$$\mathbf{B_1} = \left[\mathbf{A} - \mathbf{A} \underset{\sim}{v_1} \underset{\sim}{v_1}^T \frac{1}{\left\| \underset{\sim}{v_1} \right\|_2^2} \right]$$

Que al considerar la definición de valor y vector propio, resulta

$$\mathbf{B_1} = \left[\mathbf{A} - \lambda_1 \underset{\sim}{v_1} \underset{\sim}{v_1}^T \frac{1}{\left\| \underset{\sim}{v_1} \right\|_2^2} \right]$$

Así el proceso iterativo para modos intermedios es:

Dado:

$$\underset{\sim}{y_k} \ \varepsilon \ \Re^N$$

- $\underset{\sim}{x}_k = \underset{\sim}{y}_k \dfrac{1}{\left\| \underset{\sim}{y}_k \right\|}$

- $\underset{\sim}{y}_{k+1} = \quad \mathbf{B_1} \quad \underset{\sim}{x}_k$

Con $\mathbf{B_1} = \mathbf{A}\left[\mathbf{I} - \underset{\sim}{v}_1 \underset{\sim}{v}_1^T \dfrac{1}{\left\| \underset{\sim}{v}_1 \right\|_2^2}\right] = \left[\mathbf{A} - \lambda_1 \underset{\sim}{v}_1 \underset{\sim}{v}_1^T \dfrac{1}{\left\| \underset{\sim}{v}_1 \right\|_2^2}\right]$

- $\left(\alpha_{k+1}\right)_i = \left(\underset{\sim}{y}_{k+1}\right)_i \dfrac{1}{\left(\underset{\sim}{x}_k\right)_i}$

y cuando "k" sea suficientemente grande convergerá $\left(\alpha_{k+1}\right)_i \to \lambda_2$ para cualquier componente i; $\underset{\sim}{y}_k$ se alineará con $\underset{\sim}{v}_2$.

Este resultado obtenido para matrices simétricas, es válido para cualquier matriz diagonizable si se cumple que

$$\mathbf{B} = \mathbf{A} \; - \lambda_1 \underset{\sim}{v}_1 \cdot \underset{\sim}{u}_1^T$$

siendo $\underset{\sim}{u}_1 \in \Re^N$ un vector tal que:

$$\underset{\sim}{v}_1^T \bullet \underset{\sim}{u}_1 = 1$$

entonces se puede probar que **B** tiene como autovalores 0, $\lambda_2, \lambda_3, \ldots, \lambda_N$, siendo $\lambda_2, \lambda_3, \ldots, \lambda_N$ autovalores de **A**.

Esto se conoce como Método de la Deflación.

Según sea la elección del vector $\underset{\sim}{u}_1$ se tienen distintas variantes del Método de la Deflación.

5 EJEMPLO DEL MÉTODO DE LA POTENCIA

Sea la matriz $A = \begin{bmatrix} 10 & -1 \\ -1 & 45 \end{bmatrix}$

Los valores exactos para este ejemplo son:

$$\lambda_{MAYOR} = 45.028548 \qquad v = \begin{Bmatrix} 1 \\ -35.028548 \end{Bmatrix} v_1$$

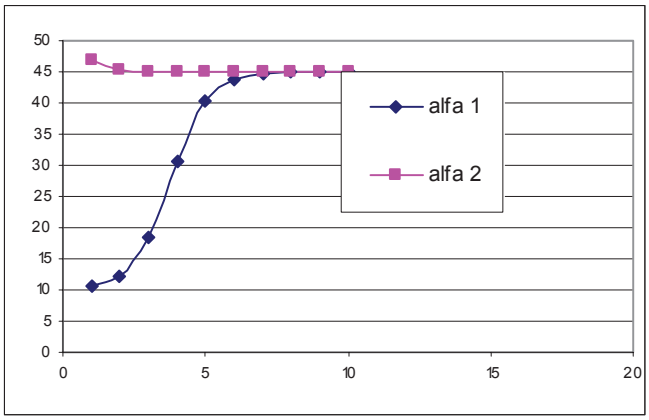
$$\lambda_{MENOR} = 9.971452 \qquad v = \begin{Bmatrix} 1 \\ 35.028548 \end{Bmatrix} v_1$$

Se aplica el método de la potencia para obtener el valor y vector propio a partir del vector inicial Y0={2; -1}

Sin Escalamiento		
Y0	**2**	
	-1	
		alfa=yk+1/yk
Y1	21	10,5
	-47	47
Y2	257	12,23809524
	-2136	45,44680851
Y3	4706	18,31128405
	-96377	45,12031835
Y4	143437	30,47960051
	-4E+06	45,04882908
Y5	6E+06	40,2688358
	-2E+08	45,03303728
Y6	3E+08	43,84993839
	-9E+09	45,02954215
Y6	1E+10	44,76053293
	-4E+11	45,02876826
Y6	5E+11	44,96884151
	-2E+13	45,02859689
Y6	2E+13	45,01530871
	-8E+14	45,02855894
Y6	1E+15	45,02561544
	-4E+16	45,02855053

Con Escalamiento				
		X0	**1**	
			-0,5	
				alfa=yk+1/xk
Y1	10,5	x1	1	10,5
	-23,5		-2,2381	47
Y2	12,2381	x1	1	12,23809524
	-101,714		-8,31128	45,44680851
Y3	18,31128	x1	1	18,31128405
	-375,008		-20,4796	45,12031835
Y4	30,4796	x1	1	30,47960051
	-922,582		-30,2688	45,04882908
Y5	40,26884	x1	1	40,2688358
	-1363,1		-33,8499	45,03303728
Y6	43,84994	x1	1	43,84993839
	-1524,25		-34,7605	45,02954215
Y6	44,76053	x1	1	44,76053293
	-1565,22		-34,9688	45,02876826
Y6	44,96884	x1	1	44,96884151
	-1574,6		-35,0153	45,02859689
Y6	45,01531	x1	1	45,01530871
	-1576,69		-35,0256	45,02855894
Y6	45,02562	x1	1	45,02561544
	-1577,15		-35,0279	45,02855053

La evolución de la aproximación del valor propio dominante es la que se indica en la siguiente gráfica. Se observa que la aproximación de los cocientes de las componentes alfa1 y alfa2 tiene distinta velocidad de convergencia. Si bien alfa2, que es el cociente de las segundas componentes en pocas iteraciones converge a la solución; alfa 1, es más lenta. Es se debe a que si bien el autovalor converge rápidamente, el autovector no es tan rápido.



Esto se observa más claramente al seguir las componentes del vector normalizado x.

Es decir si el objetivo es el valor propio sin importar los autovectores, en pocas iteraciones se obtiene convergencia. Pero si los autovectores también son de interés, el proceso iterativo debe mantenerse hasta alcanzar una alta precisión en los autovalores.

INTERPOLACIÓN Y APROXIMACIÓN POLINOMIAL

1 INTRODUCCIÓN

Sea un conjunto de **(n+1) puntos de coordenadas (x_i; y_i)** con i variando desde 0 hasta n, que se indicará en adelante como: i=0,n. Se asume que estos puntos pertenecen a una cierta función y=f(x) cuya expresión analítica no se conoce. Se dice, entonces, que esta **función está dada en forma discreta**. El principal objetivo de esta unidad temática es encontrar una forma analítica de esta función que sea representable computacionalmente y para ello se usarán funciones polinómicas.
Se tiene y=f(x) de R $\rightarrow$ R definida en forma discreta mediante

(n+1) puntos **(x_i; y_i=f(x_i))** con i=0,n.

Se propone una representación de esta función en forma de combinación lineal de funciones bases $\phi_k(x)$ conocidas y linealmente independientes, es decir

$$y= P_m(x)= \Sigma a_k \phi_k(x),$$

con k=0,m. Para cada punto x_i queda definida una diferencia entre el valor de la función dada en forma discreta **y_i=f(x_i)** y el valor que toma la combinación lineal. Esta diferencia se denomina **residuo** y se calcula para cada x_i, en la forma

$$r_i= f(x_i) - P_m(x_i) = y_i - \Sigma a_k \phi_k(x_i) \qquad k=0,m;\ i=0,n.$$

Se llama **r** al vector de R^{n+1} formado por (r_0, r_1,...,r_n) y se dirá que es una INTERPOLACIÓN cuando se imponga una condición de nulidad "fuerte" del residuo en cada punto, es decir r_i=0, i=0,n, o **r=0**, haciendo que la función $P_m(x)= \Sigma a_k \phi_k(x)$ con k=0,m pase por los puntos datos.

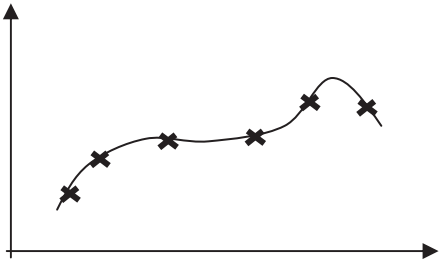
Se dirá que es una APROXIMACIÓN cuando se imponga una condición de nulidad "débil" del residuo en el dominio de interés, es decir que r_i puede no ser cero para algún i, o sea que puede ser **r≠0** . Se verá como forma de aproximación al Método de Mínimos Cuadrados.

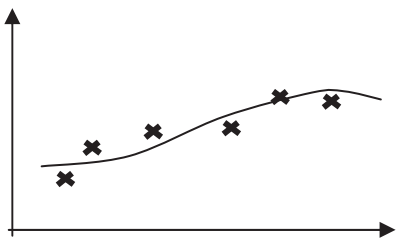
Dada **f(x) en forma discreta**

Se propone $P_m(x)= \Sigma\ a_k \phi_k(x)$ con k=0,m.

Se define $r_i= f(x_i) - P_m(x_i) = y_i - \Sigma\ a_k \phi_k(x_i)$ con k=0,m; i=0,n

INTERPOLACIÓN	APROXIMACIÓN
Forma Fuerte: r_i=0, para todo x_i	Forma Débil: los r_i son nulos en promedio





2 INTERPOLACIÓN

Dada la función y=f(x) de R $\rightarrow$ R definida en forma discreta mediante el conjunto de (n+1) puntos de coordenadas **(x_i; y_i=f(x_i))** con i=0, n, se propone una representación de esta función en forma de combinación lineal de **n funciones bases ϕ_k(x) conocidas**, es decir $y= P_n(x)= \Sigma\, a_k\, \phi_k(x)$,

con k=0,n. Nótese que el número de funciones bases m, en el contexto de interpolación, se toma igual a n (m=n). Para cada punto x_i queda definida una diferencia entre el valor de la función dada en forma discreta **y_i=f(x_i)** y el valor que toma la combinación lineal. Esta diferencia se denomina **residuo** y se calcula para cada x_i, en la forma

$$r_i= f(x_i) – P_n(x_i) = y_i - \Sigma\, a_k\, \phi_k(x_i),$$

con k,i = 0,n. Se impone una condición "fuerte" sobre el residuo exigiendo la nulidad del mismo en cada punto dato de abscisa x_i. Así se exige que,

$$r_i= f(x_i) – P_n(x_i) = y_i - \Sigma\, a_k\, \phi_k(x_i)= 0$$

para todo i,k=0,n. Con lo que resulta

$$\Sigma a_k\, \phi_k(x_i)= y_i \qquad (1.1)$$

para todo i,k=0,n. Esto es un sistema de ecuaciones lineales de la forma

$$\begin{pmatrix} \phi_0(x_0) & \phi_1(x_0) & \phi_2(x_0) & .... & \phi_n(x_0) \\ \phi_0(x_1) & \phi_1(x_1) & \phi_2(x_1) & .... & \phi_n(x_1) \\ \phi_0(x_2) & \phi_1(x_2) & \phi_2(x_2) & .... & \phi_n(x_2) \\ .... & .... & .... & .... & .... \\ \phi_0(x_n) & \phi_1(x_n) & \phi_2(x_n) & .... & \phi_n(x_n) \end{pmatrix} \begin{Bmatrix} a_0 \\ a_1 \\ a_2 \\ ... \\ a_n \end{Bmatrix} = \begin{Bmatrix} y_0 \\ y_1 \\ y_2 \\ ... \\ y_n \end{Bmatrix} \qquad (1.2)$$

de donde es posible encontrar los coeficientes a_i de la combinación lineal propuesta.

OBSERVACIÓN 1:

Es posible demostrar que por n+1 puntos pasa **un único polinomio de grado a lo sumo n**. A pesar de ello, según sean las elecciones de las funciones bases ϕ_k(x) se tiene distintos métodos, como los siguientes: Directo, de Lagrange, de Newton, de Legendre, etc. Pero todos estos métodos conducen a expresiones distintas del **único** polinomio. Por otra parte las funciones bases que utilizan cada uno de estos métodos son linealmente independientes entre sí para generar un subespacio del espacio de todos los posible polinomios.

OBSERVACIÓN 2:

El producto escalar entre dos funciones f y g definidas en el intervalo [a, b] y a valores reales, se puede definir como

$$\langle f, g \rangle = \int_a^b f(x) \cdot g(x)\, dx \cdot$$

Dos funciones se llamarán "ortogonales" cuando el producto escalar entre ellas valga 0.
Llamando r(x) a la función residuo, r(x)=f(x)-P_n(x), cuando se interpola se está exigiendo que la función residuo r(x) sea ortogonal, en el dominio de interés, a las funciones delta de Dirac definidas en cada punto x_i. Esto es

$$\langle \delta, r \rangle \int_a^b \delta(x - x_i) \cdot (f(x) - P_n(x))\, dx = 0 \qquad \forall x_i \cdot$$

2.1 Método Directo

Se toman como funciones bases conocidas $\phi_k(x)$ los polinomios elementales (que generan un subespacio del espacio de funciones de todos los posibles polinomios).

Base = $\{\phi_0(x), \phi_1(x), \phi_2(x), ... ,\phi_n(x)\}=\{1, x ,x^2, x^3,.....,x^n\}$

O bien,

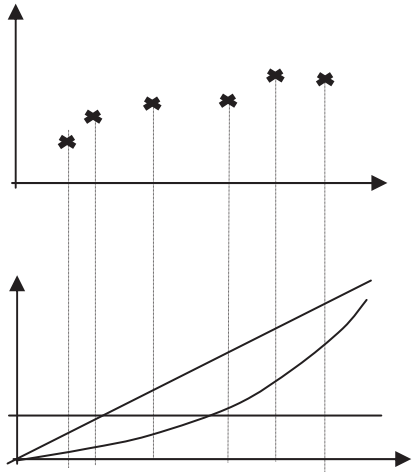
$\phi_0(x)=x^0=1$

$\phi_1(x)=x^1=x$

$\phi_2(x)=x^2$

.....

$\phi_n(x)=x^n$



Al reemplazar estos polinomios bases, evaluados en cada abscisa xi de los puntos datos, en el sistema de ecuaciones (1.2), éste resulta

$$\begin{pmatrix} 1 & x_0 & x_0^2 & .... & x_0^n \\ 1 & x_1 & x_1^2 & .... & x_1^n \\ 1 & x_2 & x_2^2 & .... & x_2^n \\ .... & .... & .... & .... & .... \\ 1 & x_n & x_n^2 & .... & x_n^n \end{pmatrix} \begin{Bmatrix} a_0 \\ a_1 \\ a_2 \\ ... \\ a_n \end{Bmatrix} = \begin{Bmatrix} y_0 \\ y_1 \\ y_2 \\ ... \\ y_n \end{Bmatrix}$$

El determinante de la matriz de coeficientes resultante, se conoce como determinante de Vandermonde. Este determinante será nulo, y por lo tanto el sistema de ecuaciones será singular si y sólo si hay más de un punto con igual valor de abscisa x_i. Es decir, si $x_i=x_k$, para i distinto de k.

Ventaja del método:

- Es de simple interpretación y formulación.

Desventajas del método:

- La matriz de coeficientes puede resultar muy mal condicionada si hay algún par de valores de x_i próximos.
- Para resolver el sistema se debe invertir una matriz de coeficientes donde en general todos los coeficientes son distintos de cero.
- No es posible representar asíntotas verticales que puedan existir en f(x) en el dominio de interés.
- Si se busca agregar un punto a los puntos datos, los polinomios bases cambian todos y se debe calcular todo de nuevo.

Ejercicio1: Dados dos puntos de coordenadas (x0 y0); (x1, y1), obtener el polinomio de interpolación posible usando este método directo.

2.2 Método de polinomios de Lagrange

Se toman como funciones bases conocidas $\phi_k(x)$, los llamados **polinomios de Lagrange**. Estos polinomios tienen como particularidad que, basándose en uno de los puntos datos, *valen uno en la abscisa de ese punto dato de referencia y cero en las abscisas del resto de los puntos datos*. De esta manera el resto de los puntos datos son raíces (ceros) de ese polinomio de Lagrange.
El polinomio de Lagrange $l_0(x)$, deberá valer 1 en x_0 y cero en el resto de las abscisas de los demás puntos datos. Así, es posible escribir este polinomio como producto de binomios de la forma "x menos la raíz". Es decir,

$l_0(x)=C0\ (x-x_1)\ (x-x_2)\ (x-x_3)..... (x-x_n)$.

Para determinar el coeficiente C0 se impone la condición

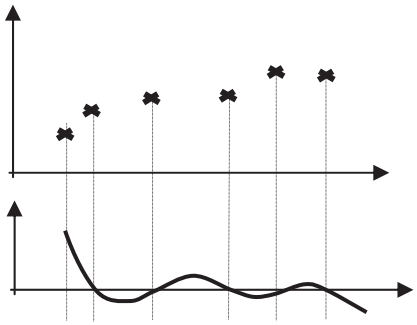
$l_0(x_0)=1$, esto es: $C0\ (x_0-x_1)\ (x_0-x_2)\ (x_0-x_3)..... (x_0-x_n)=1$,

de donde es posible obtener

$C0=1/[(x_0-x_1)\ (x_0-x_2)\ (x_0-x_3)..... (x_0-x_n)]$.

Y así resulta,

$$l_0(x)=\frac{(x-x_1)(x-x_2)(x-x_3).....(x-x_n)}{(x_0-x_1)(x_0-x_2)(x_0-x_3).....(x_0-x_n)}.$$



Análogamente para un punto de abscisa x_i, se tiene

$l_i(x)=Ci\ (x-x_0)(x-x_1)\ (x-x_2)\ (x-x_3).....(x-x_{i-1})(x-x_{i+1}).....\ (x-x_n)$,

$l_i(x_i)=1$, esto es: $Ci\ (x_i-x_0)(x_i-x_1)\ (x_i-x_2)\ (x_i-x_3).....(x_i-x_{i-1})(x_i-x_{i+1}).....\ (x_i-x_n)=1$,

de donde es posible obtener

$Ci=1/[\ (x_i-x_0)(x_i-x_1)\ (x_i-x_2)\ (x_i-x_3).....(x_i-x_{i-1})(x_i-x_{i+1}).....\ (x_i-x_n)]$.

Y así resulta,

$$l_i(x)=\frac{(x-x_1)(x-x_2)(x-x_3).....(x-x_n)}{(x_i-x_1)(x_i-x_2)(x_i-x_3).....(x_i-x_n)},$$

$$l_i(x)=\prod_{\substack{k=0\\k\neq i}}^{n}\frac{(x-x_k)}{(x_i-x_k)}.$$

Los polinomios de Lagrange generan un subespacio del espacio de funciones de todos los posibles polinomios. Esto es:

Base=$\{\phi_0(x), \phi_1(x), \phi_2(x),..... ,\phi_n(x)\} =\{l_0(x), l_1(x), l_2(x),..... , l_n(x)\}$.

Al reemplazar estos polinomios bases, evaluados en cada abscisa x_i de los puntos datos, en el sistema de ecuaciones (1.2), éste resulta

$$\begin{pmatrix} 1 & 0 & 0 & .... & 0 \\ 0 & 1 & 0 & .... & 0 \\ 0 & 0 & 1 & .... & 0 \\ .... & .... & .... & .... & .... \\ 0 & 0 & 0 & .... & 1 \end{pmatrix} \begin{Bmatrix} a_0 \\ a_1 \\ a_2 \\ ... \\ a_n \end{Bmatrix} = \begin{Bmatrix} y_0 \\ y_1 \\ y_2 \\ ... \\ y_n \end{Bmatrix},$$

de donde se desprende que, tomando como funciones bases los polinomios de Lagrange, los coeficientes de la combinación lineal a_i son directamente los valores datos y_i. Resulta así que la interpolación con polinomios de Lagrange es

$$P_n(x) = \Sigma[y_k \; l_k(x)] \qquad \text{con } k=0,n.$$

Ventajas del método:

- Es de simple interpretación.
- Es simple su implementación computacional.
- No se debe invertir ninguna matriz.
- La matriz de coeficientes resulta siempre diagonal.

Desventajas del método:

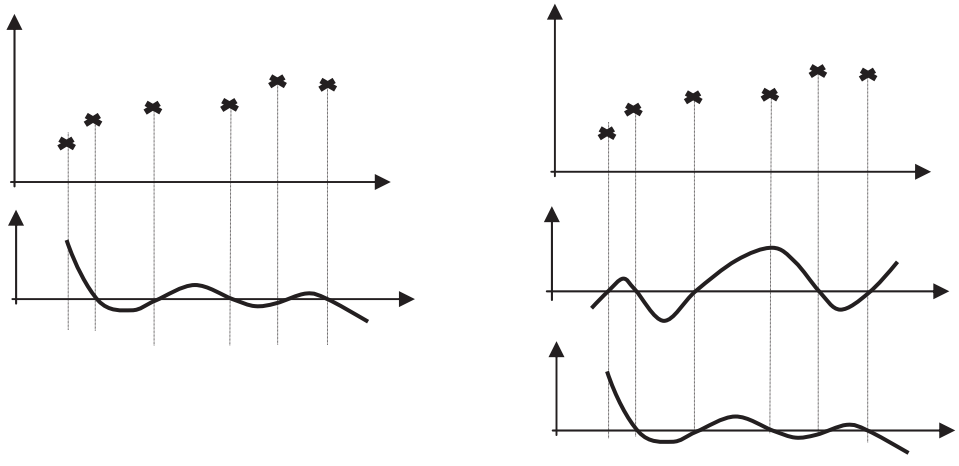
- Si se busca agregar un punto a los puntos datos, los polinomios bases cambian todos y se debe calcular todo de nuevo.
- Puede ser difícil obtener una expresión simplificada del polinomio obtenido.
- No es posible representar asíntotas verticales que puedan existir en f(x) en el dominio de interés.

Ejercicio 1:

Dados dos puntos de coordenadas (x0, y0); (x1, y1), obtener la interpolación posible usando el método de Lagrange.

Ejercicio2:

Dados tres puntos de coordenadas (x0, y0); (x1, y1) (x2, y2), obtener la interpolación posible usando el método de Lagrange. Notar si es posible usar expresiones, polinomios bases o lo que se pueda del ejercicio anterior.



2.3 Método de polinomios de Newton

Se toman como funciones bases conocidas $\phi_k(x)$, los llamados **polinomios de Newton**. Estos polinomios tienen como particularidad que se basan en los polinomios bases anteriores.

$\phi_0(x) = n_0(x) = 1$

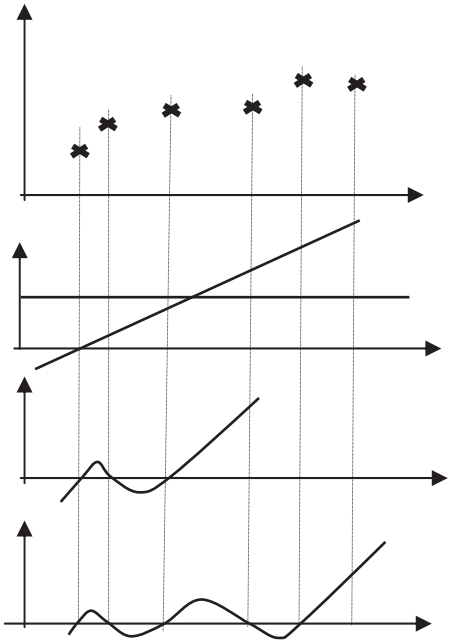
$\phi_1(x) = n_1(x) = 1(x-x_0)$

$\phi_2(x) = n_2(x) = 1(x-x_0)(x-x_1)$

$\phi_3(x) = n_3(x) = 1(x-x_0)(x-x_1)(x-x_2)$

.....

$\phi_n(x) = n_n(x) = 1(x-x_0)(x-x_1)(x-x_2).....(x-x_{n-1})$.



En general se pueden expresar como

$\phi_0(x) = n_0(x) = 1$,

$\phi_k(x) = n_k(x) = n_{k-1}(x)\cdot(x-x_{k-1})$, para todo $k \geq 1$.

