

Taller de Métodos Numéricos

Guía de estudio

Marcos Prunello

Cecilia Rapelli

2020-03-06

Índice general

Acerca del Taller	3
Horario y lugar de cursado	3
Consultas	3
Material de estudio	3
Evaluación	3
Campus virtual	3
1 Introducción a los Métodos Numéricos	5
1.1 Cifras significativas	5
1.2 Notación Científica o Exponencial	6
1.3 Sistemas numéricos	6
1.4 ¿Por qué nos interesa el sistema binario?	7
1.5 Formato de coma flotante	7
1.6 Propiedades importantes del formato IEEE-754	8
1.7 Análisis de los errores	11
2 Solución Numérica de Ecuaciones No Lineales	13
2.1 Generalidades	13
2.2 Método de las Aproximaciones Sucesivas o del Punto Fijo	14
2.3 Método de Newton-Raphson	18
2.4 Método de von Mises	22
2.5 Método de Newton-Raphson de 2º Orden	23
3 Solución de Sistemas de Ecuaciones Lineales	25
3.1 Generalidades	25
3.2 Repaso	25
3.3 Notación	26
3.4 Métodos de Resolución de Sistemas de Ecuaciones	26
3.5 Sistemas fáciles de resolver	26
3.6 Eliminación gaussiana	27
3.7 Estrategias de pivoteo para reducir los errores	29
3.8 Método de eliminación de Gauss-Jordan	29
3.9 Métodos Aproximados o Iterativos	31
4 Aproximación Polinomial: interpolación y extrapolación	37
4.1 Generalidades	37
4.2 Diferencias finitas	37
4.3 Interpolación de Newton para incrementos constantes	39
4.4 Interpolación de Lagrange	44

Acerca del Taller

La presente guía resume algunos conceptos desarrollados en el Taller de Métodos Numéricos de la Licenciatura en Estadística en la Facultad de Ciencias Económicas y Estadística, Universidad Nacional de Rosario. La misma irá siendo revisada a lo largo del cuatrimestre y no está exenta de presentar errores o expresar ideas que puedan ser mejoradas. ¡Esperamos que juntos podamos enriquecerla al dar nuestros primeros pasos en la programación!

Horario y lugar de cursado

Martes de 8:00 a 10:00 en el Laboratorio de Estadística (2do piso).

Consultas

A través del Foro de Consultas en el Campus Virtual o al finalizar las clases.

Material de estudio

El material de estudio está compuesto por esta guía y prácticas que, junto con cualquier otro material que necesitemos, estarán disponibles en nuestro espacio en el Campus Virtual de la UNR.

Todo el material fue creado utilizando el lenguaje de programación estadística **R** y las herramientas del entorno de desarrollo integrado **RStudio**.

Evaluación

En construcción.

Campus virtual

Además de alojar todo el material del curso, utilizaremos este espacio para la entrega de trabajos y para realizar consultas, que esperamos puedan ser debatidas y respondidas entre el estudiantado.

Unidad 1

Introducción a los Métodos Numéricos

- Son aquellos algoritmos que permiten resolver de forma aproximada problemas matemáticos que involucran el cálculo de determinados valores y no pueden ser abordados mediante técnicas analíticas, o cuya resolución analítica exacta tiene un costo o complejidad muy elevados.
- Por ejemplo, si tenemos que hallar el valor de x tal que $x + 2 = 5$, el cálculo es muy sencillo.
- Pero se complica si tenemos que calcular:

$$F(x_0) = \int_{-\infty}^{x_0} \frac{1}{\sqrt{2\pi}\sigma^2} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2} dx$$

- Es decir, el Análisis Numérico trata de diseñar métodos para aproximar, de una manera eficiente, las soluciones de problemas expresados matemáticamente.
- Naturalmente, a los métodos numéricos les interesa controlar la diferencia entre la solución aproximada y el valor verdadero, que recibe el nombre de **error**.
- Están relacionados con muchas disciplinas, como la algorítmica y la programación, y se aplican en diversas áreas, como en la Estadística.
- El estudio y desarrollo de los métodos numéricos abarca los siguientes aspectos:
 - Análisis de las propiedades de convergencia de los métodos numéricos
 - Estudio de la programación de los métodos numéricos (implementación)
 - Aplicación de los métodos numéricos en un determinado campo
- Algunos de los problemas que toca el Análisis Numérico son los siguientes:
 - Resolución aproximada de ecuaciones algebraicas y sistemas de ecuaciones no lineales.
 - Problemas de optimización, en los que se maximiza o se minimiza una función.
 - Problemas de tipo matricial (hallar valores propios, invertir matrices, etc.)
 - Resolución de sistemas de ecuaciones lineales con gran número de ecuaciones e incógnitas que por su costo de cálculo son irresolubles por métodos clásicos como la regla de Cramer.
 - Resolución aproximada de ecuaciones diferenciales.
 - Problemas de interpolación.
 - Problemas de integración o derivación aproximada de funciones poco manejables.

Ejemplo 1

- No hay dudas que:

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

- ¿Pero qué sucede si hacemos $(10^{-9} + 10^9) - 10^9$ en la calculadora?
- En esta unidad nos dedicaremos a repasar y definir conceptos relacionados con la representación numérica de magnitudes, para intentar entender qué sucedió con la operación anterior en nuestra calculadora.

1.1. Cifras significativas

- Las cifras significativas de un número son las que aportan alguna información.
- Reglas:

- Son los coeficientes 1, 5, 6 y 3 los que definen la representación de este número como 1563 en el sistema decimal.
- Pero si en vez de usar potencias con base 10, usamos potencias con base 2, el número 1563 se escribe como:

$$\begin{aligned}
 1563 &= 1024 + 512 + 16 + 8 + 2 + 1 \\
 &= (1 \times 2^{10}) + (1 \times 2^9) + (1 \times 2^4) + (1 \times 2^3) + (1 \times 2^1) + (1 \times 2^0) \\
 &= (1 \times 2^{10}) + (1 \times 2^9) + (0 \times 2^8) + (0 \times 2^7) + (0 \times 2^6) + \\
 &\quad (0 \times 2^5) + (1 \times 2^4) + (1 \times 2^3) + (0 \times 2^2) + (1 \times 2^1) + (1 \times 2^0)
 \end{aligned}$$

- Como en el sistema decimal, son los coeficientes los que definen la representación del número, por lo que 1563 en **sistema binario** o en **base dos** es igual a $11000011011_{(2)}$.

1.4. ¿Por qué nos interesa el sistema binario?

- Porque es el sistema de representación numérica que utilizan las computadoras.
- Este sistema es natural para las computadoras ya que su memoria consiste de un enorme número de dispositivos de registro electrónico, en los que cada elemento sólo tiene los estados de “encendido” y “apagado”.
- Estos elementos constituyen la unidad mínima de información, sólo pueden tomar dos valores posibles, 0 o 1, y reciben el nombre de **bit** (*binary digit*).
- Aunque nosotros no nos damos cuenta, toda operación numérica que le indicamos a la computadora en sistema decimal, es traducida y procesada internamente en binario.
- Por lo tanto es muy importante entender cómo opera la computadora, para entender qué sucede con las operaciones que queremos que realice. Por ejemplo...

1.4.1. Ejemplo 3

- Escribir un programa para realizar las siguientes operaciones, empleando estructuras iterativas para las sumatorias:
 - a) $10\,000 - \sum_{i=1}^{100\,000} 0,1$
 - b) $10\,000 - \sum_{i=1}^{80\,000} 0,125$
- ¿Cuál es el resultado exacto en estos cálculos? ¿Qué resultados arrojó la computadora?
- ¿Por qué ocurre esto?
 - Pensemos en el número decimal periódico $1/3 = 0,\bar{3}$.
 - Para aproximarlos, sólo podemos usar una cantidad finita de cifras, por ejemplo, 0,333 o 0,33333. Estas aproximaciones guardan cierto error, que depende de la cantidad de cifras empleadas.
 - Con los números binarios ocurre exactamente lo mismo.
 - $0,1_{(10)} = 0,0001100110011\dots_{(2)} = 0,0\bar{0011}_{(2)}$. Es decir, la representación de 0,1 en binario es periódica, la computadora necesariamente debe redondear o truncar para almacenar y operar.
 - Por esta razón, sumar 100 mil veces 0,1 no da exactamente 10000.
 - Por el contrario, 0,125 en binario no es periódico, la computadora lo puede representar exactamente y no se produjo error.