Al reemplazar estos polinomios bases de Newton, evaluados en cada abscisa x_i de los puntos datos en la matriz de coeficientes del sistema de ecuaciones (1.2), resulta

$$\begin{pmatrix} 1 & 0 & 0 & ..... & 0 \\ 1 & x_1 - x_0 & 0 & ..... & 0 \\ 1 & x_2 - x_0 & (x_2 - x_0)(x_2 - x_1) & ..... & 0 \\ .. & ..... & ..... & ..... & 0 \\ 1 & x_n - x_0 & (x_n - x_0)(x_n - x_1) & ..... & (x_n - x_0)(x_n - x_1).....(x_n - x_{n-1}) \end{pmatrix},$$

de donde es posible obtener los coeficientes de la combinación lineal a_k, por sustitución hacia adelante. Así resulta:

$a_0 = y_0$,

$$a_1 = \frac{y_1 - y_0}{x_1 - x_0},$$

$$a_2 = \frac{y_2 - a_1(x_2 - x_1) - y_0}{(x_2 - x_0)(x_2 - x_1)}$$

que, si se suma y resta y_1 en el numerador, se puede expresar como:

$$a_2 = \frac{\dfrac{y_2 - y_1}{x_2 - x_1} - \dfrac{y_1 - y_0}{x_1 - x_0}}{x_2 - x_0}$$

A los valores de a_1 y a_2 se los conoce como diferencias divididas de Newton de orden 1 y 2 respectivamente. En general los coeficientes de la combinación lineal, usando como

base polinomios de Newton, son las Diferencias Divididas de Newton.

Ejercicio resuelto:

Dados tres puntos de coordenadas (1, 2); (2, -3) y (5, 6), obtener la interpolación posible usando el método de Newton.

x	f(x)	diferencia dividida de orden 1	diferencia dividida de orden 2
1	2 (a_0)	$\frac{-3-2}{2-1}=\frac{-5}{1}=-5$ (a_1)	$\frac{3-(-5)}{5-1}=\frac{8}{4}=2$ (a_2)
2	-3	$\frac{6-(-3)}{5-2}=\frac{9}{3}=3$	
5	6		

Como

$$P_2(x)= a_0 + a_1 (x - x_0) + a_2 (x - x_0) (x - x_1),$$

se obtiene

$$P_2(x)= 2 + (-5) (x - 1) + 2 (x - 1) (x - 2).$$

Ventajas del método:

- Es de simple interpretación.
- Es simple su implementación computacional.
- No se debe invertir ninguna matriz.
- Si se agrega un punto a los puntos datos, los polinomios bases no cambian y es fácil calcular el nuevo coeficiente; no es necesario calcular todo de nuevo.

Desventajas del método:

- Puede ser difícil obtener una expresión simplificada del polinomio obtenido.
- No es posible representar asíntotas verticales que puedan existir en f(x) en el dominio de interés.

2.4 Error de Interpolación

Un tratamiento detallado se puede ver en:

- *Análisis Numérico*, R. Burden y D. Faires (1998). Capitulo 3, Teorema 3.3.
- *Análisis Numérico*, W. Smith (1988). Capitulo 7, Teorema 2.

Dados (n+1) puntos de coordenadas **(x_i; y_i=f(x_i))** con i=0, n, el polinomio de interpolación con **n funciones bases $\phi_k(x)$ conocidas** (Lagrange, Newton, etc), se expresa en la forma:

$$y= P_n(x)= \Sigma\ a_k\ \phi_k(x),$$

El *Error de interpolación* es la diferencia entre la función f(x) y el polinomio de interpolación, y se puede expresar en la forma:

$$E(x) = f(x) - P_n(x)$$

Esta función ***Error de interpolación*** tiene (n+1) ceros, ya que en cada uno de los x_i datos, los valores de $f(x_i)$ coinciden con los de $P_n(x_i)$, por lo que **se puede expresar como un polinomio de al menos grado (n+1)**, como

$$E\ (x)= C\ (x-x_0)\ (x-x_1)\ (x-x_2)\ldots\ldots\ (x-x_n)$$

La constante C se determinará de modo que la función auxiliar W(x) sea cero para todo valor de x. Siendo,

$$W(x)= f(x) - P_n(x)-E(x)$$

Es decir, la constante C se determinará de modo que la igualdad $f(x)=P_n(x)+E(x)$, se cumpla para cualquier valor de x.

La función *W* (x) vale cero en cada x_k dato (k=0,n), es decir tiene n+1 ceros. Pero para cualquier otro $x_i \neq x_k$, se elige C de modo que *W* (x_i)=0, entonces

W(x) tiene (n+2) ceros, para cada x_i;

$\frac{d^1W}{dx^1}$ tiene (n+2-1) ceros, para cada x_i;

$\frac{d^2W}{dx^2}$ tiene (n+2-2) ceros, para cada x_i;

$\frac{d^3W}{dx^3}$ tiene (n+2-3) ceros, para cada x_i;

……

$\frac{d^nW}{dx^n}$ tiene (n+2-n) ceros, para cada x_i,

$\frac{d^{n+1}W}{dx^{n+1}}$ tiene (n+2-n-1) cero, para cada x_i,

Así la derivada de *W*(x) de orden (n+1) tiene un cero. Esta derivada es:

$$\frac{d^{n+1}W}{dx^{n+1}} = \frac{d^{n+1}f}{dx^{n+1}} - \frac{d^{n+1}P_n}{dx^{n+1}} - \frac{d^{n+1}E}{dx^{n+1}}$$

$$\frac{d^{n+1}W}{dx^{n+1}} = \frac{d^{n+1}f}{dx^{n+1}} - 0 - C\cdot(n+1)!$$

De donde es posible obtener C que anula $\frac{d^{n+1}W}{dx^{n+1}}$ para cada valor de x, es decir

$C = \frac{\frac{d^{n+1}f(\xi)}{dx^{n+1}}}{(n+1)!}$ con ξ ε (x_0, x_n), que da un *valor medio* de la derivada n-ésima de f(x). Así para cada valor de x existe una C que hace el error de interpolación se pueda expresar

$$E(x) = \frac{\frac{d^{n+1}f(\xi)}{dx^{n+1}}}{(n+1)!}(x-x_0)(x-x_1)(x-x_2).....(x-x_n)$$

Esta expresión establece que:

- El error de interpolación en las abscisas datos es cero
- La interpolación es exacta si f(x) es un polinomio hasta de grado n.

2.5 Método de polinomios de Hermite

Uno de los inconvenientes de interpolar con polinomios de Lagrange o de Newton de grado superior al 4 son las oscilaciones que tienen los polinomios resultantes entre los puntos datos. Para evitar estas oscilaciones se recurre a interpolar valores de la función y=f(x) y también de su derivada primera y'=f'(ξ) que se deberán tener como datos.

De esa manera es posible imponer los siguientes dos conjuntos de condiciones:

$P(x_i) = y_i$ con i=0,n,

$P'(x_i) = y'_i$ con i=0,n,

que representan 2n+2 condiciones Estas permiten determinar 2n+2 coeficientes y así el grado del polinomio resultante será 2n+1.

Los polinomios de Hermite resultan de interpolar el valor de la función y su derivada primera entre dos puntos. Es decir se conoce en dos puntos de abscisas x_i y x_{i+1} los valores de la función y_i e y_{i+1}, y los valores de sus derivadas primeras y'_i e y'_{i+1}.

(x_i, y_i); (x_{i+1}, y_{i+1})

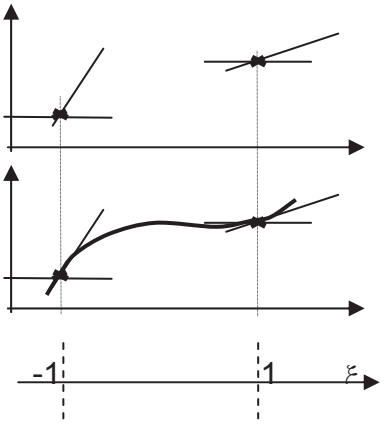
(x_i, y'_i); (x_{i+1}, y'_{i+1})

Se tienen cuatro condiciones a cumplir, de modo que es posible plantear un polinomio de orden cúbico,

$P(x) = a_0 + x\,a_1 + x^2\,a_2 + x^3\,a_3$

$P'(x) = a_1 + 2\,x\,a_2 + 3\,x^2\,a_3$

Al imponer las cuatro condiciones conocidas en los puntos, resulta el siguiente sistema de ecuaciones lineales,



$$\begin{bmatrix} 1 & x_i & x_i^2 & x_i^3 \\ 1 & x_{i+1} & x_{i+1}^2 & x_{i+1}^3 \\ 0 & 1 & 2x_i & 3x_i^2 \\ 0 & 1 & 2x_{i+1} & 3x_{i+1}^2 \end{bmatrix} \begin{Bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{Bmatrix} = \begin{Bmatrix} y_i \\ y_{i+1} \\ y_i' \\ y_{i+1}' \end{Bmatrix}$$

cuya solución da los coeficientes de la combinación lineal.

Es posible deducir estos polinomios en un domino "estándar o unitario" planteado el siguiente cambio de variables, o mapeo del eje x en el eje ξ,

$$\frac{x - x_i}{x_{i+1} - x_i} = \frac{\xi + 1}{2}$$ o bien

$x(\xi) = x_i + \frac{x_{i+1} - x_i}{2}(\xi + 1)$,y la relación inversa $\xi(x) = \frac{2}{x_{i+1} - x_i}(x - x_i) - 1$

Así el polinomio se puede expresar

$$P(\xi) = \begin{bmatrix} 1 & \xi & \xi^2 & \xi^3 \end{bmatrix} \begin{Bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{Bmatrix}$$ que matricialmente es $P(\xi)$=**Ha**

De imponer que se verifiquen: (-1, y_i); (-1, y'_i); (1, y_{i+1}); (1, y'_{i+1}); se obtiene el siguiente sistema

$$\begin{bmatrix} 1 & -1 & 1 & -1 \\ 0 & 1 & -2 & 3 \\ 1 & 1 & 1 & 1 \\ 0 & 1 & 2 & 3 \end{bmatrix} \begin{Bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{Bmatrix} = \begin{Bmatrix} y_i \\ y_i' \\ y_{i+1} \\ y_{i+1}' \end{Bmatrix}$$ que matricialmente es **Ca=y** con **y=** $\begin{Bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{Bmatrix} = \begin{Bmatrix} y_i \\ y_i' \\ y_{i+1} \\ y_{i+1}' \end{Bmatrix}$

Su solución es $\mathbf{a}=\mathbf{C}^{-1}\mathbf{y}$, siendo

$$\mathbf{C}^{-1} = \frac{1}{4}\begin{bmatrix} 2 & 1 & 2 & -1 \\ -3 & -1 & 3 & -1 \\ 0 & -1 & 0 & 1 \\ 1 & 1 & -1 & 1 \end{bmatrix}$$

Así resulta $P(\xi)$=$\mathbf{H}*\mathbf{C}^{-1}\mathbf{y}$, o bien $P(\xi)$=$\mathbf{N}*\mathbf{y}$, con $\mathbf{N}= \mathbf{H}*\mathbf{C}^{-1}$

$$\mathbf{N}=\begin{bmatrix}1 & \xi & \xi^2 & \xi^3\end{bmatrix}\frac{1}{4}\begin{bmatrix}2 & 1 & 2 & -1\\ -3 & -1 & 3 & -1\\ 0 & -1 & 0 & 1\\ 1 & 1 & -1 & 1\end{bmatrix}=\begin{bmatrix}N_1 & N_2 & N_3 & N_4\end{bmatrix}$$

$N_1(\xi)$= *(1/4)(2 - 3ξ + ξ³)* = *(1/4)(2+ ξ)(1– ξ)²*

$N_2(\xi)$= *(1/4)(1 - ξ – ξ² + ξ³)* = *(1/4)(1+ ξ)(1– ξ)²*

$N_3(\xi)$= *(1/4)(2 + 3ξ – ξ³)* = *(1/4)(2- ξ)(1+ ξ)²*

$N_4(\xi)$= *(1/4)(-1 - 1ξ + ξ² + ξ³)* = *(1/4)(ξ–1)(1+ ξ)²*

Con las relaciones de *x(ξ), ξ(x), $N_i(\xi)$* (i=1, 4), es posible calcular cualquier valor intermedio, y también obtener cualquier derivada. Para ello se debe considerar que

$$\frac{dP}{dx}=\frac{dP}{d\xi}\cdot\frac{d\xi}{dx}=\frac{2}{x_{i+1}-x_i}\frac{dP}{d\xi}=\left(\frac{2}{x_{i+1}-x_i}\right)\sum_{k=1}^{4}\frac{dN_k(\xi)}{d\xi}\cdot y_k$$

En forma matricial se puede expresar

$$\frac{dP}{dx}=\left(\frac{2}{x_{i+1}-x_i}\right)\begin{bmatrix}\frac{dN_1(\xi)}{d\xi} & \frac{dN_2(\xi)}{d\xi} & \frac{dN_3(\xi)}{d\xi} & \frac{dN_4(\xi)}{d\xi}\end{bmatrix}\begin{Bmatrix}y_1\\ y_2\\ y_3\\ y_4\end{Bmatrix}=(2/(x_{i+1}-x_i)*\mathbf{B}*\mathbf{y}$$

3 INTERPOLACIÓN CON SPLINES CÚBICOS

Dados n+1 puntos o nodos t_0, t_1, ... , t_n, de componentes (x_i, y_i) i=0, 1, ..., n, se busca una función spline S(x) de grado k que satisface los siguientes requisitos:

- S es un polinomio de grado menor o igual que k en cada subintervalo [x_i, x_{i+1}].
- S tiene derivada continua de orden k-1 en todo el intervalo [x_0, x_n].
- S es un polinomio continuo en [x_0, x_n] de grado menor o igual a k, con derivadas continuas de orden k-1, pero definido por tramos.

Los splines más usados son los cúbicos que tienen las siguientes características:

- En cada subintervalo [x_i, x_{i+1}] el polinomio $S_i(x)$ es cúbico por consiguiente tiene 4 coeficientes a determinar.
- Existe continuidad de la función S(x), su derivada primera S'(x), y su derivada segunda S"(x), en todos los puntos del intervalo [x_0, x_n].

Se debe determinar 4 coeficientes de cada polinomio cúbico, es decir, 4n coeficientes.

Se tiene como condiciones

- en cada intervalo [x_i; x_{i+1}] el polinomio debe tomar el valor dato, esto es:

$$\left.\begin{aligned}S_i(x_i)&=y_i\\ S_i(x_{i+1})&=y_{i+1}\end{aligned}\right\}$$ en n subintervalos, o sea son 2n condiciones

- continuidad de S' en todos los t_i, es decir $S'_{i-1}(x_i) = S'_i(x_i)$ en todos los puntos, salvo los extremos t_0, t_n. Es decir son n –1 condiciones.
- Continuidad de S" en todos los t_i $S''_{i-1}(x_i) = S''_i(x_i)$ son n –1 condiciones.

Se tienen entonces 2n + n –1 + n –1 = 4n – 2 condiciones y 4n coeficientes a determinar. La elección de las dos condiciones adicionales se usa según convenga.

Para determinar los coeficientes de los polinomios en cada intervalo, se deben relacionar las condiciones de continuidad a cumplir con los datos que se tienen, que son las coordenadas (x_i, y_i) de los n puntos.

Por cada par de valores (puntos) (x_i, y_i) (x_{i+1}, y_{i+1}) se debe encontrar los coeficientes del polinomio cúbico cumpliendo con los requisitos de continuidad.

Se tienen como datos

(x_i, y_i) (x_{i+1}, y_{i+1}) $\quad h_i=x_{i+1}-x_i$.

Se define $S''(x_i)=z_i$ $\quad S''(x_{i+1})=z_{i+1}$.

Los valores de curvatura z_i no se conocen y deben determinarse a partir de los requisitos de continuidad y de los puntos (x_i, y_i). Para ello se considera que si la función de interpolación que se busca es de grado 3, su curvatura (derivada segunda) tiene una variación lineal en x y se la puede escribir como

$$S''_i(x) = \frac{z_i}{h_i}(x_{i+1} - x) + \frac{z_{i+1}}{h_i}(x - x_i)$$

que es una interpolación de la curvatura usando polinomios de Newton de orden 1.

Si se integra $S''_i(x)$ respecto de x, resulta

$$S'_i(x) = -\frac{z_i}{z_{h_i}}(x_{i+1} - x)^2 + \frac{z_{i+1}}{2h_i}(x - x_i)^2 + C$$

e integrando una vez más, $S_i(x) = \frac{z_i}{6h_i}(x_{i+1} - x)^3 + \frac{z_{i+1}}{6h_i}(x - x_i)^3 + Cx + D$

Las constantes de integración C y D se obtienen al imponer (colocar) que el polinomio $S_i(x)$ pase por los puntos datos. Es decir,

$$\begin{cases} S_i(x_i) = y_i = z_i \dfrac{h_i^2}{6} + Cx_i + D \\ S_i(x_{i+1}) = y_{i+1} = z_{i+1} \dfrac{h_i^2}{6} + Cx_{i+1} + D \end{cases},$$

que es un sistema lineal en C y D

$$\begin{bmatrix} 1 & x_i \\ 1 & x_{i+1} \end{bmatrix} \begin{Bmatrix} D \\ C \end{Bmatrix} = \begin{Bmatrix} y_i - z_i \dfrac{h_i^{\,2}}{6} \\ y_{i+1} - z_{i+1} \dfrac{h_i^{\,2}}{6} \end{Bmatrix} = h_i \begin{Bmatrix} a \\ b \end{Bmatrix} \text{ siendo } \begin{cases} a = \dfrac{y_i}{h_i} - z_i \dfrac{h_i}{6} \\ b = \dfrac{y_{i+1}}{h_i} - z_{i+1} \dfrac{h_i}{6} \end{cases}.$$

Para resolver el sistema de 2 x 2 se puede hacer

$$\left(\begin{array}{cc|c} 1 & x_i & h_i a \\ 1 & x_{i+1} & h_i b \end{array}\right) \rightarrow \left(\begin{array}{cc|c} 1 & x_i & h_i a \\ 0 & x_{i+1} - x_i & h_i b - h_i a \end{array}\right), \text{ de donde}$$

$$C = \frac{h_i (b-a)}{x_{i+1} - x_i} = b - a \quad .$$

$$D = h_i a - C x_i = h_i a - (b-a) x_i$$

Así el término lineal Cx+D de $S_i(x)$ resulta,

$$\begin{aligned} Cx+D &= (b-a)\, x + h_i\, a - x_i (b-a) \\ &= bx - ax + h_i\, a - x_i\, b + x_i\, a \\ &= a\,(-x + h_i + x_i) + b\,(x - x_i) \\ &= a\,(x_{i+1} - x) + b\,(x - x_i) \end{aligned}$$

y de esta manera $S_i(x)$ resulta

$$S_i(x) = \frac{z_i}{6h_i}(x_{i+1} - x)^3 + \frac{z_{i+1}}{6h_i}(x - x_i)^3 + \left(\frac{y_i}{h_i} - \frac{z_i h_i}{6}\right)(x_{i+1} - x) + \left(\frac{y_{i+1}}{h_i} - \frac{z_{i+1} h_i}{6}\right)(x - x_i)$$

y $S'_i(x) = -\dfrac{z_i}{2h_i}(x_{i+1} - x)^2 + \dfrac{z_{i+1}}{2h_i}(x - x_i)^2 + \left(\dfrac{y_{i+1}}{h_i} - \dfrac{z_{i+1} h_i}{6}\right) - \left(\dfrac{y_i}{h_i} - \dfrac{z_i h_i}{6}\right).$

Para asegurar continuidad en S'(x) se debe imponer $S'_i(x_i) = S'_{i-1}(x_i)$.

De evaluar $S'_i(x)$ en $x = x_i$ se obtiene

$$S'_i(x_i) = -z_i \frac{h_i}{2} + \frac{y_{i+1}}{h_i} - z_{i+1} \frac{h_i}{6} - \frac{y_i}{h_i} + z_i \frac{h_i}{6}$$

$$S'_i(x_i) = -z_i \frac{h_i}{3} - z_{i+1} \frac{h_i}{6} + \frac{y_{i+1}}{h_i} - \frac{y_i}{h_i} \quad .$$

Si se considera la expresión de $S'_i(x)$, se puede deducir que

$$S'_{i-1}(x) = -\frac{z_{i-1}}{2h_{i-1}}(x_i - x)^2 + \frac{z_i}{2h_{i-1}}(x - x_{i-1})^2 + \left(\frac{y_i}{h_{i-1}} - z_i\frac{h_{i-1}}{6}\right) - \left(\frac{y_{i-1}}{h_{i-1}} - z_{i-1}\frac{h_{i-1}}{6}\right)$$

$$S'_{i-1}(x) = z_i\frac{h_{i-1}}{2} + \frac{y_i}{h_{i-1}} - z_1\frac{h_{i-1}}{6} - \frac{y_{i-1}}{h_{i-1}} + z_{i-1}\frac{h_{i-1}}{6}$$

$$S'_{i-1}(x) = z_i\frac{h_{i-1}}{3} + z_{i-1}\frac{h_{i-1}}{6} + \frac{y_i}{h_{i-1}} - \frac{y_{i-1}}{h_{i-1}}$$

De igualar $S'_i(x_i) = S'_{i-1}(x_i)$ se obtiene

$$-z_i\frac{h_i}{3} - z_{i+1}\frac{h_i}{6} + \frac{y_{i+1}}{h_i} - \frac{y_i}{h_i} = z_i\frac{h_{i-1}}{3} + z_{i-1}\frac{h_{i-1}}{6} + \frac{y_i}{h_{i-1}} - \frac{y_{i-1}}{h_{i-1}}$$

$$z_{i-1}\frac{h_{i-1}}{6} + z_i\left(\frac{h_{i-1}}{3} + \frac{h_i}{3}\right) + z_{i+1}\frac{h_i}{6} = \frac{y_{i+1}}{h_i} - \frac{y_i}{h_i} - \frac{y_i}{h_{i-1}} + \frac{y_{i-1}}{h_{i-1}}$$

$$z_{i-1}h_{i-1} + z_i 2(h_{i-1} + h_i) + z_{i+1}h_i = \frac{6}{h_i}(y_{i+1} - y_i) - \frac{6}{h_{i-1}}(y_i - y_{i-1})$$

siendo las incógnitas de esta ecuación los valores de las curvaturas z_{i-1}, z_i, z_{i+1}.

Esta condición de continuidad de S'(x) se puede plantear para i = 1 hasta i = n –1 dando un sistema de (n – 1) ecuaciones con (n+1) incógnitas:

$$\begin{bmatrix} h_0 & u_1 & h_1 & & & & & & \\ & h_1 & u_2 & h_2 & & & & & \\ & & h_2 & u_3 & h_3 & & & & \\ & & & & \ddots & & & & \\ & & & & & \ddots & & & \\ & & & & & h_{n-3} & u_{n-2} & h_{n-2} & \\ & & & & & & h_{n-2} & u_{n-1} & h_{n-1} \end{bmatrix} \begin{Bmatrix} z_0 \\ z_1 \\ z_2 \\ \vdots \\ z_{n-2} \\ z_{n-1} \\ z_n \end{Bmatrix} = \begin{Bmatrix} \gamma_1 \\ \gamma_2 \\ \vdots \\ \vdots \\ \vdots \\ \gamma_{n-2} \\ \gamma_{n-1} \end{Bmatrix},$$

siendo

$$h_i = x_{i+1} - x_i$$
$$u_i = 2(h_i + h_{i-1})$$
$$\gamma_i = \frac{6}{h_i}(y_{i+1} - y_i) - \frac{6}{h_{i-1}}(y_i - y_{i-1})$$

Para resolver el sistema de (n-1) ecuaciones con (n+1) incógnitas, se eligen dos de las incógnitas. La elección más conveniente es la que origina los **SPLINES CUBICOS NATURALES** y es $z_0 = z_n = 0$, y el sistema tridiagonal resulta de (n-1)·(n-1).

Así, dados n +1 puntos (x_i, y_i) la solución del sistema de ecuaciones da los valores de curvaturas de los splines cúbicos naturales definidos por tramos mediante la expresión $S_i(x)$, que aseguran continuidad hasta segundo orden.

4 MÉTODO DE MÍNIMOS CUADRADOS

Dados n puntos de coordenadas (x_i, y_i), se asume que pertenecen a una función y(x) que no se conoce. Se busca **APROXIMAR** dicha función mediante una combinación lineal de m funciones bases conocidas. Así,

$$f_m(x) = \Sigma a_j \varphi_j(x) \qquad j=1,m,$$

siendo $\phi_j(x)$ funciones bases conocidas que son linealmente independientes y a_j coeficientes a determinar.

Observación 1: a diferencia de lo que se hace para interpolar, acá se considera un número **m < n** de funciones bases. En caso de tomar m=n, si no se tiene dos puntos datos con igual abscisa, este método generará el mismo polinomio que los métodos de interpolación.

Observación 2: en general, al aproximar una función por mínimos cuadrados, no hay problema cuando dos puntos datos presentan la misma abscisa; el problema aparece al interpolar (o cuando se aproxima por mínimos cuadrados y se toma m=n).

Cuando se evalúa la aproximación $f_m(x)$ en las coordenadas x_i, aparece un RESIDUO

$$r_i = y_i - \Sigma a_j \varphi_j(x_i) \qquad i=1,n$$

como diferencia entre el valor conocido y_i, y el valor de la función aproximada en x_i. Llamando **r** al vector n dimensional de componentes r_i, i=1,n, se tiene que los coeficientes a_j son tales que

$$\min \left\| \underline{\mathbf{r}} \right\|_2^2 = \min\left(\Sigma(r_i\, r_i)\right),$$

es decir, minimizan la Suma de los Cuadrados de los Residuos.

La condición para que exista el mínimo es que

$$\frac{\partial \left\| r \right\|_2^2}{\partial a_j} = \frac{\partial}{\partial a_j}\left(\sum_i (r_i\, r_i) \right) = 0 \qquad i = 1,n;\ j = 1,m.$$

$$\frac{\partial}{\partial a_j}\left[\sum_i \left(r_i^{\,2}\right)\right] = 0$$

$$\sum_i \frac{\partial}{\partial a_j}\left(r_i^{\,2}\right) = 0$$

$$\sum_i 2\, r_i \frac{\partial}{\partial a_j}(r_i) = 0$$

$$2\sum_i r_i \frac{\partial}{\partial a_j}\left(y_i - \sum_k \varphi_k(x_i)\, a_k\right) = 0 \qquad \begin{matrix} i = 1,n \\ j = 1,m \\ k = 1,m \end{matrix}$$

$$2\sum_i r_i \left(0 - \varphi_j(x_i)\right) = 0$$

$$\sum_i r_i \; \varphi_j(x_i) = 0$$

$$(2) \qquad \sum_i \left(y_i - \sum_k \varphi_k(x_i)\, a_k\right)\varphi_j(x_i) = 0$$

$$\sum_i \sum_k \varphi_j(x_i)\, \varphi_k(x_i)\, a_k = \sum_i \varphi_j(x_i)\, y_i$$

Los coeficientes a_j se obtienen de imponer la Condición de Normalidad (ortogonalidad) (2) entre los residuos r_i y las funciones bases $\varphi_j(x_i)$ como vectores. Llamando $\underline{a}$, $\underline{y}$, y $\underline{\varphi_j}$ a los siguientes vectores y Φ a la siguiente matriz,

$$\underline{a} = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_m \end{bmatrix},\ \underline{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix},\ \underline{\varphi_j} = \begin{bmatrix} \varphi_j(x_1) \\ \varphi_j(x_2) \\ \vdots \\ \varphi_j(x_n) \end{bmatrix},\ \Phi = \begin{bmatrix} \varphi_1(x_1) & \varphi_2(x_1) & \cdots & \varphi_m(x_1) \\ \varphi_1(x_2) & \varphi_2(x_2) & \cdots & \varphi_m(x_2) \\ \vdots & \vdots & \ddots & \vdots \\ \varphi_1(x_n) & \varphi_2(x_n) & \cdots & \varphi_m(x_n) \end{bmatrix}$$

se observa que

$$\underline{r} = \begin{bmatrix} y_1 - \sum_k a_k \varphi_k(x_1) \\ y_2 - \sum_k a_k \varphi_k(x_2) \\ \vdots \\ y_n - \sum_k a_k \varphi_k(x_n) \end{bmatrix} = \underline{y} - \Phi\, \underline{a}$$

y la expresión **(2)** se puede expresar en forma vectorial:

$$\sum_i r_i \, \varphi_j(x_i) = 0 \qquad j = 1, m$$

$$\underline{\varphi}_j^T \underline{r} = 0 \qquad j = 1, m$$

$$\Phi^T \underline{r} = \underline{0}$$

$$\Phi^T \left(\underline{y} - \Phi \underline{a} \right) = \underline{0}$$

Para finalmente obtener

$$\Phi^T \Phi \underline{a} = \Phi^T \underline{y},$$

que es el sistema de ecuaciones cuya resolución da los coeficientes a_j buscados.

Casos especiales:

Aproximación Lineal

Datos (x_i, y_i) i=1, n
Base $\{\varphi_1(x), \varphi_2(x)\}=\{1, x\}$

$$\Phi = \begin{bmatrix} \varphi_1(x) & \varphi_2(x) \end{bmatrix} = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_n \end{bmatrix}$$

$$\Phi^T \Phi \, \underline{a} = \Phi^T \underline{y}$$

$$\begin{bmatrix} 1 & 1 & \cdots & 1 \\ x_1 & x_2 & \cdots & x_n \end{bmatrix} \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_n \end{bmatrix} \begin{Bmatrix} a_1 \\ a_2 \end{Bmatrix} = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ x_1 & x_2 & \cdots & x_n \end{bmatrix} \begin{Bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{Bmatrix}$$

$$\begin{bmatrix} n & \sum_{I=1}^{n} x_i \\ \sum_{I=1}^{n} x_i & \sum_{I=1}^{n} x_i^2 \end{bmatrix} \begin{Bmatrix} a_1 \\ a_2 \end{Bmatrix} = \begin{Bmatrix} \sum_{i=1}^{n} y_i \\ \sum_{i=1}^{n} y_i x_i \end{Bmatrix}$$

Aproximación Cuadrática

Datos (x_i, y_i) i=1, n
Base $\{\varphi_1(x), \varphi_2(x), \varphi_3(x)\}=\{1, x, x^2\}$
Encontrar la matriz de coeficientes Φ y la matriz de coeficientes del sistema de ecuaciones lineales a resolver para obtener los coeficientes a_i.

INTEGRACIÓN NUMÉRICA

1 ACERCA DE LOS PROBLEMAS DE LA INTEGRACIÓN Y LA DERIVACIÓN NUMÉRICAS EN GENERAL

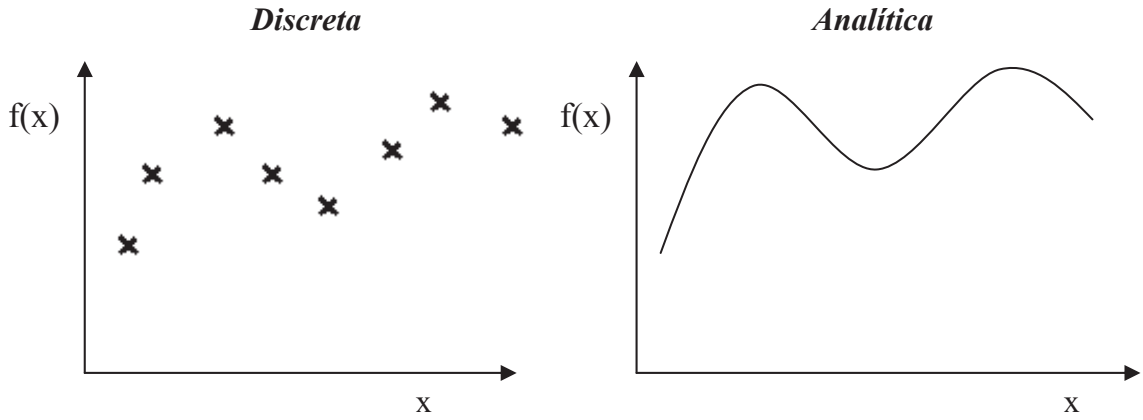
En distintos problemas de matemática aplicada es necesario calcular el valor de la derivada de una función en un punto o el valor de la integral de una *función conocida en forma analítica*; dichas operaciones en general no ofrecen dificultad, aunque en ciertas circunstancias es engorroso su cálculo o no resulta simple su implementación en un sistema computacional.

Cuando la *función es conocida en forma discreta*, el problema del cálculo de una derivada en un punto o de una integral de dicha función no es tan directo.

El propósito de esta unidad temática es abordar el cálculo de integrales definidas de funciones dadas en forma analítica o en forma discreta. Se plantearán los distintos métodos para una sola variable, dado que la extensión de los métodos a dos o tres dimensiones es posible.

A lo largo de toda esta unidad se supone que se tiene una función $y = f(x)$: R $\longrightarrow$ R, que es

- no singular,
- continua (al menos por tramos) y
- está dada en forma analítica o discreta.



f(x) discreta se conoce en x_i, con i = 0,1,2,...,n.

f(x) analítica se conoce en todo $x \in [x_0;x_n]$.

Se busca encontrar:

$$I = \int_a^b f(x)dx \quad \text{o} \quad V = \left.\frac{df(x)}{dx}\right|_{x=a} = f'(a),$$

que en términos generales se pueden expresar mediante algún operador lineal L(·),

$$I = L[f(x)] \quad \text{o} \quad V = L[f(x)].$$

En ambos casos es posible encontrar un polinomio (interpolante) que pase por puntos conocidos, y sobre esa aproximación, aplicar el operador de interés.

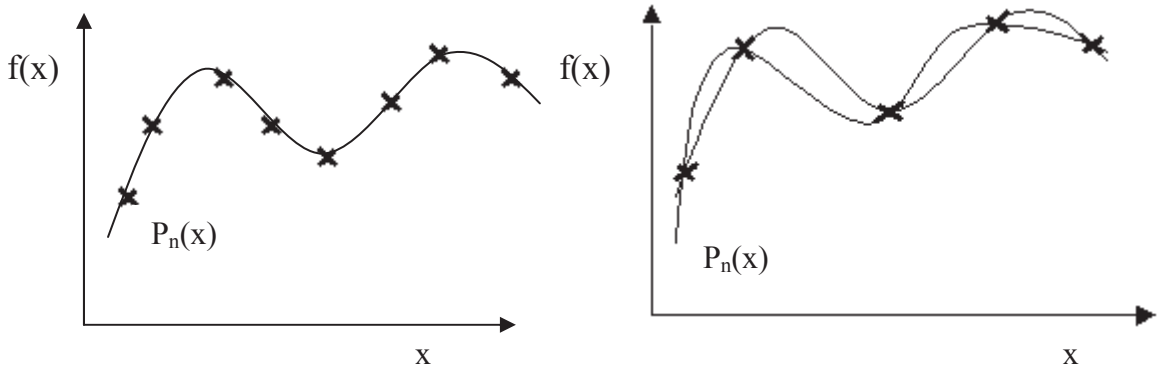
Si f(x) está dada en forma discreta es posible ***interpolar*** f(x) colocando un polinomio $P_n(x)$, de grado n, por los (n+1) puntos datos.

Si f(x) está dada en forma ***analítica*** se pueden *"extraer"* esos (n+1) puntos, es decir que se puede obtener como discreta evaluando la función f(x) en los valores x_i, i=0,...,n, para obtener los n+1 puntos $(x_i ; y_i = f(x_i))$, i = 0,1,2,...,n.

Así resulta

$$f(x) = P_n(x) + E_n(x),$$

donde $E_n(x)$ es el **error de interpolación**, $E_n(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!}(x - x_0)\cdots(x - x_n)$.



Cualquier operador lineal L[x] (como la integral o la derivada) aplicado a f(x) resulta:

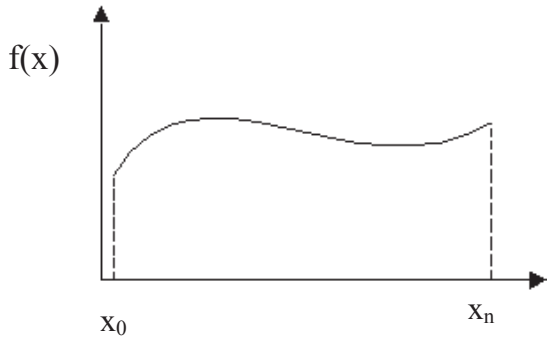
$$L\,[f(x)] = L\,[P_n(x)] + L\,[E_n(x)].$$

2 INTEGRACIÓN NUMÉRICA

Se busca encontrar la integral definida I ∈ R,

$$I = \int_{X_0}^{X_n} f(x)dx\,,$$

asumiendo que y = f(x): R →R
cumple las condiciones planteadas en el apartado 1 (es no singular en $[x_0;x_n]$; es continua, al menos por tramos, en $[x_0;x_n]$; está dada en forma analítica o discreta).



En virtud de la igualdad

$$I = \int_{X_0}^{X_n} f(x)dx = \int_{X_0}^{X_n} \big(P_n(x) + E_n(x)\big)dx = \int_{X_0}^{X_n} P_n(x)dx + \int_{X_0}^{X_n} E_n(x)dx = I_n + \mathcal{E}_n\,,$$

se evalúa la integral definida I como la suma

$$I = I_n + \mathcal{E}_n. \qquad (1)$$

En esta suma, I_n se llama **cuadratura** y $\mathcal{E}_n$ es el error de truncamiento.

Todos los métodos que veremos tienen la misma estructura: la cuadratura I_n se expresa como una suma $I_n = \sum_{j=0}^{n} \omega_j f(x_j)$, en que los coeficientes ω_i son dados por cada método.

Se define como **orden de la regla de cuadratura (integración)**, al máximo grado del polinomio que dicha regla integra en forma exacta, es decir, para el que $\mathcal{E}_n = 0$.

Veremos dos tipos de **cuadratura o regla de integración**. Los coeficientes ω_j se determinan en cada regla de integración, según la cantidad de puntos con que se formula cada regla.

2.1 Cuadratura de Newton-Cotes

- Los valores x_j donde se conoce $f(x_j)$ son predeterminados. Son datos fijos para la Regla de Integración.
- Los ω_j se determinan en cada regla para esos valores x_j.
- El paso puede ser fijo o variable. Se denomina ***paso*** a la distancia entre las abscisas dato.

2.2 Cuadratura de Gauss-Legendre

- Los valores x_j y los coeficientes ω_j se determinan en cada Regla de Integración.

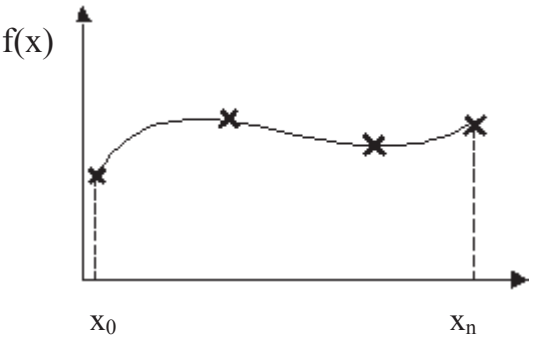
3 REGLAS DE INTEGRACIÓN DE NEWTON – COTES

Dada y = f(x): R ⟶ R en forma discreta mediante) puntos $(X_i ; Y_i)$, i = 0,...,n, se coloca un polinomio de grado n

$$P_n(x) = \sum_{i=0}^{n} Y_i l_i(x),$$

por el método de polinomios de Lagrange. En este caso cada polinomio base es

$$l_i(x) = \prod_{\substack{j=0 \\ j\neq i}}^{n} \frac{(x - x_j)}{(x_i - x_j)},$$



y el grado del polinomio P_n y la cantidad de polinomios bases $l_i(x)$ dependen del número de puntos datos, n+1.

Así se obtiene $f(x) = P_n(x) + E_n(x)$ y se tiene que

$$I = \int_{X_0}^{X_n} f(x)dx = \int_{X_0}^{X_n} [P_n(x) + E_n(x)]dx = \int_{X_0}^{X_n} \sum_{i=0}^{n} Y_i l_i(x)dx + \int_{X_0}^{X_n} E_n(x)dx,$$

$$I = \sum_{i=0}^{n} \int_{X_0}^{X_n} Y_i l_i(x)dx + \mathcal{E}_n = \sum_{i=0}^{n} Y_i \underbrace{\int_{X_0}^{X_n} l_i(x)dx}_{\omega_i} + \mathcal{E}_n.$$

Así, $I = I_n + \mathcal{E}_n$, donde

$$I_n = \sum_{i=0}^{n} \omega_i Y_i \quad con \quad \omega_i = \int_{X_0}^{X_n} l_i(x)dx$$

y $\mathcal{E}_n = \int_{X_0}^{X_n} E_n(x)\,dx.$ $\int_{X0}^{Xn} E_n(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!}\int_{X0}^{Xn}(x-x_0)\cdots(x-x_n)dx$

El error es proporcional a la derivada de orden (n+1) de la función a integrar, y al valor de la integral del primer polinomio (grado n+1) para el cual se comete error.
Los distintos métodos surgen de interpolar mediante polinomios de distintos grados.

3.1 Regla de los Trapecios

La regla de los trapecios es una regla de integración de orden 1, porque es exacta para polinomios de grado 1. Permite aproximar la integral I mediante la fórmula

$$\boxed{I = \frac{h_i}{2}(Y_i + Y_{i+1}) - \frac{1}{12}h_i^{\,3}f''(\theta),}$$

siendo θ un valor entre x_i y x_{i+1}.
Veremos el desarrollo de esta regla y el cálculo de su error de distintas maneras:
3.1.1 Desarrollo de la Regla de los Trapecios mediante integración del Polinomio Interpolante.
3.1.2 Cálculo del error de la regla de los trapecios usando serie de Taylor.
3.1.3 Cálculo del error de la regla de los trapecios a partir del error de interpolación.
3.1.4 Regla de los trapecios y su error, por el método de los coeficientes indeterminados.

3.1.1 Desarrollo de la Regla de los Trapecios mediante Integración Del Polinomio Interpolante.

Es una regla de integración de Newton – Cotes por 2 puntos $(x_i;y_i)$, $(x_{i+1};y_{i+1})$. Dada

$$I = \int_{x_i}^{x_{i+1}} f(x)\,dx,$$

se interpola por colocación mediante un polinomio de grado 1:

$$f(x) = P_1(x) + E_1(x), \quad \text{con} \quad P_1(x) = Y_i l_i(x) + Y_{i+1} l_{i+1}(x), \tag{3}$$

donde

$l_i(x) = \frac{x - x_{i+1}}{x_i - x_{i+1}} = -\frac{x - x_{i+1}}{x_{i+1} - x_i}$ es una recta que vale 1 en x_i y 0 en x_{i+1},

$l_{i+1}(x) = \frac{x - x_i}{x_{i+1} - x_i}$ es una recta que vale 0 en x_i y 1 en x_{i+1}.

El error de interpolación es $E_1(x) = \frac{f^{(2)}(\theta)}{2!}(x - x_i)(x - x_{i+1})$, siendo $\theta \in (x_i ; x_{i+1})$.

Integrando (3) se obtiene

$$I = \int_{x_i}^{x_{i+1}} [P_1(x) + E_1(x)]dx = \int_{x_i}^{x_{i+1}} [Y_i l_i(x) + Y_{i+1} l_{i+1}(x)]dx + \int_{x_i}^{x_{i+1}} E_1(x)dx,$$

y, llamando **paso h_i** a la diferencia x_{i+1}-x_i y operando, se puede llegar a

$$I = h_i\left[\frac{Y_i}{2} + \frac{Y_{i+1}}{2}\right] + \mathcal{E}_1 = I_1 + \mathcal{E}_1,$$

con

$$\boxed{I_1 = \frac{h_i}{2}(Y_i + Y_{i+1}),} \tag{4}$$

y donde

$$\omega_i = \frac{h_i}{2} = \int_{x_i}^{x_{i+1}} l_i(x)dx,$$

$$\omega_{i+1} = \frac{h_i}{2} = \int_{x_i}^{x_{i+1}} l_{i+1}(x)dx.$$

Observación: como en el método de los trapecios se trabaja con un solo intervalo, ocasionalmente nos referiremos al paso h_i sencillamente como paso h, a lo largo de esta sección.

3.1.2 Cálculo del Error en la Regla De Trapecios a partir del Error de Interpolación

Sabemos que al calcular la integral definida de la función f entre x_a y x_b por el método de los trapecios, se comete un error $\mathcal{E}_1$:

$$I = \int_{x_b}^{x_a} f(x)dx = I_1 + \mathcal{E}_1 ,$$

donde

$$I_1 = \frac{h}{2}Y_a + \frac{h}{2}Y_b \text{ y}$$

$$\mathcal{E}_1 = \int_{x_a}^{x_b} E_1(x)dx = \int_{x_a}^{x_b} \frac{f^{(2)}(\xi)}{2!}(x - x_a)(x - x_b)dx = \frac{f^{(2)}(\xi)}{2!}\int_{x_a}^{x_b}(x - x_a)(x - x_b)dx . \qquad (5)$$

Se propone hacer un cambio de variable mediante

$$\frac{x - x_a}{x_b - x_a} = \frac{t - 0}{1 - 0},$$

Definiendo $h=x_b-x_a$, es posible despejar,

$$x - x_a = \underbrace{(x_b - x_a)}_{h}t,$$

$$x - x_b + (x_b - x_a) = ht$$

$$x - x_b + h = ht$$

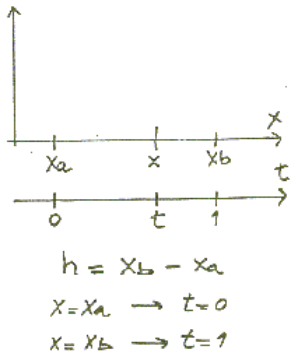
así resulta

$$x = x_a + ht$$

$$x - x_a = ht,$$

$$x - x_b = h(t-1)$$

$$dx = hdt.$$



Así se puede hallar

$$\int_{x_a}^{x_b}(x - x_a)(x - x_b)dx = \int_0^1 ht \cdot h(t-1) \cdot hdt = h^3\int_0^1 (t^2 - t)dt = -h^3\frac{1}{6},$$

de manera que, sustituyéndola en (5), se obtiene

$$\mathcal{E}_1 = \frac{f^{(2)}(\xi)}{2!}\left(-\frac{1}{6}\right)h^3 = -\frac{1}{12}h^3 f^{(2)}(\xi), \text{ para cierto punto } \xi \in (x_a, x_b).$$

3.1.3 Cálculo Del Error De La Regla De Los Trapecios usando Serie De Taylor

Por definición $E_1 = I - I_1$. Dada

$$I = \int_{X_i}^{X_{i+1}} f(x)dx\,,$$

si existe la primitiva de f(x), es decir, si existe una función G(x) tal que

$$\frac{dG(x)}{dx} = G'(x) = f(x)\,,$$

la integral es

$$I = G(x_{i+1}) - G(x_i)\,.$$

Se puede desarrollar $G(x_{i+1})$ en Serie de Taylor alrededor de x_i:

$$G(x_{i+1}) = G(x_i) + h_i G'(x_i) + \frac{h_i^{\,2}}{2}G''(x_i) + \frac{h_i^{\,3}}{6}G'''(x_i) + \cdots,$$

reordenando y sustituyendo f por G' se obtiene

$$\begin{aligned} G(x_{i+1}) - G(x_i) &= h_i f(x_i) + \frac{h_i^{\,2}}{2}f'(x_i) + \frac{h_i^{\,3}}{6}f''(x_i) + \cdots \\ I &= h_i f(x_i) + \frac{h_i^{\,2}}{2}f'(x_i) + \frac{h_i^{\,3}}{6}f''(x_i) + \cdots \end{aligned} \qquad (6)$$

Por otra parte, de (4), se tiene

$$I_1 = \frac{h_i}{2}\left[Y_i + Y_{i+1}\right] \quad \text{con} \quad Y_{i+1} = f(x_{i+1}) \quad \text{y} \quad Y_i = f(x_i). \qquad (7)$$

Si se desarrollan Y_{i+1} e Y_i en serie de Taylor alrededor de x_i, se obtiene:

$$Y_{i+1} = Y_i + h_i f'(x_i) + \frac{h_i^{\,2}}{2}f''(x_i) + \frac{h_i^{\,3}}{6}f'''(x_i) + \cdots$$

$$Y_i = Y_i$$

Sustituyendo en (6), queda

$$I_1 = \frac{h_i}{2}\left(Y_i + Y_i + h_i f'(x_i) + \frac{h_i^{\,2}}{2}f''(x_i) + \frac{h_i^{\,3}}{6}f'''(x_i) + \cdots \right),$$

$$I_1 = Y_i h_i + \frac{h_i^{\,2}}{2}f'(x_i) + \frac{h_i^{\,3}}{4}f''(x_i) + \frac{h_i^{\,4}}{12}f'''(x_i) + \cdots$$

y, reemplazando esta última expresión y la expresión (6) en (1), se obtiene la expresión del **error** en la regla de trapecios:

$$\boxed{E_1 = I - I_1 = \left(\frac{1}{6} - \frac{1}{4}\right)h_i^{\,3}f''(x_i) = -\frac{1}{12}h_i^{\,3}f''(\theta), \quad \theta \in (x_i, x_{i+1})}.$$

Decimos que el error en la regla de trapecios es del orden de h^3: $O(h^3)$.

Observación: no se debe confundir el orden de la regla de integración, que en este caso es 1, con el orden del error para la regla de integración, que en este caso es $O(h^3)$.

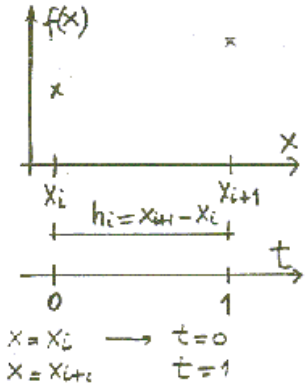
Hemos obtenido la expresión completa para la **regla de los trapecios**:

$$I = \frac{h_i}{2} Y_i + \frac{h_i}{2} Y_{i+1} - \frac{1}{12} h_i^{\,3} f''(\theta),$$

donde $\theta \in (x_i, x_{i+1})$.