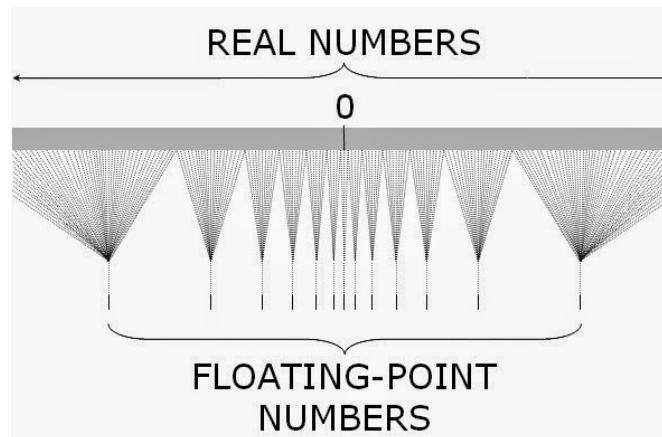
1.5. Formato de coma flotante

- La **representación de coma flotante** (en inglés, *floating point*) es una forma de notación científica usada en los computadores con la cual se pueden representar números reales extremadamente grandes y pequeños de una manera muy eficiente y compacta, y con la que se pueden realizar operaciones aritméticas.
- El estándar que establece las reglas para este tipo de representación se llama **IEEE-754**.
- En este formato, la base es 2 y tanto el exponente como el coeficiente pueden contener una cierta cantidad limitada de cifras binarias, lo cual implica que dada una cantidad x , la computadora sólo es capaz de almacenar una aproximación binaria a x :

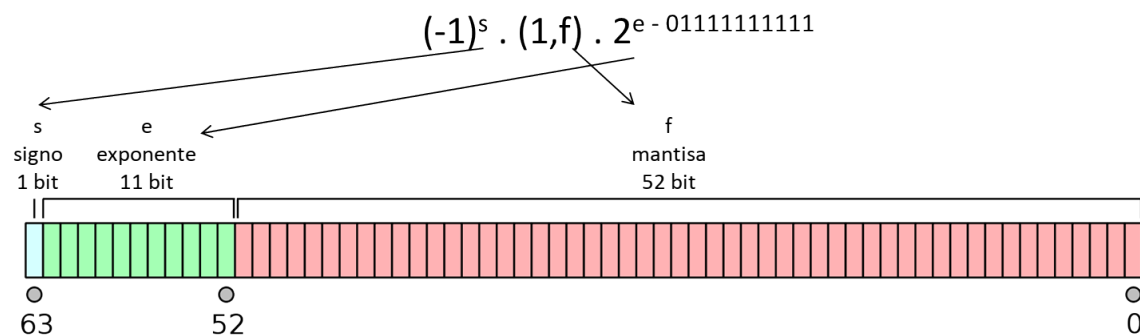
$$x \approx \pm m \times 2^n$$

donde el exponente n y coeficiente m (también llamado **mantisa**) son números binarios con una longitud establecida.

- A través de este sistema, las computadoras sólo pueden representar un subconjunto de los números racionales y no pueden representar números irracionales como π o $\sqrt{3}$ dado que tienen infinitos decimales no periódicos.
- Esto hace que en la representación surjan los errores de redondeo que ya vimos en el ejemplo 3.



- El **IEEE-754** define dos tipos de formatos: el de precisión simple (en el cual cada número ocupa 32 bit de memoria) y el de precisión doble (un número ocupa 64 bit).
- El formato de doble precisión en 64 bit, empleado actualmente en casi todas las computadoras, tiene la siguiente estructura.



- El bit que indica el signo, s , vale 0 si el número es positivo o 1 si es negativo.
- El exponente e ocupa 11 bit dando lugar a $2^{11} = 2048$ valores distintos. Se dice que es un exponente sesgado porque se le resta $0111111111_{(2)} = 1023_{(10)}$, por lo que puede variar entre -1023 y 1024.
- La mantisa ($1, f$) se registra *normalizada*, es decir con un solo dígito antes de la coma igual a 1, de manera que dicho 1 no se almacena, es implícito. Luego de la coma, la parte fraccionaria f ocupa 52 bit (2^{52} valores distintos).
- Por ejemplo, el siguiente conjunto de 64 bit representa al número decimal -74,5 (verificación opcional).

[illegible]

- En [este link](#) se puede encontrar una calculadora que convierte números entre sus representaciones en decimal y en coma flotante.

1.6. Propiedades importantes del formato IEEE-754

- Además de representar números, este sistema permite almacenar ciertos valores especiales. Uno de ellos es *NaN* (*Not a Number*), para resultados no definidos matemáticamente. Por ejemplo:

0/0

```
[1] NaN
```



```
sqrt(-1)
```

```
[1] NaN
```

Operaciones como $x + \text{NaN}$, $\text{NaN} \times x$, NaN / x , x / NaN resultan en NaN (excepto $\text{NaN}^0 = 1$).

****Observación**:** en R `NaN` no es lo mismo que `NA` (*Not Available*). `NA` es una construcción de R, que

- El infinito tiene su propia representación en este estándar y puede ser positivo o negativo. Un valor infinito puede darse como resultado de ciertas operaciones o por exceder los límites de almacenamiento. En R se representan con `Inf` y `-Inf`.

```
Inf
```

```
[1] Inf
```

```
Inf - Inf
```

```
[1] NaN
```

```
Inf / Inf
```

```
[1] NaN
```

- La división por cero resulta en infinito, aunque matemáticamente sea una indefinición. Y como contrapartida, dividir por infinito da cero.

```
3/0
```

```
[1] Inf
```

```
-3/0
```

```
[1] -Inf
```

```
3/Inf
```

```
[1] 0
```

- Infinito es el próximo valor al mayor valor que puede registrarse bajo este formato. En R, existe un objeto llamado `.Machine` que es una lista con información numérica de la máquina en la que se está trabajando, por ejemplo, cuál es el mayor número en coma flotante normalizada. Si a este valor lo multiplicamos, por ejemplo, por 2, tenemos infinito:

```
.Machine$double.xmax
```

```
[1] 1.797693e+308
```

```
.Machine$double.xmax * 2
```

```
[1] Inf
```

- El 0 tiene signo, puede ser $+0$ o -0 . -0 actúa como 0 y se muestra como 0, pero el signo negativo se propaga en las operaciones.

```
-0
```

```
[1] 0
```

```
3 * -0
```

```
[1] 0
```

```
1 / 0
```

```
[1] Inf
```

```
1 / -0
```

```
[1] -Inf
```

- Como ya dijimos, bajo este sistema no podemos representar todos los números reales. El **error de máquina** o **epsilon de máquina** es una medida del espacio que hay entre dos números consecutivos en la computadora. Se define como el menor número $\epsilon > 0$ tal que $1 + \epsilon > 1$.

Si bien $1 + x > 1$ debería verificarse para cualquier $x > 0$, en la computadora no es así, porque si x es muy pequeño, $1 + x$ será redondeado a 1. Por eso, el ϵ de máquina es aquella cantidad mínima necesaria para que $1 + \epsilon$ se distinga de 1.

En R, podemos averiguar el epsilon de máquina así:

```
.Machine$double.eps
```

```
[1] 2.220446e-16
```

Podemos usar la función `print` para elegir la cantidad de decimales que queremos ver y así experimentar con el ϵ de máquina:

```
print(1 + .Machine$double.eps, digits = 20)
```

```
[1] 1.000000000000000222
```

```
print(1 + .Machine$double.eps * 2, digits = 20)
```

```
[1] 1.0000000000000004441
```

```
print(1 + .Machine$double.eps / 2, digits = 20)
```

```
[1] 1
```

Aunque nos parezca despreciable, esto puede ser una limitación a considerar, por ejemplo, si trabajamos con medidas de partículas subatómicas.

Si hacemos $1000 + \epsilon$, encontraremos que el resultado es 1000. Es decir, $1000 + \epsilon$ no se distingue de 1000. Es porque ϵ está definido con respecto a 1. Como los números en coma flotante son cada vez más espaciados cuanto más nos alejamos de 0, a 1000 habrá que sumarle un x algo más grande para que $1000 + x$ se distinga de 1000.

```
print(1000 + .Machine$double.eps, digits = 20)
```

```
[1] 1000
```

```
print(1000 + 100 * .Machine$double.eps, digits = 20)
```

```
[1] 1000
```

```
print(1000 + 300 * .Machine$double.eps, digits = 20)
```

```
[1] 1000.000000000001137
```

- **Overflow (desbordamiento)**. Es el fenómeno que ocurre cuando un cálculo produce un resultado cuya magnitud es mayor que la que se puede almacenar o representar (en R, `.Machine$double.xmax`). Aunque este número probablemente es mucho mayor que cualquier número que utilicemos en la práctica, debería ser tenido en cuenta, por ejemplo, si hacemos cálculos en cosmología.

```
.Machine$double.xmax
```

```
[1] 1.797693e+308
```

```
.Machine$double.xmax * 2
```

```
[1] Inf
```

- **Underflow (desbordamiento)**. Es el fenómeno que ocurre cuando el resultado de una operación

en coma flotante es menor en magnitud (más cercano a cero) que el menor número representable (en R, `.Machine$double.xmin`). Es difícil mostrar un ejemplo de esto porque, dependiendo de la máquina, la representación en coma flotante puede recurrir a números no normalizados para lograr números incluso por fuera de los límites de desbordamiento:

```
# Mínimo representable en coma flotante normalizada
.Machine$double.xmin
```

```
[1] 2.225074e-308
```

```
# Igual es posible conseguir números menores con representación no normalizada
.Machine$double.xmin / 2
```

```
[1] 1.112537e-308
```

Observación: el mínimo representable `.Machine$double.xmin` es menor que el ϵ de máquina ya que $1 < 2$.

1.7. Análisis de los errores

- Como ya hemos dicho, los métodos numéricos proporcionan una solución aproximada de los problemas que tratan de resolver.
- Y aunque es cierto que algunos de los métodos que se estudiarán idealmente proporcionarían una solución exacta, esto tampoco ocurrirá por las aproximaciones que realiza la computadora en la representación numérica.
- Denominamos **error** a la diferencia entre la solución que los métodos, una vez programados, devuelven y la solución exacta del problema que se trata de resolver. Hay dos tipos fundamentales de errores.

1.7.1. Error de truncamiento

- Dado que un método numérico propone un algoritmo para resolver de forma aproximada un problema que no se puede resolver mediante métodos analíticos, se llama **error de truncamiento** a la diferencia entre el valor aproximado propuesto por el método y la solución exacta del problema.
- Ocurre cuando un proceso que requiere un número infinito de pasos se detiene en un número finito de pasos.
- Por ejemplo, podemos recordar el desarrollo en serie de Taylor de la función $f(x) = e^{x^2}$:

$$e^{x^2} = 1 + x^2 + \frac{x^4}{2!} + \frac{x^6}{3!} + \dots + \frac{x^{2n}}{n!} + \dots$$

- Si nos quedamos sólo con los primeros 4 términos, estamos aproximando una suma que tiene infinita cantidad de sumandos sólo con los primeros 4, de manera que dicha aproximación presentará un error de truncamiento.
- Este tipo de error no depende directamente del sistema numérico que se emplee.

1.7.2. Error de redondeo

- Resulta de reemplazar un número por su forma en coma flotante, es decir, por su representación en la computadora mediante un número finito de bits.
- Recibe este nombre ya sea que la aproximación se realice con redondeo o poda (también conocida como *truncamiento*, pero no debe confundirse con el error de truncamiento).
- El error de redondeo está ligado fundamentalmente al tipo de precisión que se emplee (algo determinado por el procesador y el software usados).
- Sin embargo, el efecto final de los errores de redondeo depende también del algoritmo propuesto por el método numérico y por la forma de programarlo.
- Existen operaciones que son especialmente sensibles a los errores de redondeo o un algoritmo puede hacer que los mismos se amplifiquen.

1.7.3. Ejemplos de operaciones “delicadas”

Sustracción de números casi iguales en valor absoluto

- Da lugar a importantes pérdidas de precisión.

- El resultado tiene menos cifras significativas que los valores originales (**pérdida de cifras significativas o cancelación catastrófica**).
- Por ejemplo: sean $p = 3.1415926536$ y $q = 3.1415957341$.
- Tienen 11 cifras significativas cada uno.
- Sin embargo, $p - q = -0.0000030805$ tiene sólo 5 cifras significativas.
- Esto puede producir una reducción en la precisión final de la respuesta calculada.

División por cantidades pequeñas

- Un error mínimo en el dividendo se traduce en uno mucho mayor en el resultado, de modo que la falta de precisión podría ocasionar un *overflow* o pérdida de cifras significativas.
- Veremos un ejemplo de esto cuando resolvamos sistemas de ecuaciones lineales con la estrategia de pivoteo en el algoritmo de eliminación de Gauss.
- Dado que los números de punto flotante están más concentrados cerca del cero entonces al dividir por un número más grande es más probable conseguir una mejor aproximación.

Adición de un número grande y uno pequeño

- Puede hacer que el pequeño desaparezca.
- En ciertos casos esto no ocasiona un problema ya que, si tenemos un número de gran magnitud probablemente podamos considerar al más pequeño despreciable.
- Sin embargo debe tenerse mucho cuidado con el orden de las operaciones ya que si, por ejemplo, sumamos una gran cantidad de números pequeños entre ellos (que juntos tienen un peso considerable) y luego se lo sumamos a un número grande, todo funcionará correctamente.
- Pero si vamos sumando uno por uno los números pequeños al grande entonces en cada paso el número pequeño será considerado despreciable y llegaremos a un resultado erróneo.

1.7.4. Medida del error

Existen dos formas fundamentales de medir los errores que se cometen en la aproximación de la solución de un problema mediante un método numérico. Siendo x el valor exacto y \hat{x} su aproximación, definimos:

- **Error absoluto:** $E_a = |x - \hat{x}|$. Se mide en las mismas unidades de la variable que se trata de aproximar (por ejemplo, en amperios si se quiere aproximar la corriente que circula por un circuito eléctrico).
- **Error relativo:** $E_r = \frac{|x - \hat{x}|}{|x|}$, $x \neq 0$. Relaciona el error obtenido con la magnitud de la propia solución y se interpreta como porcentaje. Esta medida del error es invariante a los cambios de escala, ya que es independiente de las unidades de medida.

1.7.5. Error propagado

- Se define como **error propagado** al error que se tiene al final de una cadena de operaciones sucesivas por la existencia de diferentes errores en los pasos intermedios.
- Por ejemplo, si tenemos dos valores exactos p y q con valores aproximados \hat{p} y \hat{q} cuyos errores son ϵ_p y ϵ_q de modo que $\hat{p} = p + \epsilon_p$ y $\hat{q} = q + \epsilon_q$, al realizar la suma entre los valores aproximados:

$$\hat{p} + \hat{q} = (p + \epsilon_p) + (q + \epsilon_q) = (p + q) + (\epsilon_p + \epsilon_q)$$

1.7.6. Error propagado

- Si bien es normal que en una cadena los errores iniciales se propaguen, es deseable que un error pequeño en el comienzo produzca errores pequeños en el resultado final.
- Un algoritmo con esta cualidad se llama **estable** (el error se puede acotar), en caso contrario se dice **inestable**.
- Supongamos que ϵ representa un error inicial y que $\epsilon(n)$ representa el crecimiento de dicho error después de n operaciones:
 - Si $|\epsilon(n)| \propto n\epsilon$, el crecimiento es **lineal**.
 - Si $|\epsilon(n)| \propto k^n \epsilon$, el crecimiento es **exponencial**.
 - Si $k > 1$, el error crece cuando $n \rightarrow \infty$ sin que podamos acotarlo, el proceso es inestable.
 - Si $0 < k < 1$, el error decrece cuando $n \rightarrow \infty$, se puede acotar, el proceso es estable.

Unidad 2

Solución Numérica de Ecuaciones No Lineales

2.1. Generalidades

- En esta unidad estudiaremos uno de los problemas básicos y antiguos de la aproximación numérica: la **solución de ecuaciones**.
- Consiste en obtener una solución de una ecuación $F(x) = 0$.
- Se presenta en una gran variedad de problemas.
- Las soluciones de una ecuación se llaman **raíces** o **ceros**.
- Una **ecuación lineal** es una igualdad que involucra una o más variables elevadas a la primera potencia y no contiene productos entre las variables (involucra solamente sumas y restas de las variables). Por ejemplo: $3x + 2 = 8$.
- Para este tipo de ecuaciones es posible hallar analíticamente una expresión para su solución.
- En una **ecuación no lineal** las incógnitas están elevadas a potencias distintas de 1, o aparecen en denominadores o exponentes, o están afectadas por funciones no lineales (como el logaritmo o las trigonométricas).
- Un tipo de ecuación no lineal es la **ecuación algebraica**, que se trata de un polinomio igualado a cero:

$$P_n(x) = a_0x^n + a_1x^{n-1} + \dots + a_{n-1}x + a_n = 0$$

donde $a_0 \neq 0, n \in \mathbb{N}$ y a_0, \dots, a_n son constantes.

- Ejemplo: $x^3 - x^2 + 5x - 8 = 2x^5$.
- Sabemos que si, por ejemplo, $n = 2$, la solución de $ax^2 + bx + c = 0$ está dada por la resolvente:

$$x_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

- Sin embargo, la solución analítica para este tipo de ecuaciones existe sólo para $n \leq 4$.
- Las restantes ecuaciones no lineales se dice que son **trascendentes**, por ejemplo:

$$\begin{aligned} x^3 - \ln x + \frac{3}{x} &= 2 \\ \operatorname{tg}(x + 45) &= 1 + \operatorname{sen}(2x) \\ xe^x &= 1 \\ 5^x &= 9^{x+1}3^x \end{aligned}$$

- En general, tampoco es posible hallar de manera analítica una solución exacta para estas ecuaciones.
- Excepto para algunos problemas, las ecuaciones no lineales carecen de solución exacta, por lo que requieren ser resueltas con métodos numéricos.
- Una técnica fundamental de los métodos numéricos es la **ITERACIÓN** (métodos iterativos).

- Se trata de **repetir un proceso** hasta que se obtiene un resultado para el problema.
- En la unidad se verán distintos métodos iterativos para encontrar las raíces, cada con sus propias ventajas y limitaciones.
- Requieren dos pasos generales:
 1. Determinación de un valor aproximado de la raíz que se busca.
 2. Mejoramiento de la solución hasta lograr un grado de precisión preestablecido.

2.2. Método de las Aproximaciones Sucesivas o del Punto Fijo

2.2.1. Punto fijo

Definición de Punto Fijo

- Un punto fijo de una función $f(x)$ es un número real P tal que $f(P) = P$.
- Ejemplos:
 - $f(x) = x^2 - 3x + 4$, 2 es un punto fijo de f porque $f(2) = 2$.
 - $f(x) = x^2$, 0 y 1 son puntos fijos de f porque $f(0) = 0$ y $f(1) = 1$.

¿Cómo encontrar un punto fijo de $f(x)$?