3.1.4 *Regla De Los Trapecios Por El Método De Los Coeficientes Indeterminados*

Para calcular $I = \int_{X_i}^{X_{i+1}} f(x)dx$, se busca trabajar en el dominio $t \in [0,1]$.



Para ello se consideran

$x = h\,t + x_i$,
$dx = h \cdot dt$
y $G(t) = f(x) = f(x_i + ht)$.

Este método propone calcular la integral de una función G de la siguiente manera:

$$\int_0^1 G(t)dt = a_0 G(0) + a_1 G(1) + R,$$

donde R es el error cometido al integrar por esta regla y los coeficientes a_0 y a_1 se deben determinar de manera que la regla sea exacta cuando se integren las funciones {1, t} (es decir que para esas funciones debe ser R=0).

Integral	Resultado exacto	Resultado hallado por la regla
$\int_0^1 1dt$	1	$a_0 1 + a_1 1$
$\int_0^1 tdt$	$\frac{1}{2}$	$a_0 0 + a_1 1$

El requisito de que la regla sea exacta para integrar las funciones 1 y t nos permite plantear un sistema de ecuaciones:

$$\begin{cases} a_0 1 + a_1 1 = 1 \\ a_0 0 + a_1 1 = \frac{1}{2} \end{cases}$$

de donde resultan $a_0 = a_1 = \frac{1}{2}$.

Entonces

$$\int_0^1 G(t)dt = \frac{1}{2} G(0) + \frac{1}{2} G(1) + R \quad \text{y, así}$$

$$I = \int_{x_i}^{x_{i+1}} f(x)dx = \int_0^1 G(t)h\,dt = h\left(\frac{1}{2} G(0) + \frac{1}{2} G(1) + R\right) = h\left[\frac{1}{2} f(x_i) + \frac{1}{2} f(x_{i+1})\right] + h\,R.$$

Para determinar el error de truncamiento R, se considera

$R = \frac{G^{(n)}(\theta)}{n!}\alpha_n$ siendo α_n el error que se comete al integrar un polinomio de grado n.

La *regla de los trapecios* es exacta (por definición) para polinomios hasta grado 1.
Para polinomios cuadráticos,

$$\int_0^1 t^2 dt = \frac{1}{3} = \frac{1}{2}G(0) + \frac{1}{2}G(1) + \alpha_2 = 0 + \frac{1}{2} + \alpha_2,$$

de donde $\alpha_2 = -\frac{1}{6}$.

Así $R = -\frac{1}{6}\frac{G^{(2)}(\theta)}{2!}$.

Para ponerlo en términos de f(x), teniendo en cuenta que

$$G' = \frac{dG}{dt} = \frac{df}{dx}\frac{dx}{dt} = f'h \quad y \quad G'' = \frac{dG'}{dt} = \frac{d}{dx}(f'h)\frac{dx}{dt} = f''h^2,$$

se llega a $R = -\frac{1}{6}\frac{f^{(2)}(\theta)}{2!}h^2$.

La regla de los trapecios resulta: $$I = \int_{X_i}^{X_{i+1}} f(x)dx = h_i\left[\frac{Y_i}{2} + \frac{Y_{i+1}}{2}\right] - \frac{1}{12}h_i{}^3 f^{(2)}(\xi).$$

EJEMPLO: Sea f(x) = sen(x). Se plantea la integral para los intervalos [0, π/2] y [0, π].

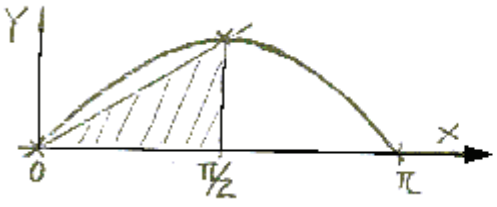
En forma **exacta**:

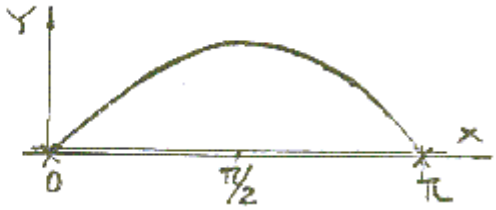
$$I = \int_{X_i}^{X_{i+1}} sen(x)dx = \cos(x_i) - \cos(x_{i+1}).$$

$x_i = 0$; $x_{i+1} = \pi/2$
$I = \cos(0) - \cos(\pi/2)$
$I = 1$

$x_i = 0$; $x_{i+1} = \pi$
$I = \cos(0) - \cos(\pi)$
$I = 2$

Resolviendo por la **regla de los trapecios,** $I_1 = \frac{h}{2}[Y_i + Y_{i+1}]$, se obtiene los resultados aproximados:





$h = \pi/2$
$I_1 = \pi/2*1/2*[\ sen(0) + sen(\pi/2)]$
$I_1 = \pi/4 = 0{,}7854$

$h = \pi$
$I_1 = \pi/2[\ sen(0) + sen(\pi)]$
$I_1 = 0$

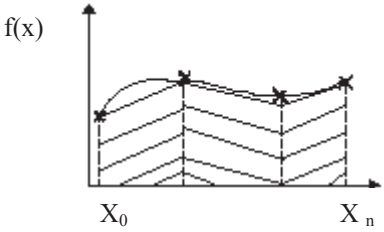
3.2 Regla De Los Trapecios Múltiple o Trapecios Compuesta

$$\boxed{I = \frac{h}{2}\left[Y_0 + 2\sum_{i=1}^{n-1} Y_i + Y_n \right] + -\frac{(x_n - x_0)}{12} h^2 f^{(2)}(\xi)}$$

3.2.1 Desarrollo de la fórmula de los trapecios múltiple y su error

Se busca $I \in R$,

$$I = \int_{X_0}^{X_n} f(x)dx .$$



Se divide el intervalo $[x_0; x_n]$ en subintervalos $[x_0; x_1]$, $[x_1; x_2]$, …, $[x_{n-1}; x_n]$. Así

$$I = \int_{X_0}^{X_1} f(x)\,dx + \int_{X_1}^{X_2} f(x)dx + \cdots + \int_{X_{n-1}}^{X_n} f(x)\,dx .$$

En cada uno de los n subintervalos se aplica la regla de los trapecios,

$$I = h_0 \frac{(Y_0 + Y_1)}{2} + \mathcal{E}_1(h_0{}^3) + h_1 \frac{(Y_1 + Y_2)}{2} + \mathcal{E}_1(h_1{}^3) + \ldots\ldots\ldots\ldots + h_{n-1} \frac{(Y_{n-1} + Y_n)}{2} + \mathcal{E}_1(h_{n-1}{}^3) .$$

Si todos los intervalos tienen igual longitud h_i=h, esa fórmula se simplifica y se tiene la **regla de trapecios múltiple**:

$$\boxed{I = h\left[\frac{Y_0}{2} + \sum_{i=1}^{n-1} Y_i + \frac{Y_n}{2} \right] + \mathcal{E}_{1M}},$$

donde $\mathcal{E}_{1M}$ es el error total que se acumula al sumar los n errores provenientes de la aplicación de la regla en cada subintervalo. Para aproximar ese error $\mathcal{E}_{1M}$, hacemos:

$\mathcal{E}_{1M} = \sum_{i=1}^{n} -\frac{1}{12} h^3 f^{(2)}(\theta_i)$, donde $\theta_1 \in (x_0, x_1)$, ..., $\theta_n \in (x_{n-1}, x_n)$. Entonces,

$$\mathcal{E}_{1M} = -\frac{1}{12} h^3 \sum_{i=1}^{n} f^{(2)}(\theta_i) . \qquad (8)$$

Se toma al promedio de las derivadas segundas en puntos interiores a cada subintervalo como aproximación de la derivada segunda en un cierto punto ξ interior al intervalo $[x_0; x_n]$:

$f^{(2)}(\xi) \cong \frac{1}{n} \sum_{i=1}^{n} f^{(2)}(\theta_i)$. Esta expresión nos permite sustituir $\sum_{i=1}^{n} f^{(2)}(\theta_i)$ por $n \cdot f^{(2)}(\xi)$ en (8):

$$\mathcal{E}_{1M} = -\frac{1}{12} h^3 n f^{(2)}(\xi) .$$

Pero en esta expresión nos queda el error en términos de h y de n. Queremos eliminar n; como se ha supuesto que todos los h_i son iguales, entonces es $h = \frac{(x_n - x_0)}{n}$ y así,

$$\mathcal{E}_{1M} = -\frac{1}{12} h h^2 n f^{(2)}(\xi) = -\frac{1}{12} \frac{(x_n - x_0)}{n} h^2 n f^{(2)}(\xi) .$$

De manera que el error en la regla de trapecios múltiple es

$$\mathcal{E}_{1M} = -\frac{(x_n - x_0)}{12} h^2 f^{(2)}(\xi).$$

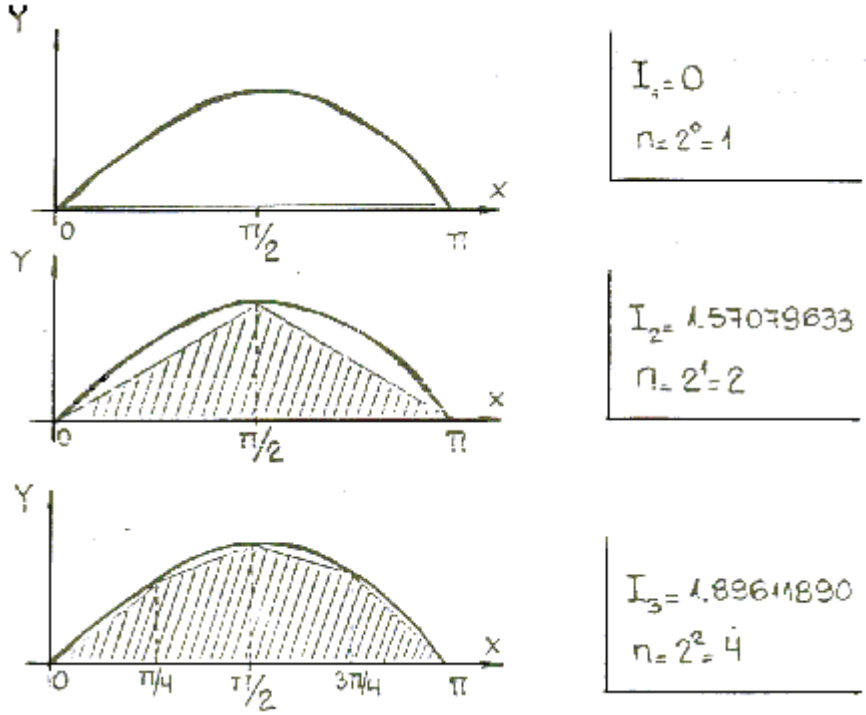
Observación: al pasar de la regla de trapecios a la regla de trapecios compuesta el orden del error pasó de $O(h^3)$ a $O(h^2)$, disminuyendo la precisión.

EJEMPLO:

$$I = \int_0^{\pi} \text{sen}(x)dx$$

$$I \cong h \cdot \left[\frac{Y_0}{2} + \sum_{i=1}^{n-1} Y_i + \frac{Y_n}{2} \right] = h \cdot S, \text{ donde } Y_i = \text{sen}(x_i),\ i=0,\ldots,n.$$

X	Y	$h_1 = \pi$	$h_2 = \pi/2$	$h_3 = \pi/4$
0	0	1	1	1
$\pi/4$	$\sqrt{2}/2$	0	0	2
$\pi/2$	1	0	2	2
$3\pi/4$	$\sqrt{2}/2$	0	0	2
π	0	1	1	1
	0	S_1	S_2	S_3



El resultado exacto es $I = -\cos x \Big|_0^{\pi} = \cos 0 - \cos \pi = 2$. Es fácil concluir que la mejor aproximación es I_3.

Los errores cometidos por cada aplicación de la regla son, respectivamente $O(h_1^3)$, $O(h_2^2)$ y $O(h_3^2)$.

Nótese en el orden del error que *para un mismo valor de h* es más precisa la regla de trapecios simple que la de trapecios compuesta.

En nuestro ejemplo, la mejor aproximación es I_3 pero se debe a que trabaja con un paso h menor.

3.3 Algoritmo De Trapecios Múltiples

Caso en que f(x) está dada en forma DISCRETA y tenemos una tabla con 10 valores de x y de y, con h constante.

Algoritmo trapecios-múltiples-discreta
Var (X(10), Y(10), h, sum, int: **real**)
Var (i, **entero**)
Do for i=1,10
Escribir "Ingrese el valor de x", i
Leer X(i)
Escribir "Ingrese el valor de y", i
Leer Y(i)
End do
sum=Y(1)/2
Do for i=2,9
sum=sum+Y(i)
End do
sum=sum+Y(10)/2
h=(X(10)-X(1))/9
int=h*sum
Escribir "El valor de la integral de f(x) entre", X(1), "y", X(10), "es", int.
End

Caso en que f(x) está dada en forma ANALÍTICA. En este caso es posible definir f(x) según la "sintaxis" del lenguaje de programación que se utilice. A los efectos del pseudo código indicaremos esta definición mediante la orden o sentencia "DEFINICIÓN DE FUNCTION f(x)".

AlgoritmoTrapecios-múltiples-Equidistante-analítica
Var (x_0, x_n, n, h, sum, real)
Var (i **entera**)
DEFINICIÓN DE FUNCTION f(x)
h = (x_n - x_0)/n
x = x_0
sum = f(x_0)/2
DOFOR i =1, n-1
x = x +h
sum = sum + f(x)
ENDDO
sum = sum + f(x_n)/2
TrapEq = sum*h
END

3.4 Regla de Simpson

Es una regla de orden 3, es decir que integra en forma exacta polinomios de grado hasta 3. Permite aproximar la integral I mediante la fórmula

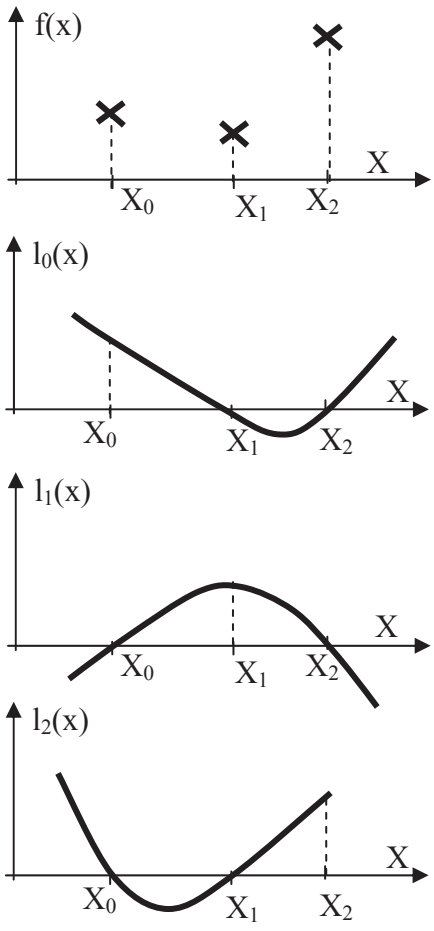
$$I = \frac{h}{3}\left[f(x_0) + 4f(x_1) + f(x_2)\right] - \frac{h^5}{90} f^{(4)}(\xi),$$

siendo ξ un valor entre x_0 y x_2.
Veremos el desarrollo de esta regla y el cálculo de su error de distintas maneras:
3.4.1 Desarrollo de la Regla de Simpson mediante integración del Polinomio Interpolante.
3.4.2 Regla Simpson por el método de los coeficientes indeterminados.
3.4.3 Error de la regla de Simpson.

3.4.1 Regla de Simpson mediante integración del polinomio interpolante.

Es una cuadratura de Newton – Cotes con n = 2, es decir con tres puntos. Se interpola mediante un polinomio de Lagrange de grado dos y luego se integra en forma aproximada ese polinomio.

$$I = \int_{X_0}^{X_2} f(x)dx,$$



con $f(x) = \sum Y_i l_i(x) + E_2(x)$, i = 0,1,2,

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

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

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

$$E_2(x) = \frac{f^{(3)}}{3!}(x - x_0)(x - x_1)(x - x_2)$$

Así $I = I_2 + \mathcal{E}_2$, donde

$$I_2 = \sum_{i=0}^{2} Y_i \omega_i$$

$$\omega_i = \int_{x_0}^{x_2} l_i(x)dx,\ i = 0,1,2.$$

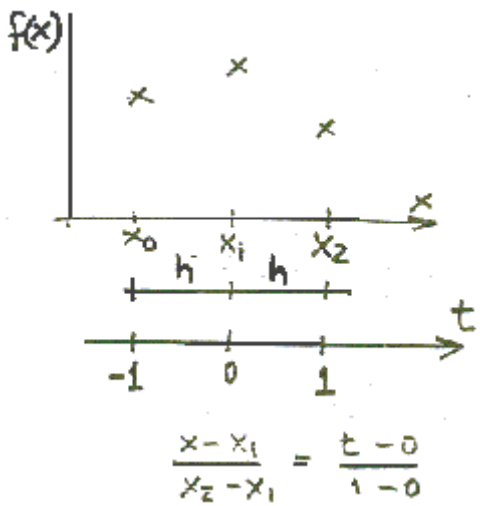
$$\mathcal{E}_2 = \int_{x_0}^{x_2} E_2(x)dx$$

Entonces, si los intervalos son iguales ($h_1 = h_2 = h$), se tiene:

$$I_2 = h\left[\frac{1}{3}Y_0 + \frac{4}{3}Y_1 + \frac{1}{3}Y_2\right].$$

3.4.2 Regla De Simpson Por El Método De Los Coeficientes Indeterminados

Se busca normalizar la $I = \int_{X_0}^{X_2} f(x)dx$ al dominio $t \in [0,1]$.



Para ello se considera:

$x = ht + x_1$
$dx = h\,dt$
$G(t) = f(ht+x_1)$
y se plantea

$$I = \int_{-1}^{1} hG(t)dt = h\int_{-1}^{1} G(t)dt = h\left(\sum_{i=-1}^{1}[C_i G_i] + R\right). \qquad (9)$$

Se propone resolver la integral de G por el método de los coeficientes indeterminados, es decir:

$$\int_{-1}^{1} G(t)dt = C_{-1}G(-1) + C_0 G(0) + C_1 G(1) + R\,,$$

con C_{-1}, C_0, C_1 a determinar, tales que la regla es exacta para $\{1, t, t^2\}$. Al igual que antes, resolvemos las integrales en forma exacta y según la regla, lo cual nos permite plantear un sistema de ecuaciones y hallar los coeficientes buscados.

$$\left.\begin{aligned} \int_{-1}^{1} 1dt &= 2 = C_{-1} + C_0 + C_1 \\ \int_{-1}^{1} t\,dt &= 0 = -1C_{-1} + 0C_0 + 1C_1 \\ \int_{-1}^{1} t^2 dt &= \frac{2}{3} = 1C_{-1} + 0C_0 + 1C_1 \end{aligned}\right\} \quad \begin{aligned} C_1 &= C_{-1} \\ C_1 &= \frac{1}{3} \\ C_0 &= \frac{4}{3} \end{aligned}$$

Sustituyendo en la ecuación (9), se llega a

$$I = h\left[\frac{1}{3}G(-1) + \frac{4}{3}G(0) + \frac{1}{3}G(1)\right] + hR$$

$$I = h\left[\frac{1}{3}f(x_0) + \frac{4}{3}f(x_1) + \frac{1}{3}f(x_2)\right] + hR = I_2 + \mathcal{E}_2$$

de donde

$$\boxed{I_2 = h\left[\frac{1}{3}Y_0 + \frac{4}{3}Y_1 + \frac{1}{3}Y_2\right]}.$$

3.4.3 Regla de Simpson - error

Para determinar el error de truncamiento R, se considera

$R = \frac{G^{(n)}(\theta)}{n!}\alpha_n$, siendo α_n el error que se comete al integrar un polinomio de grado n.

La regla de Simpson es exacta (por definición) para polinomios hasta grado 2. Como hicimos antes, para hallar el error, se aplica la regla a otros polinomios. Para polinomios de grado tres, resulta

$$\int_{-1}^{1} t^3 dt = 0 = \frac{1}{3}(-1)^3 + \frac{3}{3}(0) + \frac{1}{3}1^3 = 0,$$

lo cual nos está indicando que la regla **es exacta hasta para polinomios de grado 3**. Para determinar el error se sigue con polinomios de grado 4, y se obtiene

$$\int_{-1}^{1} t^4 dt = \frac{2}{5} = \frac{1}{3}(-1)^4 + \frac{4}{3}(0) + \frac{1}{3}1^4 + \alpha_4,$$

de donde $2/5 = 2/3 + \alpha_4$, es decir $\alpha_4 = -4/15$.

Así $R = -\frac{4}{15}\frac{G^{(4)}(\theta)}{4!} = -\frac{1}{90}G^{(4)}(\theta)$.

En términos de f(x), teniendo en cuenta que

$$G' = \frac{dG}{dt} = \frac{df}{dx}\frac{dx}{dt} = f'h, \quad G'' = \frac{dG'}{dt} = \frac{d}{dx}(f'h)\frac{dx}{dt} = f''h^2, \text{ etc.}$$

entonces $R = -\frac{h^4}{90}f^{(4)}(\theta)$, donde θ es un valor entre -1 y 1.

La Regla de Simpson resulta

$$I = I_2 + \mathcal{E}_2$$

$$I = h\left[\frac{1}{3}G(-1) + \frac{4}{3}G(0) + \frac{1}{3}G(1)\right] - h\frac{1}{90}G^{(4)}(\theta)$$

$$\boxed{I = h\left[\frac{1}{3}f(x_0) + \frac{4}{3}f(x_1) + \frac{1}{3}f(x_2)\right] - \frac{h^5}{90}f^{(4)}(\xi)},$$

donde ξ es un valor entre x_0 y x_2.

La regla de Simpson es una regla de orden 3 (integra en forma exacta polinomios de grado hasta 3) y tiene un error del orden de h^5: $O(h^5)$.

3.5 Regla de Simpson Compuesta

Es un ejercicio sencillo realizar la suma de sucesivas aplicaciones de la regla de Simpson, para llegar a la fórmula

$$\boxed{I = \frac{h}{3}\left[f(x_0) + \sum_{\text{i impares}} 4f(x_i) + \sum_{\text{i pares}} 2f(x_i) + f(x_n)\right] - \frac{(x_n - x_0)h^4}{90}f^{(4)}(\xi)}.$$

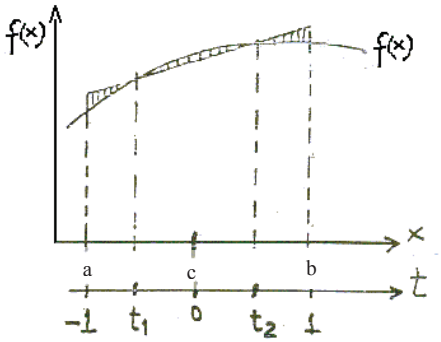
Acá, el error se halla sumando los errores provenientes de cada intervalo, de manera análoga a como se hizo para hallar el error en la regla de trapecios compuesta.

Al pasar de la regla de trapecios a la regla de trapecios compuesta el orden del error pasó de $O(h^3)$ a $O(h^2)$, disminuyendo la precisión; lo mismo sucede en el caso de la regla de Simpson, que pasa de un error de orden $O(h^5)$ a un error del orden $O(h^4)$ en Simpson compuesta.

4 CUADRATURA DE GAUSS

Planteamos el problema en el dominio unitario [-1, 1], sabiendo que se puede llevar a cualquier otro intervalo mediante un cambio de variables conveniente.

Como se observa en el gráfico, este método propone hallar los valores de abscisas t_1 y t_2 para aproximar la integral entre –1 y 1 de f(x) mediante el área bajo la **recta** graficada.