Sea f una función continua y $p_0, p_1, \dots, p_n, \dots$ una sucesión generada a partir de $p_n = f(p_{n-1})$ con un valor inicial p_0 , es decir:

$$\begin{aligned} & p_0 \\ & p_1 = f(p_0) \\ & p_2 = f(p_1) \\ & \vdots \\ & p_n = f(p_{n-1}) \\ & \vdots \end{aligned}$$

Si $\lim_{n \rightarrow \infty} p_n = P$, entonces P es un punto fijo de $f(x)$.

2.2.2. Empleo para la resolución de ecuaciones

Siendo:

$$F(x) = 0 \tag{2.1}$$

la ecuación a resolver, el **Método de las Aproximaciones Sucesivas** propone reescribirla a través de la ecuación equivalente:

$$f(x) = x$$

de manera que la tarea de hallar un valor de x que satisface (2.1) es lo mismo que hallar un punto fijo de la función $f(x)$.

Entonces, el método para resolver $F(x) = 0$ consiste en:

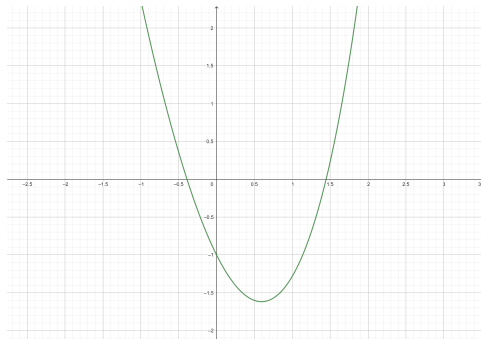
1. Expresar la ecuación en la forma $x = f(x)$.
2. Elegir un valor inicial adecuado x_0 .
3. Realizar el siguiente cálculo iterativo:

$$\begin{aligned}
 x_1 &= f(x_0) \\
 x_2 &= f(x_1) \\
 &\vdots \\
 x_n &= f(x_{n-1}) \\
 &\vdots
 \end{aligned}$$

- Si a medida que n crece los x_n se aproximan a un valor fijo, se dice que el método converge y la iteración se detiene cuando la diferencia entre dos valores consecutivos x_{n-1} y x_n sea tan pequeña como se desee.
- El valor x_n será una raíz aproximada de $F(x)$.

2.2.3. Ejemplo

- Hallar las raíces de la ecuación no lineal: $F(x) = x^2 - 3x + e^x - 2 = 0$
- Graficamos y vemos que las raíces están cercanas a -0.4 y 1.4.



- Reescribimos $F(x) = 0$ como $f(x) = x$
- Por ejemplo:

$$\begin{aligned}
 F(x) &= x^2 - 3x + e^x - 2 = 0 \\
 \Rightarrow \underbrace{\frac{x^2 + e^x - 2}{3}}_{f(x)} &= x \\
 \Rightarrow f(x) &= \frac{x^2 + e^x - 2}{3}
 \end{aligned}$$

- Para $x_0 = -1.5$, el proceso converge al valor -0.390271 que consideraremos como la aproximación para la raíz buscada.

Iteracion	x
0	-1.5000000
1	0.1577101
2	-0.2681003
3	-0.3877637
4	-0.3903555
5	-0.3902688
6	-0.3902718
7	-0.3902717

2.2.4. Criterios para detener el proceso iterativo

- Criterios para convergencia:

1. Error absoluto: $|x_{j+1} - x_j| < \epsilon$

2. Error relativo: $\left| \frac{x_{j+1} - x_j}{x_j} \right| < \epsilon$
3. Error relativo respecto al valor inicial: $\left| \frac{x_{j+1} - x_j}{x_0} \right| < \epsilon$
4. $|F(x_j)| < \epsilon$
 - **Criterios para divergencia:**
 1. $j > r$, r número máximo de iteraciones
 2. $|x_j - x_1| > k$
 3. $|F(x_j)| > k$
 4. $|x_{j+1} - x_j| > k$
 5. $\left| \frac{x_j}{x_1} \right| > k$

2.2.5. Ejemplo

- En cada paso calculamos el error relativo y nos detuvimos cuando el mismo fue menor a 1E-6.

Iteracion	x	Error
0	-1.5000000	NA
1	0.1577101	1.105140e+00
2	-0.2681003	2.699957e+00
3	-0.3877637	4.463383e-01
4	-0.3903555	6.684034e-03
5	-0.3902688	2.222629e-04
6	-0.3902718	7.690434e-06
7	-0.3902717	2.657452e-07

2.2.6. Teorema del Punto Fijo

- Pero esto no funciona siempre, para cualquier f o cualquier x_0 ...
- ¿Cuándo sí? Cuando se cumplen las condiciones del **Teorema del Punto Fijo**.
- A saber:

Dadas las siguientes condiciones:

- (a) f es una función continua en el intervalo $[a, b]$
- (b) $f(x) \in [a, b] \quad \forall x \in [a, b]$
- (c) f' existe en (a, b) con $|f'(x)| \leq m < 1 \quad \forall x \in (a, b)$

Si x_0 es cualquier número en $[a, b]$, entonces la sucesión definida por

$$x_n = f(x_{n-1}), \quad n \geq 1,$$

converge al único punto fijo que f posee en $[a, b]$.

2.2.7. Ejemplo

- En el ejemplo anterior, dada la ecuación $F(x) = x^2 - 3x + e^x - 2 = 0$, la reexpresamos como:

$$x = \frac{x^2 + e^x - 2}{3} \Rightarrow f(x) = \frac{x^2 + e^x - 2}{3}$$

$$\Rightarrow f'(x) = \frac{1}{3}(2x + e^x)$$

- Verificar condiciones del Teorema.
- Si no se cumplen las condiciones, podemos probar con otra expresión para $f(x)$.

2.2.8. Interpretación gráfica

- Dado que el método plantea encontrar el valor de x que satisface $x = f(x)$, resolver la ecuación original es equivalente a resolver el sistema:

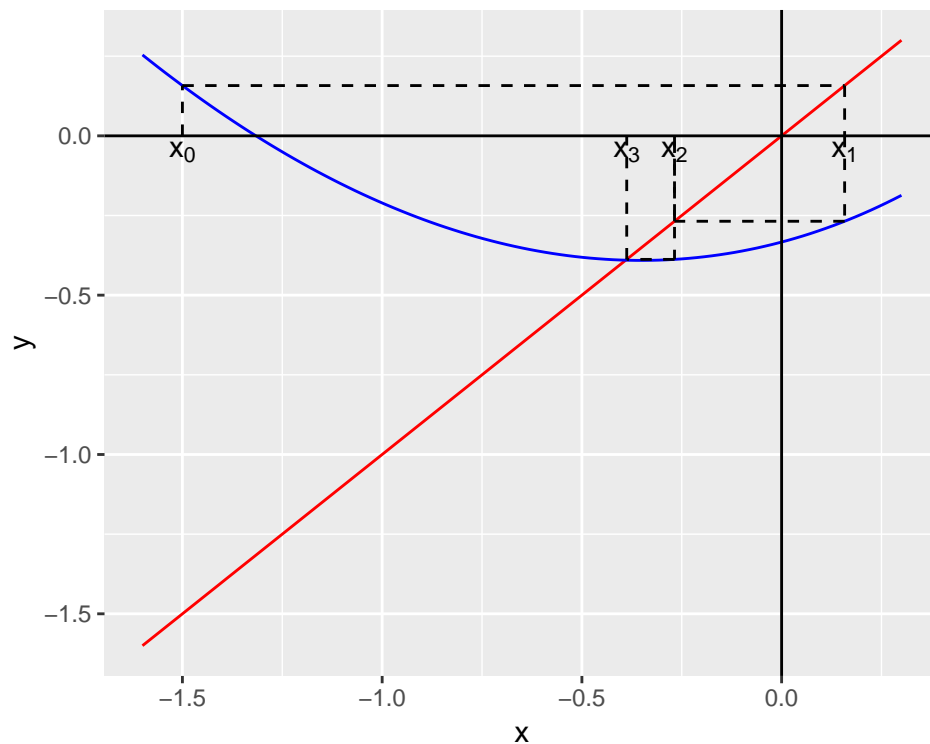
$$\begin{cases} y = f(x) \\ y = x \end{cases} \quad (2.2)$$

- Es decir, que geométricamente el valor buscado es el punto de intersección de la curva $y = f(x)$ con la recta $y = x$.

2.2.9. Ejemplo

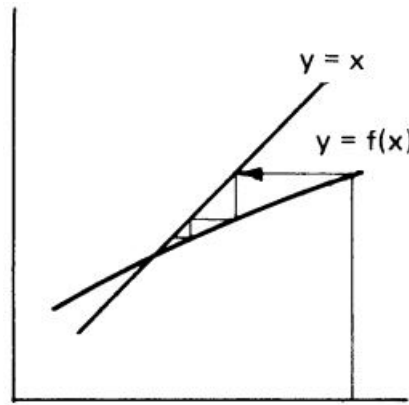
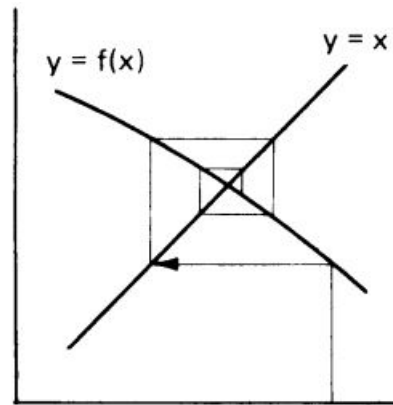
- Para $x_0 = -1.5$, el proceso converge en 7 iteraciones a la raíz -0.390271 con un error relativo menor a $-1E+6$.

Iteración	x	Error
0	-1.500000	NA
1	0.1577101	1.105140e+00
2	-0.2681003	2.699957e+00
3	-0.3877637	4.463383e-01
4	-0.3903555	6.684034e-03
5	-0.3902688	2.222629e-04
6	-0.3902718	7.690434e-06
7	-0.3902717	2.657452e-07

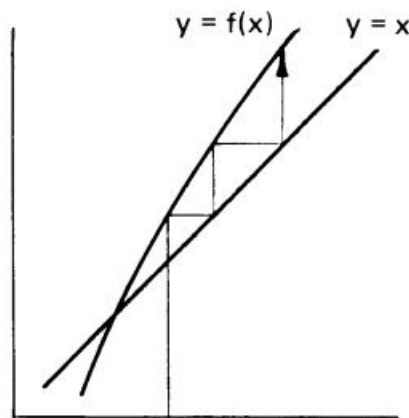
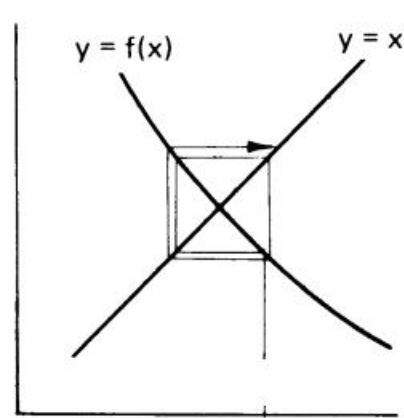


2.2.10. Algunos diagramas

Ejemplos de convergencia:

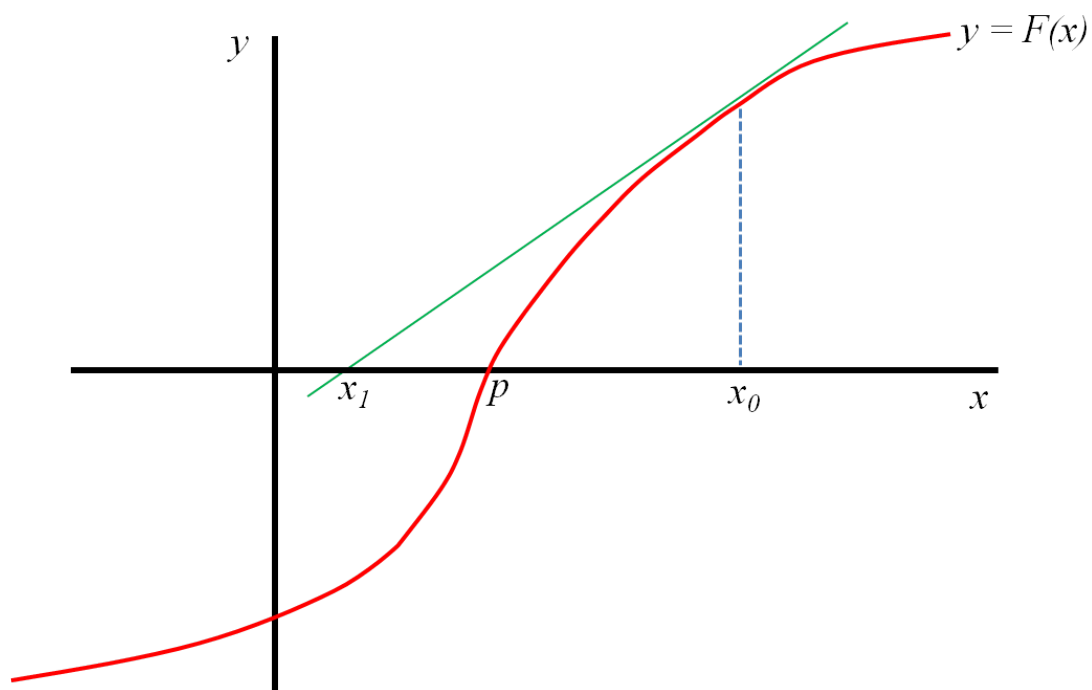
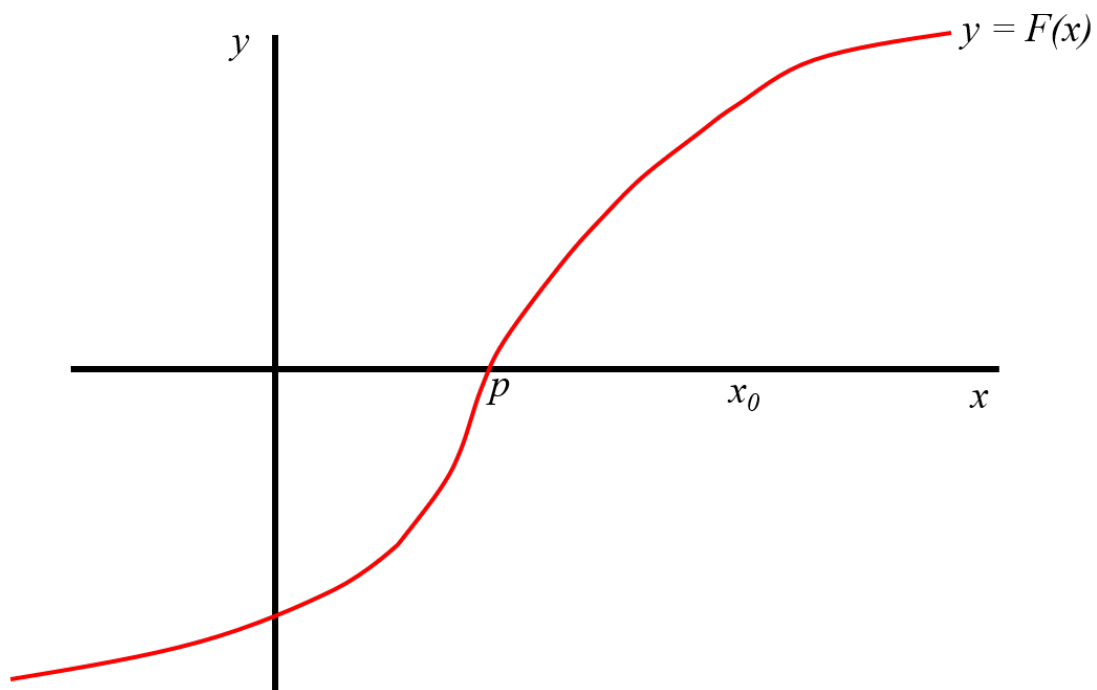
Convergencia monótona: $0 < f' < 1$ Convergencia oscilante: $-1 < f' < 0$