Esta regla de integración se define <u>buscando</u> los coeficientes y los puntos de evaluación de la función. Si

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

$x = ht + c$

$dx = h\,dt$

$G(t) = f(ht+c)$

$$I = \int_a^b f(x)dx = h\int_{-1}^{1} G(t)dt = h\left(\sum_{i=0}^{n}\omega_i G(t_i) + R\right) = h\sum_{i=0}^{n}\omega_i G(t_i) + \mathcal{E}\,.$$

4.1 <u>Regla De Dos Puntos Usando el Método De Coeficientes Indeterminados</u>

$$\int_{-1}^{1} G(t)dt = \omega_1 G(t_1) + \omega_2 G(t_2) + R$$

ω_1, ω_2, t_1, t_2 deben ser tales que el error de integración sea R = 0 para polinomios de hasta 3º grado. Comparando los resultados exactos de la integral con el resultado propuesto por la regla, se tiene:

$$\left.\begin{aligned}
&\int_{-1}^{1} 1dt = 2 = 1\omega_1 + 1\omega_2 \\
&\int_{-1}^{1} t\,dt = 0 = t_1\omega_1 + t_2\omega_2 \\
&\int_{-1}^{1} t^2dt = \frac{2}{3} = t_1{}^2\omega_1 + t_2{}^2\omega_2 \\
&\int_{-1}^{1} t^3dt = 0 = t_1{}^3\omega_1 + t_2{}^3\omega_2
\end{aligned}\right\} \quad \begin{aligned}&\omega_1 = \omega_2 = 1 \\ &-t_1 = t_2 = \frac{\sqrt{3}}{3}\end{aligned} \qquad (9)$$

y la regla queda:

$$I = h\left[G\left(\frac{-\sqrt{3}}{3}\right) + G\left(\frac{\sqrt{3}}{3}\right)\right] + \mathcal{E}\text{, o sea}$$

$$\boxed{I = h\left[f\left(-h\frac{\sqrt{3}}{3} + c\right) + f\left(h\frac{\sqrt{3}}{3} + c\right)\right] + \mathcal{E}}$$

EJEMPLO: Se desea calcular el volumen de una esfera de radio r mediante el cálculo de

$$V = \int_{-r}^{r} \pi \cdot ([\, r^2 - x^2 \,]^{1/2})^2 \, dx = \int_{-r}^{r} \pi \cdot [\, r^2 - x^2 \,] \, dx \ .$$

Llamando
$x = ht+c = ht+0$
$dx = h\ dt$,
$G(t) = f(ht+c) = f(ht+0) = \pi\ (r^2-(ht)^2)$
se plantea:

$$V = \int_{-1}^{1} \pi\left[r^2 - h^2 t^2\right] h dt \cong h\left(\omega_1 G(t_1) + \omega_2 G(t_2)\right)$$

$$= h\left(1 \cdot G\left(-\frac{\sqrt{3}}{3}\right) + 1 \cdot G\left(\frac{\sqrt{3}}{3}\right)\right) = h\left[f\left(h\left(-\frac{\sqrt{3}}{3}\right)\right) + f\left(h\left(\frac{\sqrt{3}}{3}\right)\right)\right]$$

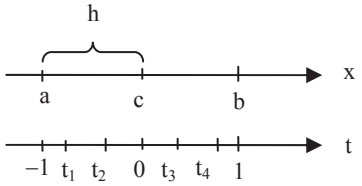
Como h = r, se obtiene:

$$V = r\pi\left(r^2 - r^2\left(-\frac{\sqrt{3}}{3}\right)^2\right) + r\pi\left(r^2 - r^2\left(\frac{\sqrt{3}}{3}\right)^2\right) = \pi r^3\left[\left(1-\frac{3}{9}\right) + \left(1-\frac{3}{9}\right)\right] = \frac{4}{3}\pi r^3 \ .$$

4.2 Generalización de la regla anterior

La regla de dos puntos presentada anteriormente puede generalizarse a más puntos. Es decir que dada la función f analíticamente, se puede replantear el problema en el dominio unitario igual que antes, pero con el número de puntos deseado. La idea de la transformación es siempre la misma:

$$h = \frac{b-a}{2}$$
$x = ht + c$
$dx = h\ dt$
$G(t) = f(ht+c)$



El sistema (9) deberá tener tantas ecuaciones como sea necesario para hallar las incógnitas ω_i y t_i. A continuación se incluye una tabla con las soluciones de (9) para los casos en que se plantea la regla para dos, tres y cuatro puntos:

Cantidad de Puntos de Gauss	Abscisa (t_i)	Coeficiente (ω_i)	Orden de la Derivada del Error de Truncamiento
2	$\pm\sqrt{3}/3$	1	4
3	±0.774596669 0.000000000	0.5555556 0.8888889	6
4	±0.861136312 ±0.339981044	0.3478548 0.6521452	8

5 EXTRAPOLACIÓN DE RICHARDSON

Toda vez que se tenga dos aproximaciones de una integral I (lo mismo vale para dos aproximaciones de derivadas, por ejemplo) es posible mejorar la aproximación con extrapolación de Richardson. Supongamos que se usa cierta regla de integración, con orden de error de h^n, para hallar dos aproximaciones de $I = \int_{x_0}^{x_n} f(x)dx$, utilizando dos pasos distintos, h_1 y h_2.

Parece que solo vale dos aproximaciones con la misma regla que tiene un valor dado de orden de error de truncamiento

Si, por ejemplo, se elige la regla de Trapecios Compuesta,
para un PASO h_1 se tiene un valor aproximado $I(h_1)$ con un error

$$E(h_1) = -\frac{(x_n - x_0)}{12} f''(\xi_1) h_1^{\,2},\ \xi_1 \in (x_0, x_n) \text{ y}$$

para un PASO h_2 se tiene un valor aproximado $I(h_2)$ con un error

$$E(h_2) = -\frac{(x_n - x_0)}{12} f''(\xi_2) h_2^{\,2},\ \xi_2 \in (x_0, x_n).$$

Asumiendo que $f''(\xi_1) \cong f''(\xi_2)$,

$$\frac{E(h_1)}{E(h_2)} = \left(\frac{h_1}{h_2}\right)^2 = \beta.$$

Si, por ejemplo, se elige la regla de Simpson Compuesta,
para un PASO h_1 se tiene un valor aproximado $I(h_1)$ con un error

$$E(h_1) = -\frac{(x_n - x_0) h_1^{\,4}}{180} f^{(4)}(\xi_1),\ \xi_1 \in (x_0, x_n) \text{ y}$$

para un PASO h_2 se tiene un valor aproximado $I(h_2)$ con un error

$$E(h_2) = -\frac{(x_n - x_0) h_2^{\,4}}{180} f^{(4)}(\xi_2),\ \xi_2 \in (x_0, x_n).$$

Asumiendo, en este caso, que $f^{(4)}(\xi_1) \cong f^{(4)}(\xi_2)$,

$$\frac{E(h_1)}{E(h_2)} = \left(\frac{h_1}{h_2}\right)^4 = \beta.$$

En general, si se trabaja con una regla cuyo error es del orden de h^n, llamaremos

$$\beta = \left(\frac{h_1}{h_2}\right)^n.$$

Así, en cualquier caso,

$$I = I(h_1) + E(h_1) = I(h_2) + E(h_2)$$

y, operando,

$$I(h_2) - I(h_1) = E(h_1) - E(h_2) = E(h_2)\,[\beta - 1],$$

de donde

$$E(h_2) = \frac{I(h_2) - I(h_1)}{\beta - 1}.$$

Si al valor aproximado de la intergral $I(h_2)$, se le suma esta aproximación del error E(h2), se obtiene el siguiente **valor mejorado de la integral**:

$$I = I(h_2) + E(h_2) = I(h_2) + \frac{[I(h_2) - I(h_1)]}{\beta - 1}$$

y sumando, se tiene

$$\boxed{I = \frac{\beta\, I(h_2) - I(h_1)}{\beta - 1}}.$$

Notar que se utiliza h1, que se considera mayor que h2 (la peor situación)

El error que se comete depende del error de la regla utilizada. Si $h_2 < h_1$, y la regla utilizada tiene un error de orden n, el error que se comete es del orden de $h_1^{(n+2)}$: $E(h_1^{(n+2)})$. En particular, en los ejemplos analizados antes, si se usa la regla de trapecios compuesta, el error para la aproximación que da la extrapolación de Richardson es del orden de h_1^4: $E(h_1^4)$. Si se usa la regla de Simpson compuesta, el error es del orden de h_1^6: $E(h_1^6)$.

Observación: si por falta de puntos se combina la aplicación de una regla compuesta y la misma regla pero simple, por ejemplo si se aplica Simpson simple con un paso h_1 (el error es del orden de h_1^5) y Simpson compuesta con un paso h_2 (el error es del orden de h_2^4), el error de la nueva aproximación será del orden de h_1^6.

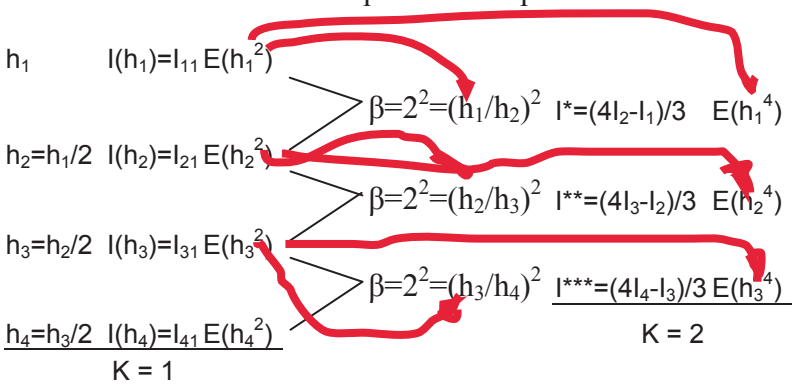
Se toma la de menos precisión y se suma 2

6 INTEGRACIÓN DE ROMBERG

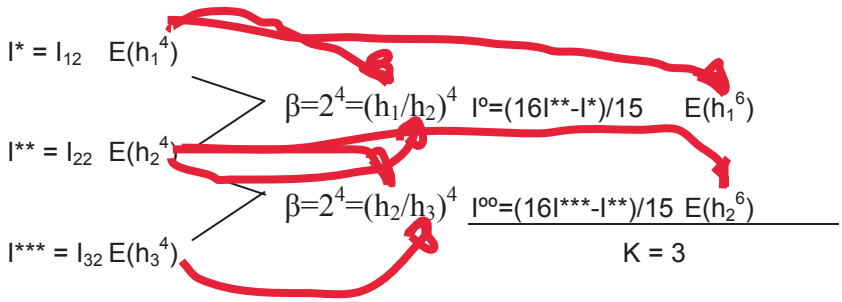
Se busca $I = \int_{X_0}^{X_n} f(x)dx$ con $f(x) : R \longrightarrow R$ **usando la regla de trapecios múltiples**.

Consiste en aplicar sucesivas extrapolaciones de Richardson, sobre una serie de aproximaciones de **trapecios múltiples**, para pasos que se reducen a la mitad.

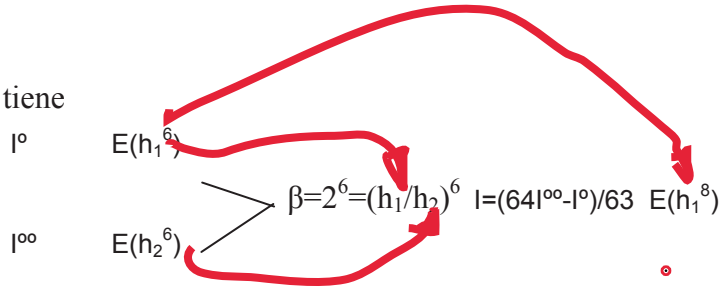
En la iteración k = 1 se tienen los valores de trapecios múltiple:



En las iteraciones k ≥ 2 se mejora con extrapolación de Richardson, por ejemplo en k = 2 se tiene:



En k =3 se tiene



En la iteración k = 1, las aproximaciones $I_{j,1}$, provienen de aplicar la regla de los trapecios múltiples, siendo j el nivel de divisiones por la mitad del paso original.

Las iteraciones k ≥ 2 que corresponden a mejoras por extrapolación de Richardson se pueden generalizar $I_{j,k} = \dfrac{4^{k-1} I_{j+1,k-1} - I_{j,k-1}}{4^{k-1} - 1}$

El criterio de parada de las iteraciones es:

$$\left| \frac{I_{1,k} - I_{1,k-1}}{I_{1,k}} \right| < \text{Tolerancia},$$

o bien k > Número Máximo de Iteraciones.

6.1 Algoritmo de Romberg

Algoritmo que halla la integral de una función dada numéricamente por 9 pares de valores (x, y).

```
Alg Romberg
Var (X(9), Y(9), h(4), Int(4,4), sum(4): real; i:entero)

Do for i=1,9
     Escribir "Ingrese el valor de x", i
     Leer X(i)
     Escribir "Ingrese el valor de y", i
     Leer Y(i)
End do
Do for i=1,4
     h(i)=(X(9)-X(1))/(2^(i-1))
End do

sum(1)=Y(1)+Y(9)
sum(2)=sum(1)+2*Y(5)
sum(3)=sum(2)+2*(Y(3)+Y(7))
sum(4)=sum(3)+2*(Y(2)+Y(4)+Y(6)+Y(8))

Do for i=1,4
   Int(i,1)=h(i)/2*sum(i)
End do

Do for i=1,3
   Int(i,2)=(4*Int(i+1,1)–Int(i,1))/3
End do
```

Do for i=1,2
  Int(i,3)=(16*Int(i+1,2)–Int(i,2))/15
End do

Int(1,4)=(64*Int(2,3)–Int(1,3))/63

Escribir "El valor mejorado de la integral es", Int(1,4).
END

FUNCTION Romberg (x_0, x_n, maxit, tol)
  Local I(10,10)
    n = 1
    $I_{1,1}$= TrapEq(x_0, x_n, n)
    Iter = 0
    DO
      Iter = iter + 1
      n = 2^{iter}
      $I_{iter+1,1}$ = TrapEq(x_0, x_n, n)
      DO k = 2, iter+1
        γ =2+iter-k
        $$I_{j,k} = \frac{4^{k-1} I_{j+1,k-1} - I_{j,k-1}}{4^{k-1} - 1}$$
      ENDDO
      Eps=Abs(($I_{1,iter+1}$-$I_{1,iter}$)/$I_{1,iter+1}$)
      IF((eps ≤ tol) or (iter ≥ maxit)) EXIT
    ENDDO
    Romberg = $I_{1,iter+1}$
END

EJEMPLO: *MÈTODO DE ROMBERG*

$$I = \int_0^{\pi} \text{sen}(x)dx \qquad I_{,j,k} = \frac{4^{k-1} I_{j+1,k-1} - I_{j,k-1}}{4^{k-1} - 1}$$

	k=1	k=2	k=3	k=4	k=5	k=6
		$I_{j,2} = \frac{4 I_{j+1,1} - I_{j,1}}{3}$	$I_{j,3} = \frac{16 I_{j+1,2} - I_{j,2}}{15}$	$I_{j,4} = \frac{64 I_{j+1,3} - I_{j,3}}{63}$	$I_{j,5} = \frac{256 I_{j+1,4} - I_{j,4}}{255}$	$I_{j,6} = \frac{1024 I_{j+1,5} - I_{j,5}}{1023}$
h	I (trapecios)					
$h_1=\pi$	$I_1=0$	2,09439511	1,99857073	2,00000555	1,9999999	2,0000000
$h_2=\pi/2$	$I_2=1,57079633$	2,00455976	1,9998313	2,0000001		
$h_3=\pi/4$	$I_3=1,89611890$	2,00026917	1,99999975	2,00000000		
$h_4=\pi/8$	$I_4=1,97423160$	2,00001695	2,00000000			
$h_5=\pi/16$	$I_5=1,99357034$	2,00000103				
$h_6=\pi/32$	$I_6=1,98839336$					
	$O(h^2)$	$O(h^4)$	$O(h^6)$	$O(h^8)$	$O(h^{10})$	$O(h^{12})$

DERIVACIÓN NUMÉRICA

1 INTRODUCCIÓN

El propósito de esta de esta Unidad es lograr obtener aproximaciones numéricas de derivadas de distinto orden de funciones dadas en forma discreta o continua; y su posible aplicación en la solución de ecuaciones diferenciales.

Si bien es posible obtener derivadas aproximadas a partir de las interpolaciones con polinomios de Newton, se ha optado por obtenerlas desde el concepto de Serie de Taylor.

Para iniciar el desarrollo se recuerdan las definiciones de derivada y Serie de Taylor, conceptos sobre los que se basa el siguiente desarrollo. Es así que se define la derivada de una función ***f*** en un punto x_0 como el siguiente límite:

$$f'(x_0) = \lim_{h \to 0} \frac{f(x_0 + h) - f(x_0)}{h}.$$

Esta definición lleva implícito un método de aproximación numérica:

$$f'(x) \cong \frac{f(x+h) - f(x)}{h}.$$

Esta aproximación numérica se denomina **derivación numérica** de ***f*** con paso ***h***.

La utilización de la serie de Taylor para el desarrollo de una función ***f(x)***, alrededor de un punto x_s, permite calcular en forma aproximada el valor de la función en un punto cercano $x = x_s + nh$; *"n"* es un número entero positivo o negativo.

Así:

$$f(x_s + nh) = f(x_s) + n\,h\,f'(x_s) + \frac{(nh)^2 f''(x_s)}{2!} + \frac{(nh)^3 f'''(x_s)}{3!} + \frac{(nh)^4 f'^{v}(x_s)}{4!} + O(h^5).$$

A partir del desarrollo de Taylor, resulta posible relacionar **valores de la función en el entorno** (vecindad) de un punto x_s **con valores de la función y sus derivadas en el punto x_s.**

Así en:

$n = -2$ $\qquad f_{s-2} = f_s - 2h f'_s + \frac{4h^2}{2!} f''_s - \frac{8h^3}{3!} f'''_s + \frac{16h^4}{4!} f_s'^{v} + O(h^5),$

$n = -1$ $\qquad f_{s-1} = f_s - h f'_s + \frac{h^2}{2!} f''_s - \frac{h^3}{3!} f'''_s + \frac{h^4}{4!} f_s'^{v} + O(h^5),$

$n = 0$ $\qquad f_s = f_s,$

$n = 1$ $\qquad f_{s+1} = f_s + h f'_s + \frac{h^2}{2!} f''_s + \frac{h^3}{3!} f'''_s + \frac{h^4}{4!} f_s'^{v} + O(h^5),$

$n = 2$ $\qquad f_{s+2} = f_s + 2h f'_s + \frac{4h^2}{2!} f''_s + \frac{8h^3}{3!} f'''_s + \frac{16h^4}{4!} f_s'^{v} + O(h^5).$

2 DERIVADAS PRIMERAS

Analizaremos las derivadas primeras en la vecindad del punto:

2.1 Hacia adelante

Considerando el desarrollo en Serie de Taylor de la función en $n = 1$, el valor aproximado de la función en $x = x_s + h$ es,

$$f_{s+1} = f_s + h f'_s + O(h^2),$$

siendo el error de truncamiento del orden $O(h^2)$:

De donde es posible despejar el valor de la derivada primera de la función en $x = x_s$, en la forma

$$\boxed{f'_s = \frac{1}{h}\left[f_{s+1} - f_s\right] - O(h)}$$

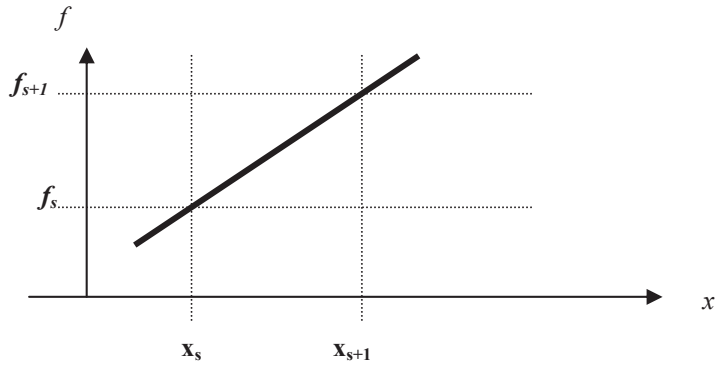
o truncando los terminos de orden $O(h)$; se puede expresar en forma aproximada,

$$f'_s \cong \frac{1}{h}\left[f_{s+1} - f_s\right] = \frac{f_{s+1} - f_s}{x_{s+1} - x_s}$$

o bien en términos de diferencias divididas,

$$f'_s \cong f\left[x_{s+1} - x_s\right].$$

Gráficamente se observa que la derivada es simplemente la pendiente de la secante que pasa por los puntos (x_s, f_s) y (x_{s+1}, f_{s+1}):



2.2 Hacia atrás

Considerando el desarrollo en Serie de Taylor de la función para $n = -1$, el valor aproximado de la función es,

$$f_{s-1} = f_s - h f'_s + O(h^2)$$

siendo el error de truncamiento del orden $O(h^2)$:

Es posible despejar el valor de la derivada primera de la función en $x=x_s$, en la forma,

$$\boxed{f'_{s-1} = \frac{1}{h}\left[f_s - f_{s-1}\right] + O(h)},$$

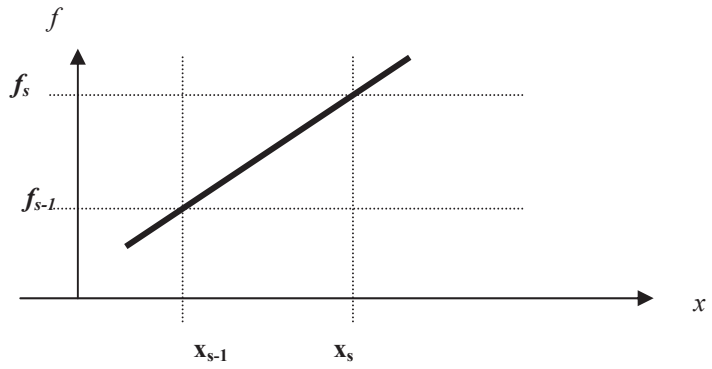
siendo el error de truncamiento de la derivada del orden de $O(h)$. Si se trunca e desarrollo en serie de la derivada, resulta una aproximación de la misma de la forma,

$$f'_s \cong \frac{1}{h}\left[f_s - f_{s-1}\right] = \frac{f_s - f_{s-1}}{x_s - x_{s-1}}$$

o bien en términos de diferencias divididas,

$$f'_s \cong f\left[x_s - x_{s-1}\right]$$

Gráficamente se observa que la derivada es simplemente la pendiente de la secante que pasa por los puntos (x_{s-1}, f_{s-1}) y (x_s, f_s):



2.3 Central

Teniendo en cuenta los desarrollos en serie de Taylor para $n = 1$ y $n = -1$:

$$f_{s+1} = f_s + h f'_s + \frac{h^2}{2} f''_s + \frac{h^3}{6} f'''_s + \cdots$$

$$f_{s-1} = f_s - h f'_s + \frac{h^2}{2} f''_s - \frac{h^3}{6} f'''_s + \cdots$$

y restando miembro a miembro,

$$f_{s+1} - f_{s-1} = 2h f'_s + 2\frac{h^3}{6} f'''_s + \cdots$$

se obtiene:

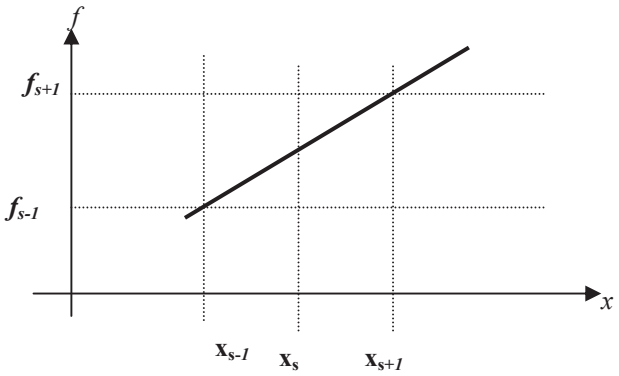
$$f'_s = \frac{1}{2h}\left[f_{s+1} - f_{s-1}\right] - \frac{h^2}{6} f'''_s - \cdots$$

$$\boxed{f'_s = \frac{\left[f_{s+1} - f_{s-1}\right]}{2h} + O\left(h^2\right)}$$

Si se truncan los términos de orden $O(h^2)$, resulta

$$f'_s \cong \frac{1}{2h}\left[f_{s+1} - f_{s-1}\right] = \frac{f_{s+1} - f_{s-1}}{x_{s+1} - x_{s-1}}$$

Gráficamente se observa que el valor de la derivada primera en el punto central es la pendiente de la recta secante entre los puntos (x_{s-1}, f_{s-1}) y (x_{s+1}, f_{s+1}):



3 DERIVADAS SEGUNDAS

Considerando los desarrollos en serie de Taylor de la función para evaluar la función en las abscisas x_{s-1} y x_{s+1}:

$$f_{s-1} = f_s - h f'_s + \frac{h^2}{2} f''_s - \frac{h^3}{6} f'''_s + \frac{h^4}{24} f^{iv}_s - \cdots$$

$$f_{s+1} = f_s + h f'_s + \frac{h^2}{2} f''_s + \frac{h^3}{6} f'''_s + \frac{h^4}{24} f^{iv}_s + \cdots$$

y sumando miembro a miembro,

$$f_{s+1} + f_{s-1} = 2f_s + h^2 f''_s + \frac{h^4}{12} f'''_s + \cdots$$

es posible despejar :

$$f''_s = \frac{1}{h^2}\left[f_{s+1} - 2f_s + f_{s-1}\right] - \frac{h^2}{12} f'''_s \cdots$$

o bien,

$$\boxed{f_s'' = \frac{1}{h^2}\left[f_{s+1} - 2f_s + f_{s-1}\right] + O\left(h^2\right)}$$

$$f_s'' = \frac{1}{h}\left[\frac{f_{s+1} - f_s}{h} - \frac{f_s - f_{s-1}}{h}\right] + O\left(h^2\right)$$

Si se truncan los términos $O(h^2)$, la derivada segunda se puede aproximar como,

$$f_s'' \cong \frac{1}{h^2}\left[f_{s+1} - 2f_s + f_{s-1}\right]$$

O bien en términos de diferencias divididas,

$$f_s'' \cong \frac{f\left[x_{s+1} - x_s\right] - f\left[x_s - x_{s-1}\right]}{h}$$