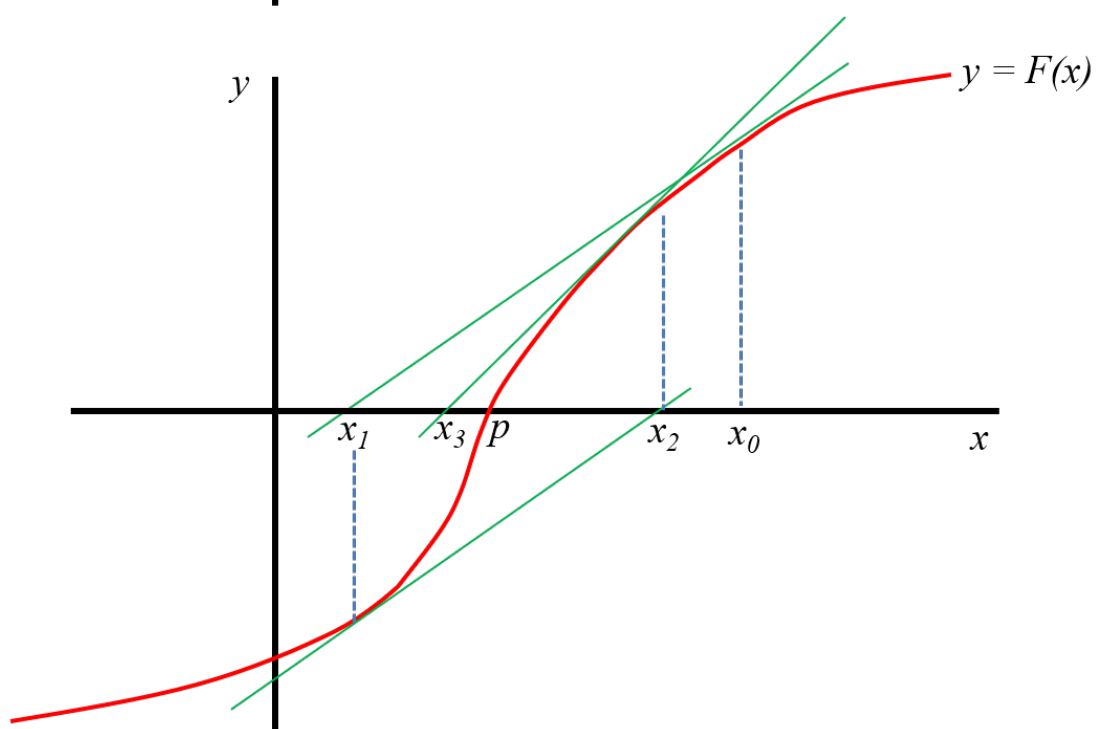
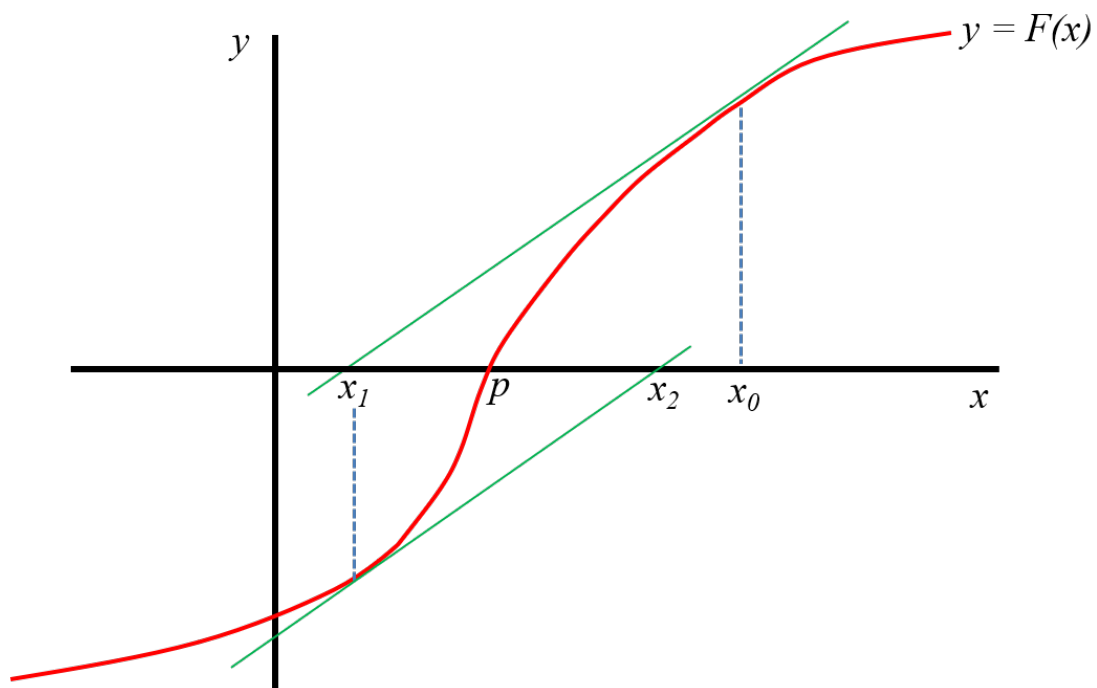
Ejemplos de divergencia:

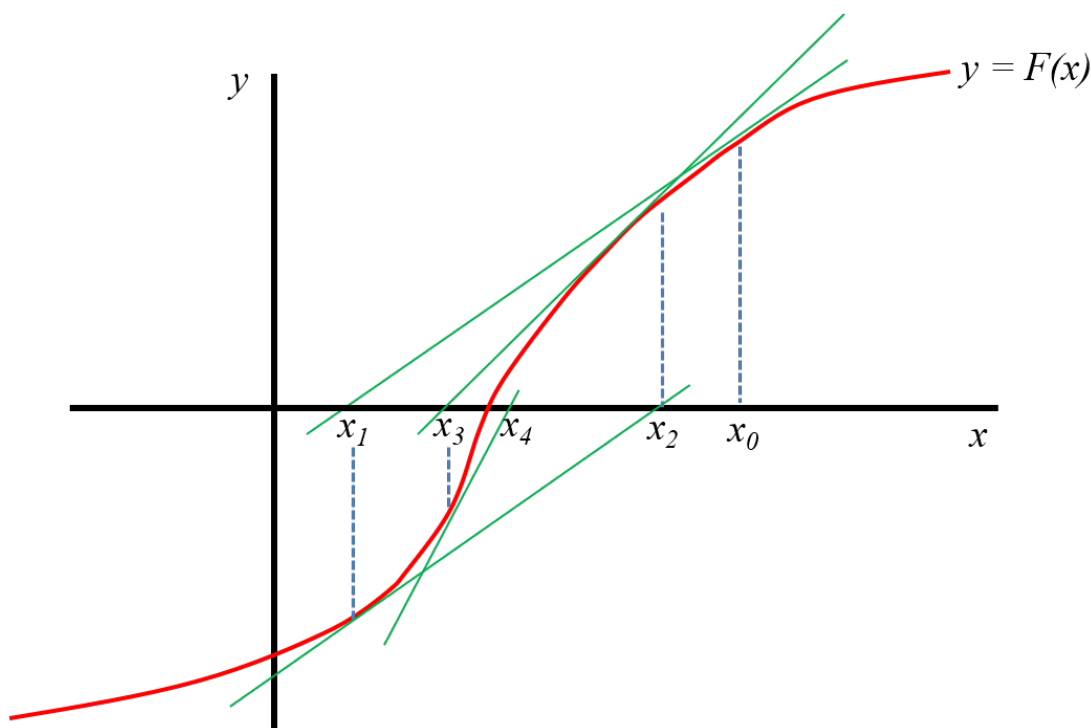
Divergencia monótona: $f' > 1$ Divergencia oscilante: $f' < -1$

2.3. Método de Newton-Raphson

- Si la función F y sus derivadas F' y F'' son continuas cerca de una raíz p , se pueden usar estas características de F para desarrollar algoritmos que produzcan sucesiones $\{x_k\}$ que converjan a p más rápidamente.
- El método de **Newton-Raphson** es uno de los más útiles y conocidos.
- Vamos a introducir este método a partir de su interpretación geométrica y su representación gráfica.
- **Recordar:** La *tangente* a una curva en un punto es una recta que toca a la curva sólo en dicho punto.
- Veamos el siguiente ejemplo donde el objetivo es hallar la raíz de la función F , es decir, el valor p tal que $F(p) = 0$.







- Supongamos que contamos con una aproximación inicial x_0 cercana a la raíz p .
- Definimos a x_1 como el punto de intersección del eje de las abscisas con la recta tangente a la curva F en x_0 .
- En el caso de la figura, se puede observar que x_1 está más cerca de p que x_0 .
- Ahora definimos a x_2 como el punto de intersección del eje de las abscisas con la recta tangente a la curva F en x_1 .
- Nuevamente, para el caso del ejemplo, podemos ver cómo x_2 está aún más cerca de p .
- Si continuamos repitiendo este proceso, esperamos encontrar un x_n que sea una buena aproximación para p .
- ¿Podemos expresar esto que observamos gráficamente a través de una fórmula?
- Es decir, a partir de x_0 , ¿podemos encontrar una fórmula para x_1 ?
- Sí, para eso hay prestarle atención a la pendiente m de la recta tangente en x_0 .
- Por un lado, sabemos que la pendiente de la recta tangente a la curva en un punto es igual a la derivada de la función en dicho punto:

$$m = F'(x_0) \quad (2.3)$$

- Pero además sabemos que para cualquier recta, la pendiente es igual a:

$$m = \frac{y_1 - y_0}{x_1 - x_0} \quad (2.4)$$

siendo (x_0, y_0) y (x_1, y_1) dos puntos distintos que pertenecen a la misma.

- Para expresar la pendiente de la recta tangente en x_0 , podemos tomar los puntos $(x_0, F(x_0))$ y $(x_1, 0)$ (el punto donde la tangente interseca al eje x), de manera que a partir de la fórmula anterior:

$$m = \frac{0 - F(x_0)}{x_1 - x_0} = -\frac{F(x_0)}{x_1 - x_0} \quad (2.5)$$

- Igualando (2.3) y (2.5) y despejando x_1 nos queda:

$$x_1 = x_0 - \frac{F(x_0)}{F'(x_0)} \quad (2.6)$$

- Si repetimos este pensamiento empezando desde x_1 con la recta tangente a F en el punto x_1 , vamos a encontrar que:

$$x_2 = x_1 - \frac{F(x_1)}{F'(x_1)} \quad (2.7)$$

- De esta manera hemos deducido una fórmula recursiva que nos permitirá hallar una aproximación para el verdadero valor de la raíz de F .
- Las ideas anteriores se formalizan analíticamente a través del siguiente teorema.
- En el mismo se deduce la fórmula recursiva a partir del desarrollo en serie de Taylor de la función F .

2.3.1. Teorema de Newton-Raphson

Supongamos que la función F es continua, con derivada segunda continua en el intervalo $[a; b]$, y que existe un número $p \in [a; b]$ tal que $F(p) = 0$. Si $F'(p) \neq 0$, entonces existe $\delta > 0$ tal que la sucesión $\{x_k\}_{k=0}^{\infty}$ definida por el proceso iterativo

$$x_k = x_{k-1} - \frac{F(x_{k-1})}{F'(x_{k-1})} \quad k = 1, 2, \dots$$

converge a p cualquiera sea la aproximación inicial $x_0 \in [p - \delta; p + \delta]$

2.3.2. Convergencia

Observación: para garantizar la convergencia, δ debe ser elegido tal que:

$$\frac{|F(x)F''(x)|}{[F'(x)]^2} < 1 \quad \forall x \in [p - \delta, p + \delta]$$

Esto significa que:

- x_0 debe estar suficientemente cerca a la raíz de $F(x) = 0$.
- $F''(x)$ no debe ser excesivamente grande.
- $F'(x)$ no debe estar muy próxima a cero.

2.3.3. Ejemplo:

- Evaluar si Newton-Raphson permite hallar la raíz positiva de $F(x) = x^2 - 3x + e^x - 2$, que no pudo ser hallada con Aproximaciones Sucesivas.

2.3.4. Ventajas y desventajas

Ventajas

- Aparece F en lugar de f .
- Converge más rápido que el método de las aproximaciones sucesivas.
- En algunos casos en que aproximaciones sucesivas diverge, N-R converge.
- Se puede adaptar para hallar raíces complejas.

Limitaciones

- Si x_0 está demasiado lejos de la raíz deseada, la sucesión $\{x_k\}$ puede converger a otra raíz (la pendiente $F'(x_0)$ es muy pequeña).
- Obtener la derivada primera de la función F puede ser difícil o imposible. En ese caso se podría aproximar $F'(x_{k-1})$ con:

$$F'(x_{k-1}) \approx \frac{F(x_{k-1} + h) - F(x_{k-1})}{h}$$

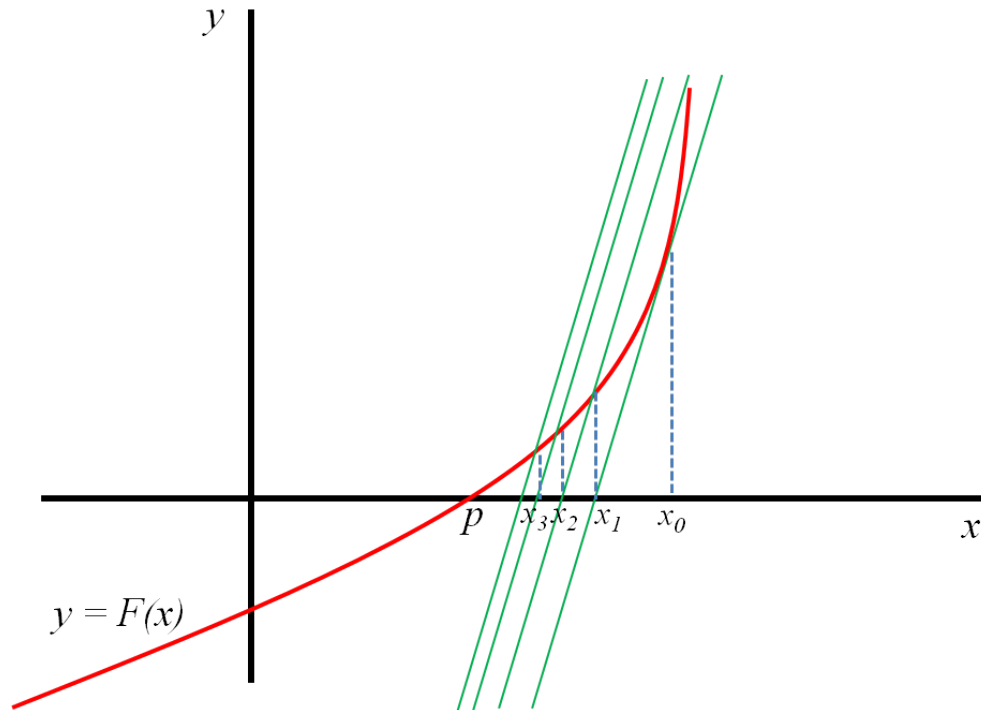
donde h es un valor pequeño, por ejemplo, $h = 0,001$.

2.4. Método de von Mises

- En el método de N-R, el denominador $F'(x_k)$ hace que geométricamente se pase de una aproximación a la siguiente por la tangente de la curva $y = F(x)$ en el punto correspondiente a la aproximación presente x_k .

- Esto puede producir problemas cuando se esté en puntos alejados de raíces y cerca de puntos donde el valor de $F'(x)$ sea cercano a 0 (tangentes cercanas a la horizontal).
- Para resolver este problema, von Mises sugirió sustituir $F'(x_k)$ en el denominador por $F'(x_0)$.
- Es decir, obtener geométricamente las siguientes aproximaciones por medio de rectas paralelas siempre a la primera tangente.
- La fórmula de recurrencia resultante es:

$$x_k = x_{k-1} - \frac{F(x_{k-1})}{F'(x_0)} \quad k = 1, 2, \dots$$



2.5. Método de Newton-Raphson de 2º Orden

- Otra modificación al método de N-R se deriva a partir de la utilización de un término más en el desarrollo por serie de Taylor de la función $F(x)$.
- Dada la existencia de las correspondientes derivadas, la fórmula de recurrencia resultante es:

$$x_k = x_{k-1} + \frac{F(x_{k-1})F'(x_{k-1})}{0.5F(x_{k-1})F''(x_{k-1}) - [F'(x_{k-1})]^2} \quad k = 1, 2, \dots$$

- El método de N-R de 2º orden llega más rápidamente a la raíz, aunque la fórmula es más difícil de obtener.

Unidad 3

Solución de Sistemas de Ecuaciones Lineales

3.1. Generalidades

- **Objetivo:** examinar los aspectos numéricos que se presentan al resolver sistemas de ecuaciones lineales de la forma:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n = b_2 \\ \vdots \\ a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nn}x_n = b_n \end{cases}$$

- n ecuaciones, n incógnitas: sistema de orden $n \times n$.
- Los coeficientes a_{ij} y los términos independientes b_i son reales fijos.

3.2. Repaso

- Representación matricial: $\mathbf{Ax} = \mathbf{b}$, de dimensión $n \times n$, $n \times 1$ y $n \times 1$, respectivamente.

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} \times \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}$$

- Llamamos **matriz ampliada** o **aumentada** a:

$$[\mathbf{A} \quad \mathbf{b}] = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} & b_1 \\ a_{21} & a_{22} & \cdots & a_{2n} & b_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} & b_n \end{bmatrix}$$

- Un sistema de ecuaciones lineales se clasifica en:
 - **Compatible determinado:** tiene una única solución
 - **Compatible indeterminado:** tiene infinitas soluciones
 - **Incompatible:** no existe solución

Teorema. Las siguientes condiciones son equivalentes:

- El sistema $\mathbf{Ax} = \mathbf{b}$ tiene solución única (es compatible determinado).
- La matriz \mathbf{A} es invertible (existe \mathbf{A}^{-1}).
- La matriz \mathbf{A} es no singular ($\det \mathbf{A} \neq 0$).
- El sistema $\mathbf{Ax} = \mathbf{0}$ tiene como única solución $\mathbf{x} = \mathbf{0}$.

- Dos sistemas de orden $n \times n$ son **equivalentes** si tienen el mismo conjunto de soluciones.
- Existen ciertas transformaciones sobre las ecuaciones de un sistema que no cambian el conjunto de soluciones (producen un sistema equivalente):
 - **Intercambio:** el orden de las ecuaciones puede cambiarse.
 - **Escalado:** multiplicar una ecuación por una constante no nula.
 - **Sustitución:** una ecuación puede ser reemplazada por una combinación lineal de las ecuaciones del sistema (*teorema fundamental de la equivalencia*)
- Realizar estas transformaciones sobre las ecuaciones es equivalente a realizar las mismas operaciones sobre las filas de la matriz aumentada.

3.3. Notación

- Notación que ayuda a expresar algoritmos con operaciones matriciales y facilita la escritura de los programas en IML o R.
- Dada una matriz \mathbf{Z} de dimensión $n \times m$, anotamos:
 - $z_{ij} = \mathbf{Z}[i, j]$: elemento en la fila i y columna j de la matriz \mathbf{Z}
 - $\mathbf{Z}[i,]$: vector fila de dimensión $1 \times m$ constituido por la i -ésima fila de la matriz \mathbf{Z}
 - $\mathbf{Z}[, j]$: vector columna $n \times 1$ constituido por la j -ésima columna de la matriz \mathbf{Z}
 - $\mathbf{Z}[i, k : l]$: matriz de dimensión $1 \times (l - k + 1)$ constituida con los elementos $z_{i,k}, z_{i,k+1}, \dots, z_{i,l}$ de la matriz \mathbf{Z} , $l \geq k$.
 - $\mathbf{Z}[c : d, k : l]$: matriz de dimensión $(d - c + 1) \times (l - k + 1)$ constituida por la submatriz que contiene las filas de \mathbf{Z} desde la c hasta la d y las columnas de \mathbf{Z} desde la k hasta la l , $d \geq c$, $l \geq k$.
- Dado un vector \mathbf{Z} de largo n , anotamos:
 - $\mathbf{Z}[i]$: elemento i -ésimo del vector \mathbf{Z}
 - $\mathbf{Z}[k : l]$: vector de largo $(l - k + 1)$ constituido con los elementos z_k, z_{k+1}, \dots, z_l del vector \mathbf{Z} , $l \geq k$.