4 DERIVADA TERCERA

Considerando los desarrollos en serie de Taylor de la función para

n = -2 $\quad f_{s-2} = f_s - 2h f_s' + 2h^2 f_s'' - \frac{4}{3}h^3 f_s''' + \frac{2}{3}h^4 f_s^{iv} + \cdots$

n = -1 $\quad f_{s-1} = f_s - h f_s' + \frac{h^2}{2} f_s'' - \frac{h^3}{6} f_s''' + \frac{h^4}{16} f_s^{iv} + \cdots$

n = +1 $\quad f_{s+1} = f_s + h f_s' + \frac{h^2}{2} f_s'' + \frac{h^3}{6} f_s''' + \frac{h^4}{16} f_s^{iv} + \cdots$

n = +2 $\quad f_{s+2} = f_s + 2h f_s' + 2h^2 f_s'' + \frac{4}{3}h^3 f_s''' + \frac{2}{3}h^4 f_s^{iv} + \cdots$

y combinándolos linealmente de la siguiente forma:

$$\begin{aligned} -f_{s-2} + 2f_{s-1} - 2f_{s+1} + f_{s+2} = \quad & -f_s + 2h f_s' - 2h^2 f_s'' + \frac{4}{3}h^3 f_s''' - \frac{2}{3}h^4 f_s^{iv} - \cdots \\ & + 2f_s - 2h f_s' + h^2 f_s'' - \frac{h^3}{3} f_s''' + \frac{h^4}{8} f_s^{iv} + \cdots \\ & - 2f_s - 2h f_s' - h^2 f_s'' - \frac{h^3}{3} f_s''' + \frac{h^4}{8} f_s^{iv} + \cdots \\ & f_s + 2h f_s' + 2h^2 f_s'' + \frac{4}{3}h^3 f_s''' + \frac{2}{3}h^4 f_s^{iv} + \cdots \end{aligned}$$

resulta

$$-f_{s-2} + 2f_{s-1} - 2f_{s+1} + f_{s+2} = \left(\frac{4}{3} - \frac{1}{3} - \frac{1}{3} + \frac{4}{3}\right)h^3 f_s''' + O\left(h^5 f_s^{v}\right) = 2h^3 f_s''' + O\left(h^5 f_s^{v}\right).$$

De donde se obtiene:

$$f_s''' = \frac{1}{2h^3}\left[-f_{s-2} + 2f_{s-1} - 2f_{s+1} + f_{s+2}\right] - \frac{1}{2h^3} O\left(h^5 f^{v}{}_s\right)$$

$$\boxed{f_s''' = \frac{1}{2h^3}\left[-f_{s-2} + 2f_{s-1} - 2f_{s+1} + f_{s+2}\right] - O\left(h^2\right).}$$

Es decir, si se trunca los términos de orden $O(h^2)$, se tiene aproximadamente,

$$f_s''' \cong \frac{1}{2h^3}\left[-f_{s-2} + 2f_{s-1} - 2f_{s+1} + f_{s+2}\right]$$

y reordenando términos queda:

$$f_s''' = \frac{1}{2h^3}\left[\left(f_{s+2} - f_{s+1}\right) - \left(f_{s+1} - f_{s-1}\right) + \left(f_{s-1} - f_{s-2}\right)\right]$$

$$f_s''' = \frac{1}{2h^3}\left[\left(f_{s+2} - f_{s+1}\right) - \left(f_{s+1} - f_s\right) - \left(f_s - f_{s-1}\right) + \left(f_{s-1} - f_{s-2}\right)\right]$$

$$f_s''' = \frac{1}{2h^2}\left[\frac{\left(f_{s+2} - f_{s+1}\right)}{h} - \frac{\left(f_{s+1} - f_s\right)}{h} - \frac{\left(f_s - f_{s-1}\right)}{h} + \frac{\left(f_{s-1} - f_{s-2}\right)}{h}\right]$$

$$f_s''' = \frac{1}{h}\left[\frac{\dfrac{\left(f_{s+2} - f_{s+1}\right)}{h} - \dfrac{\left(f_{s+1} - f_s\right)}{h}}{2h} - \frac{\dfrac{\left(f_s - f_{s-1}\right)}{h} - \dfrac{\left(f_{s-1} - f_{s-2}\right)}{h}}{2h}\right],$$

que es la diferencia dividida de tercer orden.

5 DERIVADA CUARTA

Considerando los desarrollos en serie de Taylor de la función para

$n = -2$ $\quad f_{s-2} = f_s - 2h f_s' + 2h^2 f_s'' - \frac{4}{3}h^3 f_s''' + \frac{2}{3}h^4 f_s^{iv} - \frac{4}{15}h^5 f_s^{v} + \cdots$

$n = -1$ $\quad f_{s-1} = f_s - h f_s' + \frac{h^2}{2} f_s'' - \frac{h^3}{6} f_s''' + \frac{h^4}{24} f_s^{iv} - \frac{h^5}{120} f_s^{v} + \cdots$

$n = 0$ $\quad f_s = f_s$

$n = +1$ $\quad f_{s+1} = f_s + h f_s' + \frac{h^2}{2} f_s'' + \frac{h^3}{6} f_s''' + \frac{h^4}{24} f_s^{iv} + \frac{h^5}{120} f_s^{v} + \cdots$

$n = +2$ $\quad f_{s+2} = f_s + 2h f_s' + 2h^2 f_s'' + \frac{4}{3}h^3 f_s''' + \frac{2}{3}h^4 f_s^{iv} + \frac{4}{15}h^5 f_s^{v} + \cdots$

Si se truncan las series en el término de cuarto orden se obtiene el siguiente sistema:

$$\begin{Bmatrix} f_{s-2} \\ f_{s-1} \\ f_s \\ f_{s+1} \\ f_{s+2} \end{Bmatrix} = \begin{bmatrix} 1 & -2h & 2h^2 & -\frac{4}{3}h^3 & \frac{2}{3}h^4 \\ 1 & -h & \frac{h^2}{2} & -\frac{h^3}{6} & \frac{h^4}{24} \\ 1 & 0 & 0 & 0 & 0 \\ 1 & h & \frac{h^2}{2} & \frac{h^3}{6} & \frac{h^4}{24} \\ 1 & 2h & 2h^2 & \frac{4}{3}h^3 & \frac{2}{3}h^4 \end{bmatrix} \begin{Bmatrix} f_s \\ f_s' \\ f_s'' \\ f_s''' \\ f_s^{'v} \end{Bmatrix}$$

La solución del sistema es:

$$f_s^{'v} = \frac{1}{h^4}\left[f_{s-2} - 4f_{s-1} + 6f_s - 4f_{s+1} + f_{s+2}\right]$$

$$f_s''' = \frac{1}{2h^3}\left[-f_{s-2} + 2f_{s-1} + 0f_s - 2f_{s+1} + f_{s+2}\right]$$

$$f_s'' = \frac{1}{h^2}\left[-\frac{1}{12}f_{s-2} + \frac{4}{3}f_{s-1} - \frac{5}{2}f_s + \frac{4}{3}f_{s+1} - \frac{1}{12}f_{s+2}\right]$$

$$f_s' = \frac{1}{h}\left[\frac{1}{12}f_{s-2} - \frac{2}{3}f_{s-1} + 0f_s + \frac{2}{3}f_{s+1} - \frac{1}{12}f_{s+2}\right]$$

5.1 Error

El término que aparece en las expresiones anteriores como $O(h^p)$, se conoce como error de truncamiento, ya que se obtiene al truncar la serie de Taylor. El orden de precisión de una derivación numérica viene dado por el exponente p de la potencia de h que aparece en el término del error de truncamiento.

Para obtener el orden del error de las expresiones obtenidas, se combinan linealmente los desarrollos en serie, según los coeficientes obtenidos.

Así para f_s^{iv} :

$$\begin{aligned} f_{s-2} - 4f_{s-1} + 6f_s - 4f_{s+1} + f_{s+2} = {} & f_s - 2hf_s' + 2h^2 f_s'' - \frac{4}{3}h^3 f_s''' + \frac{2}{3}h^4 f_s^{iv} - \frac{4}{15}h^5 f^{v}{}_s + O(h^6) \\ & - 4f_s + 4hf_s' - 2h^2 f_s'' + \frac{4}{6}h^3 f_s''' - \frac{4}{24}h^4 f_s^{iv} + \frac{4}{120}h^5 f_s^{v} + O(h^6) \\ & + 6f_s \\ & - 4f_s - 4hf_s' - 2h^2 f_s'' - \frac{4}{6}h^3 f_s''' - \frac{4}{24}h^4 f_s^{iv} - \frac{4}{120}h^5 f_s^{v} + O(h^6) \\ & f_s + 2hf_s' + 2h^2 f_s'' + \frac{4}{3}h^3 f_s''' + \frac{2}{3}h^4 f_s^{iv} + \frac{4}{15}h^5 f^{v}{}_s + O(h^6) \end{aligned}$$

$$f_{s-2} - 4f_{s-1} + 6f_s - 4f_{s+1} + f_{s+2} = f_s(1-4+6-4+1) + hf_s'(-2+4-4+2) + h^2 f_s''(2-2-2+2)$$

$$+h^3 f_s'''\left(-\frac{4}{3}+\frac{4}{6}-\frac{4}{6}+\frac{4}{3}\right)+h^4 f_s^{'v}\left(\frac{2}{3}-\frac{4}{24}-\frac{4}{24}+\frac{2}{3}\right)$$
$$+h^5 f_s^{v}\left(-\frac{4}{15}+\frac{4}{120}-\frac{4}{120}+\frac{4}{15}\right)+O(h^6)$$

$$f_{s-2}-4f_{s-1}+6f_s-4f_{s+1}+f_{s+2}=h^4 f_s^{'v}+O\left(h^6\right),$$

de donde

$$\boxed{f_s^{'v}=\frac{1}{h^4}\left[f_{s-2}-4f_{s-1}+6f_s-4f_{s+1}+f_{s+2}\right]+O\left(h^2\right)}$$

El orden del error es $\mathbf{e_{(4)}} = O(h^2)$.

Calculando otras combinaciones lineales se puede obtener $\mathbf{e_{(3)}} = O(h^2)$, $\mathbf{e_{(2)}} = O(h^4)$ y $\mathbf{e_{(1)}} = O(h^4)$ para las expresiones obtenidas con los desarrollos en serie de Taylor hasta orden 5.

Observación: nótese que si se calcula la derivada primera de una función en un punto hacia delante o hacia atrás (a partir de dos puntos datos), se tiene un error del orden de $O(h)$; si se hace el cálculo de dicha derivada mediante la fórmula central (a partir de tres puntos), se tiene un error del orden de $O(h^2)$; en cambio si se lo hace utilizando esta última fórmula (a partir de cinco puntos), se comete un error del orden de $O(h^4)$. Algo similar ocurre con la derivada segunda.

6 DERIVADA PRIMERA ASIMÉTRICA

Se pretende obtener una fórmula de derivada primera hacia delante que tenga orden de error superior a uno. Para ello se consideran tres puntos equidistantes, Xs; Xs+1 y Xs+2, y se platea que la derivada primera sea una combinación lineal de los valores de la función, cuya derivada se pretende calcular, en esas abscisas. Esto es:

$$f_s^{'}=\left[\alpha\cdot f_s+\beta\cdot f_{s+1}+\gamma\cdot f_{s+2}\right]$$

Se considera los desarrollos en serie de Taylor de la función f(x) en dichas abscisas,

$n = +1$ $\qquad f_{s+1}=f_s+h\,f_s'+\frac{h^2}{2}f_s''+\frac{h^3}{6}f_s'''+\frac{h^4}{24}f_s^{iv}+\frac{h^5}{120}f_s^{v}+\cdots$

$n = +2$ $\qquad f_{s+2}=f_s+2h\,f_s'+2h^2\,f_s''+\frac{4}{3}h^3\,f_s'''+\frac{2}{3}h^4 f_s^{iv}+\frac{4}{15}h^5 f_s^{v}+\cdots$

Al reemplazar estas series en la combinación lineal propuesta y agrupando términos se obtiene una nueva serie para la derivada primera en Xs

$$f_s^{'}=\left[\alpha+\beta+\gamma\right]\cdot f+f_s'\cdot\left[\beta\cdot h+\gamma\cdot 2h\right]+\frac{h^2}{2}f_s''\cdot\left[\beta+4\cdot\gamma\right]+\frac{h^3}{6}f_s'''\cdot\left[\beta+8\cdot\gamma\right]+\frac{h^4}{24}f_s^{iv}\left[\beta+16\cdot\gamma\right]+\cdot\cdot$$

Para que la nueva serie obtenida sea igual a la derivada primera en Xs, se debe cumplir que

$$0 = [\alpha + \beta + \gamma]$$

$$1 = [\beta \cdot h + \gamma \cdot 2h]$$

Siendo el error de truncamiento, el término más importante de la parte truncada evaluado en un punto cercano ξ.

$$Er = \frac{h^2}{2} f''_{\xi} \cdot [\beta + 4 \cdot \gamma]$$

De la solución del sistema de ecuaciones lineales de dos ecuaciones con tres incógnitas se tiene:

$$\alpha = [-1/h + \gamma]$$

$$\beta = [1/h - 2 \cdot \gamma]$$

De modo que la derivada primera hacia delante es

$$f'_s = [[-1/h + \gamma] \cdot f_s + [1/h - 2 \cdot \gamma] \cdot f_{s+1} + \gamma \cdot f_{s+2}]$$

Con un error de truncamiento local

$$Er = \frac{h^2}{2} f''_{\xi} \cdot [1/h + 2 \cdot \gamma]$$

Esto es válido para todo valor de γ. En particular cuando γ es cero, se recupera la derivada primera adelante y su error de truncamiento local. Mientras γ sea un coeficiente no nulo, el error de truncamiento local depende del valor de γ y de h linealmente.

Es de destacar el caso en que se adopta $\gamma = -1/(2h)$; en cuyo caso la derivada primera hacia delante es

$$f'_s = \{[-3/(2h)] \cdot f_s + [2/h] \cdot f_{s+1} + [-1/(2h)] f_{s+2}\}$$

y el error de trocamiento resulta nulo, y se debe considerar el siguiente término de la serie

$$f'_s = [\alpha + \beta + \gamma] \cdot f + f'_s \cdot [\beta \cdot h + \gamma \cdot 2h] + \frac{h^2}{2} f''_s \cdot [\beta + 4 \cdot \gamma] + \frac{h^3}{6} f'''_s \cdot [\beta + 8 \cdot \gamma] + \frac{h^4}{24} f^{iv}_s [\beta + 16 \cdot \gamma] + \cdot\cdot$$

Es decir:, el error de truncamiento resulta:

$$Er = \frac{h^3}{6} f'''_s \cdot [\beta + 8 \cdot \gamma] = \frac{h^3}{6} f'''_s \cdot [2/h] = \frac{h^2}{3} f'''_s \cdot$$

Resulta así que la derivada primera hacia adelante considerando tres puntos es exacta hasta polinomios de grado 2 y el orden de l error de truncamiento local es de h^2.

Ejercicio: Demostrar siguiendo el procedimiento anterior que la derivada primera hacia atrás al considerar tres puntos, resulta:

$$f'_s = \{[3/(2h)] \cdot f_s + [-2/h] \cdot f_{s-1} + [1/(2h)] f_{s-2}\}$$

Y su error de truncamiento es:

$$Er = -\frac{h^2}{3} f'''_s \cdot$$

7 APLICACIÓN DE DERIVADA NUMÉRICA EN LA SOLUCIÓN DE ECUACIONES DIFERENCIALES ORDINARIAS CON VALORES DE CONTORNO

Es posible usar las reglas de derivación numérica en la obtención de soluciones aproximadas de ecuaciones diferenciales ordinarias, en particular con valores de contorno. Se suele referir a esta forma de solución aproximada como el Método de Diferencias Finitas.

Es posible plantear el método mediante un ejemplo simple.

Se busca *u(x)* solución de

$$-\frac{d^2u(x)}{dx^2}+u(x)-x=0 \quad en\ \ \Omega=\{x\,\varepsilon\, R:0\le x\le 1\}$$
$$u(0)=0$$
$$u(1)=0$$

La solución exacta de esta ecuación diferencial es

$$u(x)=x-\frac{senh(x)}{senh(1)}$$

En vez de encontrar la solución en cada uno y todos los puntos del dominio Ω, se plantea encontrar la solución en forma aproximada en sólo algunos puntos elegidos del dominio y equidistantes identificados con su abscisa X_k. Para ello se divide el dominio Ω en N segmentos iguales y así quedan definidos N+1 puntos que incluyen a los bordes del dominio.

Se busca $U(X_k)$ con k=0,N; función discreta que es una aproximación de la función continua *u(x)*. En cada punto se postula la existencia de un valor aproximado de la solución buscada $U(X_k)=U_k$ con k que varía desde 0 hasta N.

En cada uno de los X_k se puede plantear la ecuación diferencial a resolver pero con una aproximación de la derivada segunda en forma de derivada numérica considerando la función discreta U_k. Así se puede escribir:

$$-\frac{1}{\Delta x^2}\left[U_{k-1}-2\cdot U_k+U_{k+1}\right]+U_k-X_k=0 \quad en\ X_k \qquad con\ \ k=1,(N-1)$$

siendo Δx=1/N la distancia entre los puntos. Es una ecuación algebraica cuyas incógnitas son las U_k. De estas ecuaciones se pueden plantear tantas como puntos interiores; es decir N-1 ecuaciones y se tienen n+1 incógnitas. Además se tienen las dos ecuaciones correspondientes a las Condiciones de Contorno, que agregan dos ecuaciones más. Así se tienen N+1 ecuaciones con N+1 incógnitas.

Caso N=2

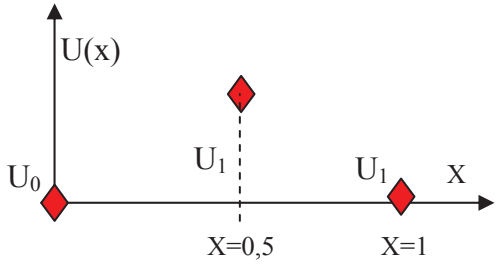
$U_0=0 \quad en\ X_0$

$$-\frac{1}{0{,}5^2}\left[U_0-2\cdot U_1+U_2\right]+U_1-X_1=0 \quad en\ X_1$$

$U_1=0 \quad en\ X_1$

O bien

$$9\cdot U_1-0{,}5=0$$



La solución aproximada es

$$U_1=1/18=0{,}055555$$

Así el error respecto de la solución exacta en ese punto es

$$E_2 = \left| \frac{1}{18} - (0{,}5 - \frac{senh(0{,}5)}{senh(1)}) \right| = 0{,}001035002$$

$$E(abs)_2 = \left| \frac{1}{18} - (0{,}5 - \frac{senh(0{,}5)}{senh(1)}) \right| / (0{,}5 - \frac{senh(0{,}5)}{senh(1)}) = 1{,}83\%$$

Caso N=4

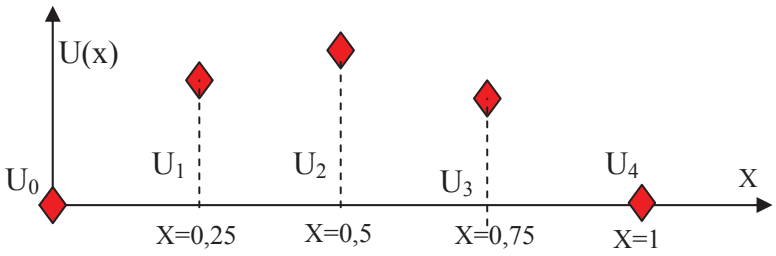
$U_0 = 0 \quad en\ X_0$

$$-\frac{1}{0{,}5^2}\left[U_0 - 2 \cdot U_1 + U_2\right] + U_1 - X_1 = 0 \quad en\ X_1 = 0{,}25$$

$$-\frac{1}{0{,}5^2}\left[U_1 - 2 \cdot U_2 + U_3\right] + U_2 - X_2 = 0 \quad en\ X_2 = 0{,}5$$

$$-\frac{1}{0{,}5^2}\left[U_2 - 2 \cdot U_3 + U_4\right] + U_3 - X_3 = 0 \quad en\ X_3 = 0{,}75$$

$U_4 = 0 \quad en\ X_4$



O bien

$$\begin{bmatrix} 33 & -16 & 0 \\ -16 & 33 & -16 \\ 0 & -16 & 33 \end{bmatrix} \cdot \begin{Bmatrix} U_1 \\ U_2 \\ U_3 \end{Bmatrix} = \begin{Bmatrix} 0{,}25 \\ 0{,}5 \\ 0{,}75 \end{Bmatrix}$$

La solución aproximada es

$$\begin{Bmatrix} U_1 \\ U_2 \\ U_3 \end{Bmatrix} = \begin{Bmatrix} 0{,}03488525 \\ 0{,}05632582 \\ 0{,}05003676 \end{Bmatrix}$$

Así el error respecto de la solución exacta en ese punto es

$$E_4 = \left| U_2 - (0{,}5 - \frac{senh(0{,}5)}{senh(1)}) \right| = 0{,}000264735$$

$$E(abs)_4 = \left| U_2 - (0{,}5 - \frac{senh(0{,}5)}{senh(1)}) \right| / (0{,}5 - \frac{senh(0{,}5)}{senh(1)}) = 0{,}47\%$$

Caso N=8

$U_0 = 0 \quad en\ X_0 = 0$

$$-\frac{1}{(1/8)^2}\left[U_0 - 2\cdot U_1 + U_2\right] + U_1 - X_1 = 0 \quad en\ X_1 = 1/8$$

$$-\frac{1}{(1/8)^2}\left[U_1 - 2\cdot U_2 + U_3\right] + U_2 - X_2 = 0 \quad en\ X_2 = 2/8$$

$$-\frac{1}{(1/8)^2}\left[U_2 - 2\cdot U_3 + U_4\right] + U_3 - X_3 = 0 \quad en\ X_3 = 3/8$$

.....

$U_8 = 0 \quad en\ X_8 = 1$

O bien

$$\begin{bmatrix} 129 & -64 & 0 & 0 & 0 & 0 & 0 \\ -64 & 129 & -64 & 0 & 0 & 0 & 0 \\ 0 & -64 & 129 & -64 & 0 & 0 & 0 \\ 0 & 0 & -64 & 129 & -64 & 0 & 0 \\ 0 & 0 & 0 & -64 & 129 & -64 & 0 \\ 0 & 0 & 0 & 0 & -64 & 129 & -64 \\ 0 & 0 & 0 & 0 & 0 & -64 & 129 \end{bmatrix} \cdot \begin{Bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \\ U_5 \\ U_6 \\ U_7 \end{Bmatrix} = \begin{Bmatrix} 1/8 \\ 2/8 \\ 3/8 \\ 4/8 \\ 5/8 \\ 6/8 \\ 7/8 \end{Bmatrix}$$

La solución aproximada es

$\mathbf{U}^T = \{0{,}0183367;\ 0{,}03500678;\ 0{,}04831759;\ 0{,}05652399;\ 0{,}05780107;\ 0{,}05021568;\ 0{,}03169615\}$

Así el error respecto de la solución exacta en ese punto es

$$E_8 = \left| U_4 - (0{,}5 - \frac{senh(0{,}5)}{senh(1)}) \right| = 6{,}65711\text{E-}05$$

$$E(abs)_8 = \left| U_4 - (0{,}5 - \frac{senh(0{,}5)}{senh(1)}) \right| / (0{,}5 - \frac{senh(0{,}5)}{senh(1)}) = 0{,}12\%$$

Evaluación del Error

Al considerar el error para distintos niveles de *disicretización*; es decir, distinto número de segmentos en que se divide el dominio, se tiene

$$E_N = \left| U_{N/2} - (0{,}5 - \frac{senh(0{,}5)}{senh(1)}) \right| \text{ y } E(abs)_N = \left| U_{N/2} - (0{,}5 - \frac{senh(0{,}5)}{senh(1)}) \right| / (0{,}5 - \frac{senh(0{,}5)}{senh(1)})$$

Cuyas evaluaciones se presentan en la siguiente Tabla

N	Δx	$U_{aprox}(0{,}5)$	E_N	$E(abs)_N$
2	0,5	0,05555556	0,001035002	1,83%
4	0,25	0,05632582	0,000264735	0,47%
8	0,125	0,05652399	6,65711E-05	0,12%
16	0,0625	0,05657389	1,66672E-05	0,03%

Si se asume una relación exponencial entre $E(abs)_N$ y Δx, la aproximación por mínimos cuadrados da:

$$E(abs)_N = C \cdot \Delta x^P = e^{-2{,}6168} \cdot \Delta x^{1{,}9861}$$

Que indica una relación del orden de $\Delta x^{1{,}9861} \cong \Delta x^2$ que es el error de truncamiento local de la aproximación de derivada segunda utilizado.

SOLUCION NUMÉRICA DE

ECUACIONES DIFERENCIALES ORDINARIAS.

1 INTRODUCCIÓN

Las ecuaciones diferenciales aparecen con frecuencia en modelos matemáticos de diversas disciplinas: biología, ecología, economía, administración, ingeniería, meteorología, oceanografía, física y sociología.

¿Qué es una ecuación diferencial?

Es una ecuación en la que aparecen funciones, sus derivadas, una o más variables independientes y una o más variables dependientes.

Las ecuaciones diferenciales se dividen en dos grupos:

Ecuaciones Diferenciales Ordinarias: (EDO) en las que aparece sólo una variable independiente ´x´.

Ecuaciones Diferenciales Parciales: (EDP) en las que aparecen más de una variable independiente.

Centramos nuestra atención en las EDO. El *objetivo* es determinar la función $y(x)$ que satisface la EDO. Por ejemplo:

a) $y''(x)=f(x)$

b) $y'(x)=-K\,y(x)$, con K una constante dada.

c) $(y''(x))^3 - 3y'(x)+x\,y = x$

Donde ( )´ indica derivada de ( ) respectoo a la variable independiente x.

¿Qué es el orden de una ecuación diferencial?

Es el entero igual al número máximo de veces que se deriva la variable dependiente de la ecuación.

En los ejemplos anteriores: a) es de segundo orden; b) es de primer orden; c) es de segundo orden.

Nos concentraremos en el estudio de *EDO de primer orden.*

$$y'=f(x,y)$$

siendo $f(x,y)$ una función conocida. La solución a esta EDO será una función tal que sustituida en la EDO la reduce a la identidad. La solución tendrá una constante arbitraria por lo que necesitaremos de una condición adicional para determinarla: Es decir, la solución de una EDO es una familia de curvas. Al especificar una condición inicial, estamos determinando cuál de esas curvas es la solución a nuestro problema.

Nos interesa resolver una *EDO de primer orden con una condición inicial:*

$$y'=f(x,y)$$
$$y(x_0)=y_0$$

¿Por qué surge la necesidad de los métodos numéricos para resolver EDO?

Existen soluciones analíticas para este tipo de EDO sólo para casos especiales de la función $f(x,y)$. Estas EDO especiales se llaman: de variables separables, exactas, de Bernoulli, homogéneas, etc. Pero la EDO de primer orden que puede llegar a nuestras manos para su resolución, puede no caer en ninguno de esos casos especiales de EDO: Con frecuencia, los

problemas de la práctica, o bien no pueden resolverse por los métodos clásicos, o bien la solución es tan difícil de obtener o tan laboriosa de evaluar que no vale la pena el esfuerzo. En un gran número de aplicaciones prácticas algunos de los coeficientes o funciones de una EDO son fuertemente no lineales, o están dados por medio de un conjunto tabulado de valores experimentales, lo que elimina las posibilidades de obtener una solución clásica analítica.

Entonces, por muchas razones, estamos obligados a buscar métodos de solución que se apliquen en los casos en los que los métodos clásicos no son útiles. Los métodos que consideraremos se generalizan fácilmente para resolver sistemas de EDO simultáneas de primer orden. Además, EDO de orden superior se pueden reducir fácilmente a un sistema de EDO simultáneas de primer orden.

Solución de una EDO de primer orden:

Sea la EDO de primer orden con condición inicial:

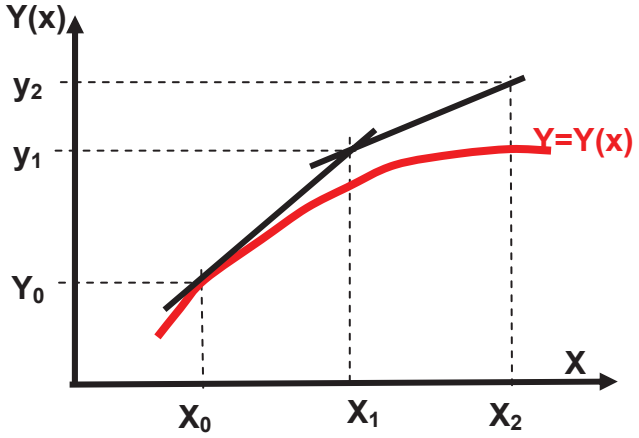
$$y' = f(x,y) \qquad (1)$$
$$y(x_0) = y_0$$

La función $f(x,y)$ es continua en un dominio D del plano (x,y). El punto (x_0, y_0) pertenece a D. La función $Y(x)$ es solución de (1) en un intervalo [a,b] si para todo x perteneciente a [a,b] se verifica:

- ✓ $(x, Y(x)) \in D$
- ✓ $Y(x_0) = Y_0$
- ✓ $\exists\ Y'(x)$ y se verifica que $Y'(x) = f(x, Y(x))$

¿Cómo se obtiene en general una solución numérica?

Buscamos una solución $Y(x)$. ¿Qué datos tenemos? : $Y(x_0) = Y_0$ y la pendiente de la curva en cualquier punto como función $f(x,y)$ de x e y. Comenzamos con el punto conocido (x_0, Y_0). Calculamos la pendiente de la curva en ese punto $(y' = f(x,y))$. Avanzamos sobre el eje x una cierta distancia. Si al incremento en x lo llamamos h, obtenemos un nuevo punto $x_1 = x_0 + h$ y con la pendiente de la recta tangente ya obtenida se determina $Y(x_1) = Y_1$. Continuando de esta manera obtenemos una secuencia de líneas rectas que aproximan a la curva verdadera que es la solución.



Lo que hacemos entonces es *discretizar* la variable x en una sucesión de puntos $\{x_m\}$. Podemos permitir una longitud de paso variable, es decir:

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

Pero en la mayoría de los problemas se considera h constante. O sea,

$$x_m = x_0 + m.h \qquad \text{con } m = 0,1, ..., n$$

La función aproximante $Y(x)$ se obtiene sólo para los puntos $\{x_m\}$. Si necesitamos obtener Y para otros valores de x, podemos usar interpolación.

Si el problema consiste en determinar $y(x)$ en $x=b$, podemos elegir n como el número de subintervalos en $[x_0 , b]$. Por lo tanto,

$$h = (b - x_0)/n$$

2 CLASIFICACIÓN DE LOS MÉTODOS.

Existen dos categorías básicas de métodos numéricos para la resolución de EDO de primer orden:

I. *Métodos de un paso:*

- ✓ Son métodos tales que para aproximar la solución en el punto de abscisa x_m usan datos sólo del punto anterior (x_{m-1} , Y_{m-1}).
- ✓ Son métodos directos, es decir, la solución en un punto no se itera.
- ✓ Tiene la desventaja de que es difícil estimar el error.
- ✓ Métodos: Desarrollo en Serie de Taylor; Métodos de Runge-Kutta.
- ✓ La forma general de estos métodos es:

$$Y_{m+1} = Y_m + h\ \Phi (x_m , Y_m, h, f)$$

La función Φ está relacionada con los conceptos de convergencia y estabilidad del método.

II. *Métodos multipaso:*

- ✓ Son métodos tales que para aproximar la solución en el punto de abscisa x_m usan información de varios puntos anteriores.
- ✓ Son métodos que requieren iteración de la solución para llegar a un valor suficientemente preciso.
- ✓ Es posible obtener una estimación del error.
- ✓ Requieren menos evaluaciones de la función.
- ✓ La mayoría de los métodos de este tipo se llaman Predictor - Corrector.
- ✓ La forma general de estos métodos es:

$$Y_{m+1} = \sum_{j=0}^{p} a_j Y_{m-j} + h \sum_{j=1}^{p} b_j f(x_{m-j}, Y_{m-j})$$

los que a su vez se clasifican en :

- ♦ *Métodos explícitos*, si $b_{-1} = 0$
- ♦ *Métodos implícitos*, si $b_{-1} \neq 0$

La determinación de los coeficientes a_j y b_j no es arbitraria, sino que está asociada a los conceptos de convergencia y estabilidad del método.

3 TIPOS DE ERRORES EN LOS MÉTODOS NUMÉRICOS PARA LA SOLUCIÓN DE EDO.

En los métodos numéricos para resolver EDO podemos tener los siguientes tipos de errores:

1. *Error de Redondeo Local:*

   Motivado por realizar operaciones con un número finito de cifras significativas. Es independiente del paso h.

2. *Error por Truncado Local:*

   Es debido al método y depende de h. Generalmente el error por redondeo local es menor que el error por truncado local.

3. *Error por Propagación:*

   El error pasa de una etapa a la siguiente y el error final puede ser apreciable si el método no es estable.

4 SOLUCIÓN EN SERIE DE TAYLOR.

Empezamos nuestro estudio con un método que teóricamente suministra una solución para cualquier EDO, pero que sin embargo tiene escaso valor computacional práctico. Su importancia estriba en que da una base para evaluar y comparar los métodos que sí son valiosos en la práctica.

Un planteo razonable para resolver nuestra EDO

$$y' = f(x,y) \qquad (1)$$
$$y(x_0)=y_0$$

sería desarrollar $y(x)$ en una Serie de Taylor con centro en x_0 y luego evaluar la serie en x_1 para así obtener $y(x_1) = y_1$. Repitiendo este proceso podemos movernos a x_2, x_3, etc. El desarrollo en Serie de Taylor de $y(x)$ con centro en x_0, se puede escribir en la forma:

$$y(x) = y_0 + y'_0 (x- x_0) + y_0''/2 (x-x_0)^2 + y_0'''/6 (x-x_0)^3 + \ldots \qquad (2)$$

$$y(x) = \sum_{k=0}^{\infty} \frac{1}{k!}(x - x_0)^k y_0^{(k)}$$

donde y_0^k es la k-ésima derivada de $y(x)$ evaluada en $x = x_0$.

Si $x_1 = x_0 + h$, entonces,

$$y(x_1) = y(x_0 + h) = \sum_{k=0}^{\infty} \frac{1}{k!}(h)^k y_0^{(k)} \qquad (3)$$

En la práctica sólo se utilizan los $(n+1)$ primeros términos del Desarrolllo en Serie de Taylor y el método recibe entonces el nombre de *Método de Taylor de orden n*

Supongamos que hemos encontrado una solución aproximada en $(n+1)$ puntos a lo largo del eje x: x_0 , x_1 , ..., x_n Los valores sucesivos de x están todos a una distancia h del precedente. Es decir,

$$x_m = x_0 + m \,.\, h \tag{4}$$

siendo h la longitud de paso. Podemos aproximar la solución en el siguiente punto x_{m+1} sustituyendo x_{m+1} por x en (2) y teniendo en cuenta que $x_{m+1} = x_m + h$ resulta:

$$Y_{m+1} = Y(x_m + h) = Y_m + h\, Y_m' + h^2/2\, Y_m'' + h^3/6\, Y_m''' + \ldots \tag{5}$$

La aproximación será mejor cuantos más términos tomemos del desarrollo en serie. Es necesario evaluar las derivadas. Así se tiene de (1) que:

$$Y_m' = f(x_m,\, Y_m) \tag{6}$$

Derivando (1) con respecto a x obtenemos:

$$y'' = \frac{df}{dx} = \frac{\partial f}{\partial x} + \frac{\partial f}{\partial y}\frac{\partial y}{\partial x} = \frac{\partial f}{\partial x} + \frac{\partial f}{\partial y} f$$

Si indicamos que

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

Es decir, las letras subíndices denotan derivadas parciales con respecto a la variable indicada en el subíndice. Escribimos la derivada segunda como:

$$y'' = f_x + f f_y \tag{7}$$

Evaluemos la derivada tercera:

$$y''' = f'' = f_{xx} + f_{xy} y' + (f_x + f_y y')f_y + f(f_{yx} + f_{yy} y') =$$

$$= f_{xx} + f_{xy} f + f_x f_y (f_y)^2 f + f f_{yx} + f^2 f_{yy}$$

$$y''' = f_{xx} + 2 f_{xy} f + f_x f_y + f(f_y)^2 + f^2 f_{yy} \tag{8}$$

Sustituyendo (6), (7) y (8) (evaluadas en (x_m, y_m) en (5) se tiene:

$$Y_{m+1} = Y_m + hf + \frac{h^2}{2}(f_x + f f_y) + \vartheta(h^3) \tag{9}$$

$\vartheta(h^3)$ se lee: "del orden de h^3" y significa que todos los términos subsecuentes contienen h elevada a la tercera potencia o superiores.

Esta es otra forma de decir que si usáramos la fórmula (9) sin el término $\vartheta(h^3)$ el *error por truncamiento* sería aproximadamente Kh^3 en que K es alguna constante.

La expresión del error de truncamiento asociada con la utilización de la expresión (5) se puede obtener al evaluar el primer término omitido en la Serie de Taylor, es decir,

$$ET = \frac{h^{n+1}}{(n+1)!} f^{(n)}\Big|_{x=\eta} = \frac{h^{n+1}}{(n+1)!} y^{(n+1)}(\eta) \tag{10}$$

donde η es algún punto entre x_j y x_{j+1}. Este error (con signo) se llamará *error de truncamiento local o de un paso.*

La solución en Serie de Taylor se clasifica como método de un paso porque para evaluar y_{m+1}

se requiere sólo información en (x_m, y_m).

¿Cuál es la dificultad práctica del método?

Que puede ser difícil, o en algunos casos imposible, encontrar f_x y f_y. Y vemos que las expresiones para evaluar las derivadas y'', y''', etc., se complican conforme aumenta el orden.

Por lo tanto, el método es generalmente impráctico desde el punto de vista computacional. Pero ahora consideraremos métodos prácticos y tendremos una base para juzgarlos: ¿hasta qué punto coinciden con el desarrollo en Serie de Taylor?

Una manera común de clasificar y comparar métodos es dar su *orden de precisión.* El orden de un método describe la expresión del error local de truncamiento:

> Si el error local de truncamiento es proporcional a h^{n+1},
>
> entonces decimos que el método es de *orden n*.

5 MÉTODOS DE RUNGE-KUTTA.

5.1 CARACTERISTICAS GENERALES.

Los métodos de Runge-Kutta tienen tres propiedades fundamentales:

- Son métodos de un paso.
- Son métodos que coinciden con la Serie de Taylor hasta los términos de orden h^p, en que p es distinto para cada uno de los métodos y se denomina el *orden del método*.
- Son métodos que no requieren la evaluación de ninguna derivada de $f(x, y)$, sino únicamente de la función f en sí.

Esta última propiedad es la que hace que estos métodos resulten más prácticos que el desarrollo en Serie de Taylor. El precio que pagamos por no tener que evaluar las derivadas, es que debemos evaluar $f(x, y)$ para más de un valor de x e y.

Comenzamos por el método más simple dentro de los métodos de Runge-Kutta, que suministra un punto inicial necesario para otras presentaciones, aunque de poca precisión. Posteriormente, se desarrollarán variantes que disminuyen los errores.

5.2

METODO DE EULER.

Es uno de los métodos más antiguos y mejor conocidos de integración numérica de ecuaciones diferenciales. Fue ideado por Euler hace más de 200 años. Es un método fácil de entender y de usar, pero no es tan preciso como los otros métodos que estudiaremos. Supongamos que para la abscisa $x = x_m$ conocemos la ordenada Y_m. Podemos entonces evaluar la pendiente de la recta tangente a la curva solución en dicho punto:

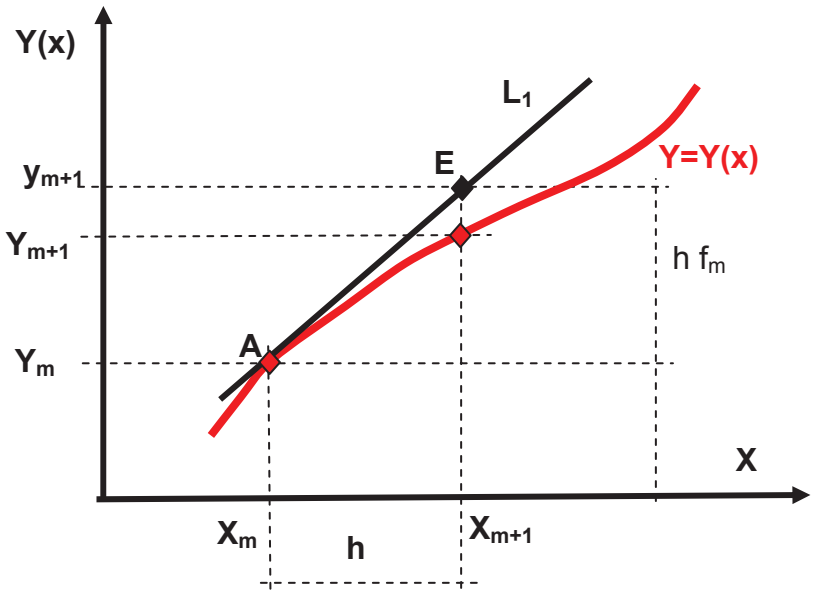
$$Y_m' = f(x_m, Y_m) \tag{11}$$

En la figura, la curva es la solución exacta que se desconoce y la recta L_1 es la recta tangente a la curva en el punto (x_m, Y_m). Hacemos coincidir Y_m con la solución exacta $y=y(x)$, pero en la práctica esto no ocurre, sino que Y_m es en general una aproximación.

La ecuación de la recta L_1 es:

$$Y = Y_m + Y_m'(x - x_m) \tag{12}$$

donde $Y_m' = f(x_m, Y_m)$.



Evaluamos el valor de la ordenada en $x_{m+1} = x_m + h$:

$$Y_{m+1} = Y_m + hf(x_m, Y_m) \tag{13}$$

Esta expresión coincide con el desarrollo en Serie de Taylor hasta el término en h . por lo tanto, el *error de truncamiento local* es:

$$ET_L = \frac{h^2}{2!} y''(\eta) \quad ; \quad \eta \in [x_m, x_{m+1}] \tag{14}$$

Se trata entonces de un Método de Runge-Kutta de Primer Orden.

Además de tener un error de truncamiento local relativamente grande, el Método de Euler es a menudo *inestable*. Es decir, un error pequeño (por redondeo, por truncamiento o inherente) se amplifica conforme aumenta el valor de x. Sólo para valores de h tendiendo a cero la precisión del método es satisfactoria.

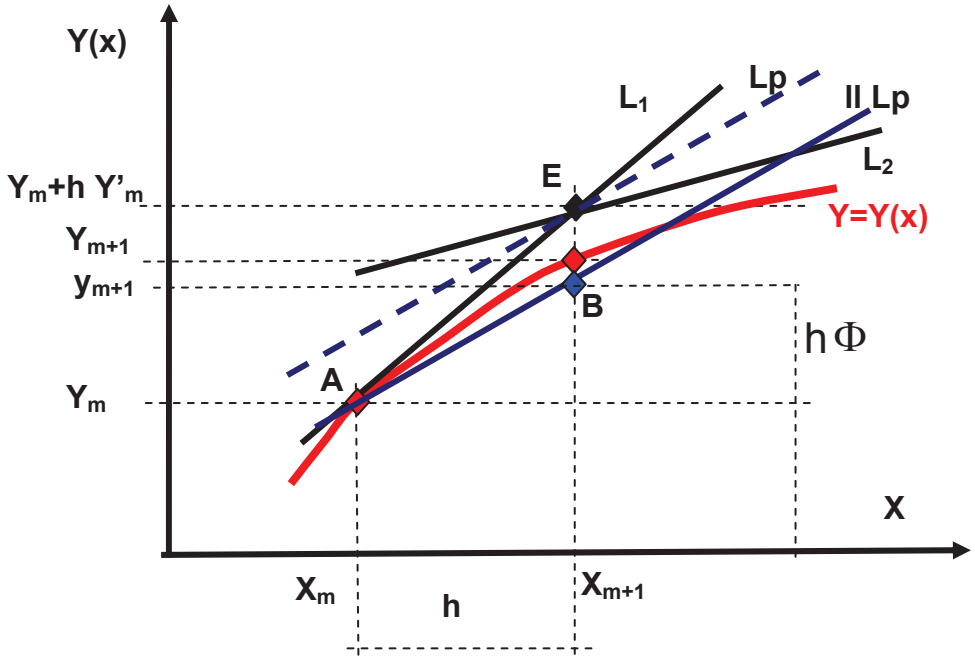
El Método de Euler usa sólo la pendiente de la recta tangente a la curva solución en el punto

(x_m, Y_m) para calcular Y_{m+1}. El método puede mejorarse de muchas maneras. Estudiaremos dos de ellas: el método Mejorado de Euler y el Método Modificado de Euler. Veremos luego que son dos métodos de una familia de métodos de Runge-Kutta de segundo orden.

5.3 METODO DE EULER MEJORADO.

En el Método Mejorado de Euler se trabaja con un promedio de pendientes. Veámoslo geométricamente:

- Determinamos con el Método de Euler el punto E $(x_m + h, Y_m + hY'_m)$ de la recta L_1.
- Calculamos en ese punto E, la pendiente de la recta tangente a la curva (L_2).
- Promediamos las dos pendientes y obtenemos la recta a trazos $\bar{L}$.
- Dibujamos una línea L paralela a $\bar{L}$ y que pasa por (x_m, Y_m).
- El punto en que esta recta intersecta a la vertical por x_{m+1} es el punto buscado $B(x_{m+1}, Y_{m+1})$



Teniendo en cuenta que $y' = f(x,y)$ veamos cuáles son las pendientes que nos interesan:

Pendiente de L_1: $$Y'_m = f(x_m, Y_m)$$

Pendiente de L_2: $$Y'_{m+1} = Y'(x_m + h) = f(x_m + h, Y_m + hY'_m)$$

Pendiente de $\bar{L}$: $$\Phi(x_m, Y_m, h) = \frac{1}{2}\left[f(x_m, Y_m) + f(x_m + h, Y_m + hY'_m)\right] \qquad (15)$$

La ecuación de la recta L resulta:

$$Y = Y_m + (x - x_m)\Phi(x_m, Y_m, h)$$

Por lo tanto,

$$Y_{m+1} = Y_m + h\,\Phi(x_m, Y_m, h) \qquad (16)$$

donde $\Phi(x_m, Y_m, h)$ está dada por la ecuación (15).

¿Cuál es el orden de precisión de este método?

Para averiguarlo veamos hasta qué términos coincide con la serie de Taylor.

La expansión en serie de la función *f(x,y)* se puede escribir como:

$$f(x,y) = f(x_m, Y_m) + (x - x_m)\left.\frac{\partial f}{\partial x}\right|_{(x_m, Y_m)} + (y - Y_m)\left.\frac{\partial f}{\partial y}\right|_{(x_m, Y_m)} + \ldots\ldots \quad (17)$$

Sustituimos:

$$x = x_m + h$$
$$y = Y_m + hY_m' = Y_m + hf(x_m, Y_m)$$

y obtenemos:

$$f(x_m + h, Y_m + hY_m') = f(x_m, Y_m) + hf_x + hf(x_m, Y_m)f_y + \vartheta(h^2) \quad (18)$$

donde $f_x = \left.\frac{\partial f}{\partial x}\right|_{(x_m, Y_m)}$ y $f_y = \left.\frac{\partial f}{\partial y}\right|_{(x_m, Y_m)}$ son las derivadas parciales y están ambas evaluadas en (x_m, Y_m).

Sustituimos este resultado en la expresión de Φ (ecuación (15):

$$\Phi(x_m, Y_m, h) = \frac{1}{2}\left[f(x_m, Y_m) + f(x_m, Y_m) + hf_x + hf(x_m, Y_m)f_y + \vartheta(h^2)\right]$$

$$\Phi(x_m, Y_m, h) = f(x_m, Y_m) + \frac{h}{2}\left[f_x + ff_y\right] + \vartheta(h^2) \quad (19)$$

Ahora reemplazamos Φ, (ecuación (19)), en la ecuación (16) para poder comparar el desarrollo en Serie de Taylor. Sustituyendo se obtiene:

$$Y_{m+1} = Y_m + h\, f(x_m, Y_m) + \frac{h^2}{2}\left[f_x + ff_y\right] + \vartheta(h^3) \quad (20)$$

Esta expresión coincide con el desarrollo en Serie de Taylor hasta los términos en h^2, así que el Método Mejorado de Euler es un Método de Runge-Kutta de segundo orden. Observamos que este método exige evaluar f dos veces. Sin embargo para obtener el mismo orden de precisión, la Serie de Taylor exige tres evaluaciones de funciones (f, f_x, f_y).

5.4

METODO DE EULER MODIFICADO.

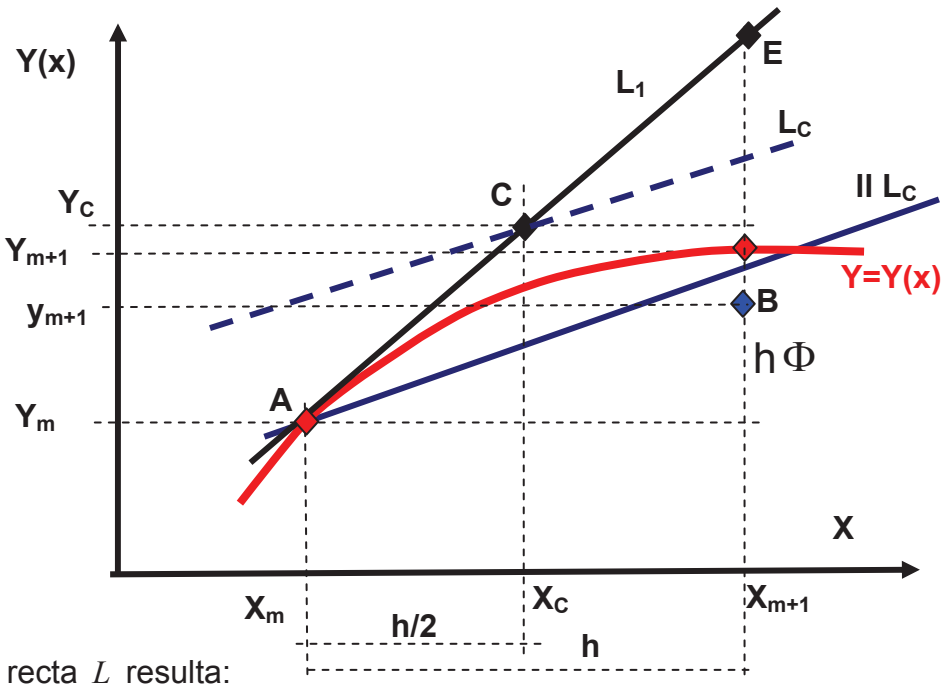
En lugar de promediar pendientes como en el método anterior, en este método modificado de Euler, se "promedian puntos". Veámoslo geométricamente:

- Determinamos con el Método de Euler el punto E $(x_m + h\,, Y_m + hY_m^{'})$ de la recta L_1.
- Ubicamos el punto C que pertenece a L_1: $C(x_m + \frac{h}{2}, Y_m + \frac{h}{2}Y_m^{'})$
- Calculamos la pendiente de la recta tangente a la curva en C:

$$\Phi(x_m, Y_m, h) = f\left[(x_m + \frac{h}{2}), (Y_m + \frac{h}{2}Y_m^{'})\right] \tag{21}$$

$$\text{con } Y_m^{'} = f(x_m, Y_m) \tag{22}$$

- $\overline{L}$ es la recta que pasa por C y tiene por pendiente a Φ (ecuación (21)).
- Dibujamos una línea L paralela a $\overline{L}$ y que pasa por (x_m, Y_m).
- El punto en que esta recta L intersecta a la vertical por x_{m+1} es el punto buscado $B(x_{m+1}, Y_{m+1})$.



La ecuación de la recta L resulta:

$$Y = Y_m + (x - x_m)\,\Phi(x_m, Y_m, h) \tag{23}$$

Por lo tanto,

$$Y_{m+1} = Y_m + h\,\Phi(x_m, Y_m, h) \tag{24}$$

donde $\Phi(x_m, Y_m, h)$ está dada por la ecuación (21).

Ejercicio: Demuestre que el Método Modificado de Euler coincide con el desarrollo en Serie de Taylor hasta los términos en h^2, y por lo tanto, es un Método de Runge-Kutta de segundo orden.

5.5 GENERALIZACION DE LOS METODOS DE RUNGE-KUTTA DE SEGUNDO ORDEN.

Hasta ahora tenemos dos métodos de Runge-Kutta de segundo orden. Sería interesante ver qué es lo que tienen en común y ver si pueden ser generalizados. Ambos están dados por una expresión de la forma:

$$Y_{m+1} = Y_m + h\,\Phi(x_m, Y_m, h) \tag{25}$$

con

$$\Phi(x_m,Y_m,h) = a_1 \cdot f(x_m,Y_m) + a_2 \cdot f\left[(x_m + b_1 h),(Y_m + b_2 h Y_m')\right] \tag{26}$$

siendo

$$Y_m' = f(x_m, Y_m) \tag{27}$$

Para el *Método Mejorado de Euler*:

$$\begin{aligned} a_1 &= a_2 = \frac{1}{2} \\ b_1 &= b_2 = 1 \end{aligned} \tag{28}$$

Para el *Método Modificdo de Euler*:

$$\begin{aligned} a_1 &= 0 \;;\; a_2 = 1 \\ b_1 &= b_2 = \frac{1}{2} \end{aligned} \tag{29}$$

Las ecuaciones (25), (26) y (27) constituyen una fórmula de tipo Runge-Kutta.

¿Cuáles son los valores permisibles de los parámetros a_1, a_2, b_1, b_2 *?*

Para obtener concordancia con el desarrollo en Serie de Taylor hasta términos en h^2 tenemos que plantear tres condiciones (que coincida el término en h y que coincidan los dos términos en h^2). Pero disponemos de cuatro parámetros para definir el Método de Runge-Kutta de segundo orden y sólo tres condiciones a satisfacer. Por lo tanto se pueden derivar muchas fórmulas diferentes de segundo orden.

Consideremos el desarrollo en serie de Taylor para f(x,y), con centro en (x_m, y_m):

$$f(x,y) = f(x_m, y_m) + (x - x_m)\frac{\partial f}{\partial x} + (y - y_m)\frac{\partial f}{\partial y} + O(h^2) \tag{30}$$

Escribimos:

$$\begin{vmatrix} x - x_m &= b_1\, h \\ y - y_m &= b_2\, h\, f \end{vmatrix} \tag{31}$$

Sustituimos en (30) queda:

$$f[(x_m + b_1 h),(Y_m + b_2 h Y_m')] = f(x_m, y_m) + b_1 h \cdot f_x + b_2 h \cdot f \cdot f_y + O(h^2) \tag{32}$$

Donde *f*, f_x , f_y están evaluadas en (x_m, y_m)

Entonces, al sustituir (32) en (26), la ecuación general (25) se puede expresar como:

$y_{m+1} = y_m + h\ \Phi(x_m, y_m, h) = y_m + h\left[a_1 f + a_2 f + h\left(a_2 b_1 f_x + a_2 b_2 f\ f_y\right)\right] + O(h^3)$

o bien agrupando términos,

$$y_{m+1} = y_m + h\,(a_1 + a_2)\,f + h^2\left(a_2 b_1 f_x + a_2 b_2 f f_y\right) + O(h^3) \qquad (33)$$

Comparamos la ecuación (33) con el desarrollo en serie de Taylor. Para que los términos coincidan es necesario que:

$$\begin{cases} a_1 + a_2 = 1 \\ a_2 \cdot b_1 = 1/2 \\ a_2 \cdot b_2 = 1/2 \end{cases} \qquad (34)$$

Tenemos 3 ecuaciones con 4 parámetros. Elegimos arbitrariamente uno de ellos.
Por ejemplo:

$$a_2 = \omega \neq 0$$

Entonces:

$$a_1 = 1 - \omega$$
$$b_1 = b_2 = \frac{1}{2\omega} \qquad (35)$$

Las ecuaciones (25), (26) y (27) que definen el método resultan:

$$y_{m+1} = y_m + h\left\{(1-\omega) f(x_m, y_m) + \omega f\left[\left(x_m + \frac{h}{2\omega}\right), \left(y_m + \frac{h}{2\omega} f(x_m, y_m)\right)\right]\right\} + O(h^3)$$

Esta es la expresión más general del método de Runge-Kutta de 2º orden. Tradicionalmente también se presenta a los métodos de Runge Kutta en la siguiente forma:

Dados x_m, y_m; y elegido el valor de h, se calcula:

$$\begin{aligned} k_1 &= h\ f(x_m, y_m) \\ k_2 &= h\ f\left[\left(x_m + \frac{h}{2\omega}\right), \left(y_m + \frac{1}{2\omega} k_1\right)\right] \\ y_{m+1} &= y_m + (1-\omega)\, k_1 + \omega\, k_2 \\ x_{m+1} &= x_m + h \end{aligned} \qquad (37)$$

Ejercicios:

- ¿Cuál es el valor de ω para el cual se obtiene el Método Mejorado de Euler?

  $\omega = 1/2$

- ¿cuál es el valor de ω para el cual se obtiene el Método Modificado de Euler?

  $\omega = 1$

5.6 Método de Runge-Kutta de cuarto orden

Los métodos de Runge-Kutta de tercero y cuarto orden se pueden desarrollar en forma análoga a la que se usó para obtener los métodos de 1º y 2º orden. Los métodos de orden 3 rara vez se utilizan. En el caso de los métodos de orden 4, que son los más usados la función f se evalúa en 4 puntos seleccionados. De los métodos de Runge-Kutta de orden 4 sólo veremos el más popular. Este método clásico se puede definir mediante las 5 ecuaciones siguientes:

$$y_{m+1} = y_m + \frac{h}{6}\left(k_1 + 2k_2 + 2k_3 + k_4\right)$$

$$k_1 = f\left(x_m, y_m\right)$$

$$k_2 = f\left(x_m + \frac{h}{2}, y_m + \frac{h\,k_1}{2}\right)$$

$$k_3 = f\left(x_m + \frac{h}{2}, y_m + \frac{h\,k_2}{2}\right) \qquad (38)$$

$$k_4 = f\left(x_m + h, y_m + h\,k_3\right)$$

Requiere 4 evaluaciones de la función f.
El error local de truncamiento es $O(h^5)$.
El error global del método es $O(h^4)$.
Se puede demostrar que el método de Runge-Kutta de 4º orden clásico (que integra a una función de x e y), es una generalización del método de Simpson que permite integrar una función que depende de x.

Ejercicio: Demuestre que el método Modificado de Euler coincide con el desarrollo en serie de Taylor hasta los términos en h^2, y por lo tanto es un método de Runge-Kutta de 2º orden.

Buscamos sustituir $f\left[\left(x_m + \frac{h}{2}\right), \left(y_m + \frac{h}{2}y`_m\right)\right]$ *en* Φ por su correspondiente desarrollo en serie de Taylor.

El desarrollo de f(x,y) se puede escribir como:

$$f(x,y) = f(x_m, y_m) + (x - x_m)\left.\frac{\partial f}{\partial x}\right|_{(x_m, y_m)} + (y - y_m)\left.\frac{\partial f}{\partial y}\right|_{(x_m, y_m)} + \ldots.$$

Sustituimos:

$$x = x_m + \frac{h}{2}$$

$$y = y_m + \frac{h}{2}y`_m = y_m + \frac{h}{2}f\left(x_m, y_m\right)$$

Queda:

$$f\left[(x_m, y_m), \left(y_m + \frac{h}{2} f(x_m, y_m)\right)\right] = f(x_m, y_m) + \frac{h}{2} f_x + \frac{h}{2} f(x_m, y_m) f_y + O(h^2)$$

$$\Phi(x_m, y_m, h) = f\left[\left(x_m + \frac{h}{2}\right), \left(y_m + \frac{h}{2} f(x_m, y_m)\right)\right] = f(x_m, y_m) + \frac{h}{2} f_x + \frac{h}{2} f(x_m, y_m) f_y + O(h^2)$$

Ahora reemplazamos Φ en la ecuación (24):

$$y_{m+1} = y_m + h\, f(x_m, y_m) + \frac{h^2}{2}\left[f_x + f\, f_y\right] + O(h^3)$$

Esta expresión coincide con el desarrollo en serie de Taylor hasta los términos en h^2, así que el método Modificado de Euler es un método de Runge-Kutta de segundo orden.

6 MÉTODOS PREDICTOR-CORRECTOR

6.1 Métodos Multipaso

Son aquellos métodos tales que, para evaluar la ordenada y_{m+1} correspondiente a la abscisa x_{m+1}, utilizan información de varios puntos anteriores. Se los suele llamar "métodos de k pasos", en el que el número de pasos es igual a la cantidad de puntos previos que se necesitan para evaluar y_{m+1}. La forma general de estos métodos es:

$$y_{m+1} = \sum_{j=0}^{p} a_j y_{m-j} + h \sum_{j=-1}^{p} b_j f(x_{m-j}, y_{m-j}) \qquad (39)$$

Donde a_j y b_j son constantes y "p" depende del método. Desarrollamos la expresión anterior

$$y_{m+1} = a_0 y_m + a_1 y_{m-1} + \ldots + a_p y_{m-p} + h\left[b_{-1} f_{m+1} + b_0 f_m + b_1 f_{m-1} + \ldots + b_p f_{m-p}\right] \qquad (40)$$

Si los coeficientes a_p o b_p o ambos son distintos de cero, el método es de p+1 pasos. Es decir, se necesitan p+1 puntos previos para determinar y_{m+1}. Por lo tanto, para iniciar el método requerimos el conocimiento de $y_0, y_1, y_2, \ldots y_n$, . La idea es que estos puntos sean calculados por otros métodos (por ejemplo, con algún método de Runge-Kutta). Aquel método que nos permite obtener los (p+1) puntos iniciales se llama Método Inicializador.

En la forma general de los métodos multipaso observamos que si b_{-1} es distinto de cero, entonces y_{m+1} aparece en el segundo miembro de la expresión:

$$y_{m+1} = a_0 y_m + a_1 y_{m-1} + \ldots + h \cdot [b_{-1} f(x_{m+1}, y_{m+1}) + \ldots + b_p f_{m-p}]$$

Podemos clasificar entonces los métodos multipaso en:

- Métodos Explícitos: si $\boxed{b_{-1} = 0}$ (y_{m+1} aparece sólo en el primer miembro de la fórmula general)
- Métodos Implícitos: si $\boxed{b_{-1} \neq 0}$ (y_{m+1} aparece también en el segundo miembro de la ecuación general).

Veamos un ejemplo de cada uno de estos métodos:

I) Método del Punto Medio:

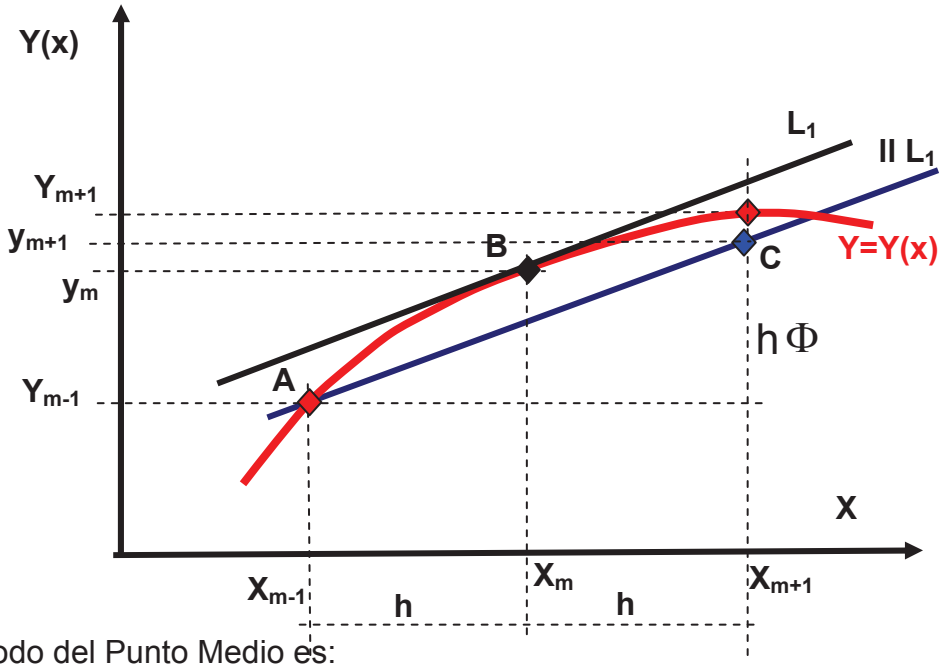
Geométricamente;

- Determinamos la pendiente en el punto (x_m, y_m): $f(x_m, y_m) = f_m$
- Trazamos la recta L_1 de pendiente f_m y que pasa por (x_m, y_m)
- Dibujamos $\overline{L}$ paralela a L_1 que pasa por (x_{m-1}, y_{m-1})
- Ubicamos y_{m+1} donde la recta $\overline{L}$ interfecta a la vertical en $x = x_{m+1}$

La ecuación de la recta $\overline{L}$ es: $y = y_{m-1} + (x - x_{m-1}).f(x_m, y_m)$

Sustituyendo $x = x_{m+1}$ obtenemos y_{m+1} :

$$\boxed{y_{m+1} = y_{m-1} + 2h f(x_m, y_m)}$$ Método del Punto Medio (41)



El Método del Punto Medio es:

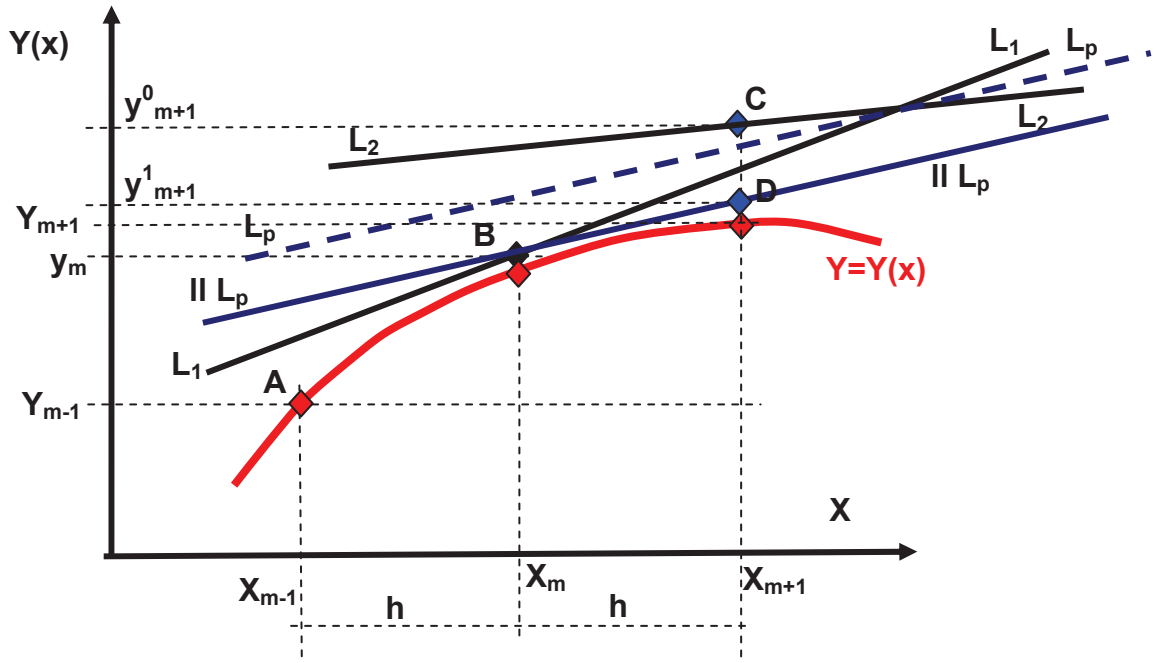
- Un método de segundo orden $ET = -\frac{h^3}{3} y^{III}(\xi)\ ;\ \xi \in [x_{m-1}, x_{m+1}]$
- Un método explícito
- Un método de dos pasos (se necesitan dos puntos previos para evaluar y_{m+1}

En la expresión general de los métodos multipaso vemos que:

$a_0 = 0\ ;\ a_1 \neq 0\ ;\ a_1 = 1\ ;\ b_{-1} = 0\ ;\ b_0 = 2 \quad \therefore \quad p = 1 \Rightarrow Método\ de\ 2\ pasos$

II) Método del Trapecio

Geométricamente: Supongamos que tenemos una primera aproximación a $y_{m+1} = y^0_{m+1}$ calculamos la pendiente aproximada de la recta L_2 que pasa por C$\left(x_{m+1}, y^0_{m+1}\right)$. Trazamos la línea L_1 de pendiente $f\left(x_m, y_m\right)$ en el punto B(x_m, y_m). Luego se calcula L_p que es la pendiente promedio de las dos pendientes anteriores (L_2 y L_1). Trazamos una recta con pendiente //L_p paralela a L_p que pasa por B$\left(x_m, y_m\right)$. Donde esta última recta intersecta a $x = x_{m+1}$ se tiene y_{m+1} (un valor corregido respecto del anterior), que es el punto D(x_{m+1}, y_{m+1}).



Educación de L: $y = y_m + (x - x_m).\frac{1}{2}\left[f\left(x_{m+1}, y^0_{m+1}\right) + f\left(x_m, y_m\right)\right]$

En $x = x_{m+1}$ $\quad \boxed{y_{m+1} = y_m + \frac{h}{2}\left[f\left(x_m, y_m\right) + f\left(x_{m+1}, y^0_{m+1}\right)\right]}$ Método del Trapecio

El Método del Trapecio es:

1. Un Método de segundo orden $\left(ET = Kh^3\right); \left(ET = \frac{-h^3}{12} y^{lll}(\xi); x_{m-1} \le \xi \le x_{m+1}\right)$
2. Es un Método implícito (y_{m+1} aparece en ambos miembros)
3. Es un Método de un paso (sólo se necesita un punto previo para evaluar y_{m+1}). En la fórmula general vemos que: $a_0 = 1 \quad a_1 = 0$ $\quad b_{-1} = \frac{1}{2} \quad b_0 = \frac{1}{2}$ como p=0 ∴ Método de (p+1)=1 paso.

Existen muchísimos métodos multipaso. Ejemplos:

- Métodos de los coeficientes Indeterminados;
- Métodos de Adams:
- Metodos Adams-Bashforth
- Métodos Adams Moulton, etc.

6.2 Métodos Predictor-Corrector: (P-C)

Como su nombre lo indica, primero se predice un valor para y_{m+1} con alguna fórmula. Luego usamos una familia diferente para corregir dicho valor, iterando hasta conseguir la precisión deseada. Existen muchísimos métodos predictor-corrector. Veremos sólo algunos de ellos.

6.2.1 Método predictor-corrector de segundo orden:

Se llama así porque tanto la fórmula predoctora como la correctora son de segundo orden.

Método predictor: Método del punto medio

$$y_{m+1}^{0} = y_{m-1} + 2hf(x_m, y_m)$$ Método explícito de 2 pasos de 2º orden $(ET = Kh^3)$ (43)

El superíndice (0) indica que esta es una primera aproximación para y_{m+1}. Pero para calcular y_1 necesitamos un punto previo a (x_0, y_0), ya que este es un método de dos pasos. Entonces se adopta un método Runge-Kutta como método inicializador (por ejemplo Runge-Kutta de 4º orden).

Método corrector: Método del Trapecio

$$y_{m+1}^{1} = y_m + \frac{h}{2}\left[f(x_m, y_m) + f\left(x_{m+1}, y_{m+1}^{0}\right)\right]$$ Método Implícito de 1 paso de segundo orden $(ET = Kh^3)$ (44)

Podríamos a su vez ahora corregir a este valor, usando $f\left(x_{m+1}, y_{m+1}^{1}\right)$. Entonces, en general la i-ésima aproximación a y_{m+1} está dada por:

$$y_{m+1}^{i} = y_m + \frac{h}{2}\left[f(x_m, y_m) + f\left(x_{m+1}, y_{m+1}^{(i-1)}\right)\right] \qquad i = 1,2,.....$$ (45)

Hasta que:

$$\left|\frac{y_{m+1}^{(i)} - y_{m+1}^{(i-1)}}{y_{m+1}^{(i)}}\right| < \varepsilon \qquad \varepsilon = valor\ positivo\ especificado$$

6.2.2 Método de Heun:

Método predictor: Método de Euler

$$y_{m+1}^{0} = y_m + hf(x_m, y_m)$$ Método explícito de 1º orden ET=Kh^2 de 1 paso (46)

Al ser un método de un paso, no requiere de un método inicializador. Pero la desventaja de este método respecto del anterior es que el ET es mas grande.

Método corrector: método del Trapecio

$$y_{m+1}^{(i)} = y_m + \frac{h}{2}\left[f(x_m, y_m) + f\left(x_{m+1}, y_{m+1}^{(i-1)}\right)\right]$$ Método implícito (de segundo orden) de 1 paso $\left(ET = Kh^3\right)$

Hasta que: $\left|\frac{y_{m+1}^{(i)} - y_{m+1}^{(i-1)}}{y_{m+1}^{(i)}}\right| < \varepsilon$ (47)

Muchos métodos predictor-corrector tienen una fórmula predictora de un orden menor que la correctora.

Se puede probar que el error por truncamiento en el método de Helin es $\left(ET = Kh^3\right)$.

6.2.3 Método de Milne

Método predictor

$$y_{m+1}^{0} = y_{m-3} + 4\frac{h}{3}(2f_m - f_{m-1} + 2f_{m-2})$$

$$ET = \frac{28}{90}h^5 y^{(5)}(\xi)$$

Método explícito de 4 pasos de 4º orden (48)

Método corrector:

$$y_{m+1}^{(i)} = y_{m-1} + \frac{h}{3}\left[f_{m+1}^{(i-1)} + 4f_m + f_{m-1}\right]$$

$$ET = -\frac{1}{90}h^5 y^{(5)}(\xi)$$

Método implícito de 2 pasos de 4º orden (49)

Ambas fórmula son de cuarto orden. Para algunos valores de h este método presenta problemas de estabilidad. Como la fórmula predoctora es de 4 pasos se debe usar un método inicializador durante los 3 primeros paso (por ejemplo: método Runge-Kutta de cuarto orden).