3.4. Métodos de Resolución de Sistemas de Ecuaciones

- **Métodos exactos:** permiten obtener la solución del sistema de manera directa.
 - Método de Eliminación de Gauss
 - Método de Gauss-Jordan
- **Métodos aproximados:** utilizan algoritmos iterativos que calculan las solución del sistema por aproximaciones sucesivas.
 - Método de Jacobi
 - Método de Gauss-Seidel
- En muchas ocasiones los métodos aproximados permiten obtener un grado de exactitud superior al que se puede obtener empleando los denominados métodos exactos, debido fundamentalmente a los errores de truncamiento que se producen en el proceso.

3.5. Sistemas fáciles de resolver

Caso 1: La matriz \mathbf{A} es diagonal

$$\begin{bmatrix} a_{11} & 0 & \dots & 0 \\ 0 & a_{22} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & a_{nn} \end{bmatrix} \times \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}$$

- El sistema se reduce a n ecuaciones simples y la solución es:

$$\mathbf{x} = \begin{bmatrix} b_1/a_{11} \\ b_2/a_{22} \\ \vdots \\ b_n/a_{nn} \end{bmatrix}$$

Caso 2: La matrix \mathbf{A} es triangular superior:

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ 0 & a_{22} & a_{23} & \cdots & a_{2n} \\ 0 & 0 & a_{33} & \cdots & a_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & a_{nn} \end{bmatrix} \times \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}$$

- La solución de x_n es inmediata y a partir de ella se encuentran las restantes siguiendo el orden inverso x_{n-1}, \dots, x_1 , aplicando el algoritmo de **sustitución regresiva**:

$$x_n = \frac{b_n}{a_{nn}} \text{ y } x_k = \frac{b_k - \sum_{j=k+1}^n a_{kj}x_j}{a_{kk}} \quad k = n-1, n-2, \dots, 1$$

- Empleando la notación matricial vista, la suma en el algoritmo puede reescribirse con productos matriciales:

$$\mathbf{x}[n] = \frac{\mathbf{b}[n]}{\mathbf{A}[n, n]} \quad \mathbf{x}[k] = \frac{\mathbf{b}[k] - \mathbf{A}[k, k+1:n] \times \mathbf{x}[k+1:n]}{\mathbf{A}[k, k]} \\ k = n-1, n-2, \dots, 1$$

- Iniciando el vector solución con ceros ($\mathbf{x} = [0 \ 0 \ \dots \ 0]$), la expresión anterior se simplifica en:

$$\mathbf{x}[n] = \frac{\mathbf{b}[n]}{\mathbf{A}[n, n]} \quad \mathbf{x}[k] = \frac{\mathbf{b}[k] - \mathbf{A}[k, :] \times \mathbf{x}}{\mathbf{A}[k, k]} \quad k = n-1, n-2, \dots, 1$$

- Ver el algoritmo y programarlo.

Caso 3: La matrix \mathbf{A} es triangular inferior:

- Obtención de la solución análoga al caso anterior.

Observación

- En los casos anteriores asumimos $a_{kk} \neq 0 \forall k$. De lo contrario el sistema no tiene solución o tiene infinitas.
- Recordar que un sistema lineal $\mathbf{Ax} = \mathbf{b}$ tiene solución única si y sólo si $\det \mathbf{A} \neq 0$ y que si un elemento de la diagonal principal de una matriz triangular es cero, entonces $\det \mathbf{A} = 0$.

3.6. Eliminación gaussiana

- Es un método para resolver un sistema lineal general $\mathbf{Ax} = \mathbf{b}$ de n ecuaciones con n incógnitas.
- El objetivo es construir un sistema equivalente donde la matriz de coeficientes sea triangular superior para obtener las soluciones con el algoritmo de sustitución regresiva.
- El método consiste en ir eliminando incógnitas en las ecuaciones de manera sucesiva.
- Ejemplo: resolver el siguiente sistema

$$\begin{cases} x + 2y - z + 3t = -8 \\ 2x + 2z - t = 13 \\ -x + y + z - t = 8 \\ 3x + 3y - z + 2t = -1 \end{cases} \quad \mathbf{A} = \begin{bmatrix} 1 & 2 & -1 & 3 \\ 2 & 0 & 2 & -1 \\ -1 & 1 & 1 & -1 \\ 3 & 3 & -1 & 2 \end{bmatrix} \quad \mathbf{x} = \begin{bmatrix} x \\ y \\ z \\ t \end{bmatrix} \quad \mathbf{b} = \begin{bmatrix} -8 \\ 13 \\ 8 \\ -1 \end{bmatrix}$$

- Matriz aumentada:

$$[\mathbf{A} \quad \mathbf{b}] = \left[\begin{array}{cccc|c} 1 & 2 & -1 & 3 & -8 \\ 2 & 0 & 2 & -1 & 13 \\ -1 & 1 & 1 & -1 & 8 \\ 3 & 3 & -1 & 2 & -1 \end{array} \right]$$

- En el primer paso eliminamos la incógnita x en las ecuaciones 2, 3 y 4, buscando que queden ceros en toda la primera columna excepto en el elemento diagonal.
- La eliminación se hace empleando las transformaciones que producen un sistema equivalente.
- Observando con detenimiento, esto se puede lograr realizando los siguientes reemplazos:

$$\begin{aligned}\text{Fila 2} &= \text{Fila 2} - 2 \times \text{Fila 1} \\ \text{Fila 3} &= \text{Fila 3} - (-1) \times \text{Fila 1} \\ \text{Fila 4} &= \text{Fila 4} - 3 \times \text{Fila 1}\end{aligned}$$

- A la Fila 1 se le dice **fila pivote** y al elemento $a_{11} = 1$, **pivote**.
- A los valores 2, -1 y 3 que multiplican a la fila pivote en los reemplazos se les dice **multiplicadores**.
- Los multiplicadores se simbolizan y obtienen con: $m_{r1} = a_{r1}/a_{11}$, $r = 2, 3, 4$.
- De esta forma, los reemplazos realizados se generalizan como: Fila $r = \text{Fila } r - m_{r1} \times \text{Fila 1}$.
- El resultado de los reemplazos anteriores es:

$$\begin{array}{l} \text{pivote} \rightarrow \\ m_{21} = 2 \\ m_{31} = -1 \\ m_{41} = 3 \end{array} \left[\begin{array}{cccc|c} 1 & 2 & -1 & 3 & -8 \\ 2 & 0 & 2 & -1 & 13 \\ -1 & 1 & 1 & -1 & 8 \\ 3 & 3 & -1 & 2 & -1 \end{array} \right] \Rightarrow \left[\begin{array}{cccc|c} 1 & 2 & -1 & 3 & -8 \\ 0 & -4 & 4 & -7 & 29 \\ 0 & 3 & 0 & 2 & 0 \\ 0 & -3 & 2 & -7 & 23 \end{array} \right]$$

- En el segundo paso, eliminamos la incógnita y en las ecuaciones 3 y 4.
- La fila pivote pasa a ser la segunda y el pivote es $a_{22} = -4$.
- Los multiplicadores son $m_{r2} = a_{r2}/a_{22}$, $r = 3, 4$, dando lugar a los reemplazos:

$$\begin{aligned}\text{Fila 3} &= \text{Fila 3} - (-3/4) \times \text{Fila 2} \\ \text{Fila 4} &= \text{Fila 4} - 3/4 \times \text{Fila 2}\end{aligned}$$

- El resultado es:

$$\begin{array}{l} \text{pivote} \rightarrow \\ m_{32} = -3/4 \\ m_{42} = 3/4 \end{array} \left[\begin{array}{cccc|c} 1 & 2 & -1 & 3 & -8 \\ 0 & -4 & 4 & -7 & 29 \\ 0 & 3 & 0 & 2 & 0 \\ 0 & -3 & 2 & -7 & 23 \end{array} \right] \Rightarrow \left[\begin{array}{cccc|c} 1 & 2 & -1 & 3 & -8 \\ 0 & -4 & 4 & -7 & 29 \\ 0 & 0 & -12 & 13 & -87 \\ 0 & 0 & 4 & 7 & -5 \end{array} \right]$$

- Finalmente, eliminamos la incógnita z en la última ecuación.
- La fila pivote es la 3ª y el pivote es $a_{33} = -12$.
- El multiplicador es $m_{43} = a_{43}/a_{33} = -4/12$.
- El reemplazo a realizar es Fila 4 = Fila 4 - $(-4/12) \times$ Fila 3.
- El resultado es:

$$\begin{array}{l} \text{pivote} \rightarrow \\ m_{43} = -4/12 \end{array} \left[\begin{array}{cccc|c} 1 & 2 & -1 & 3 & -8 \\ 0 & -4 & 4 & -7 & 29 \\ 0 & 0 & -12 & 13 & -87 \\ 0 & 0 & 4 & 7 & -5 \end{array} \right] \Rightarrow \left[\begin{array}{cccc|c} 1 & 2 & -1 & 3 & -8 \\ 0 & -4 & 4 & -7 & 29 \\ 0 & 0 & -12 & 13 & -87 \\ 0 & 0 & 0 & -136 & 408 \end{array} \right]$$

- Hemos llegado a un sistema equivalente cuya matriz de coeficientes es triangular superior, en el que aplicamos el algoritmo de sustitución regresiva:

$$\begin{cases} x + 2y - z - 3t = -8 \\ -4y + 4z - 7t = 29 \\ -12z + 13t = -87 \\ -136t = 408 \end{cases} \Rightarrow \begin{cases} x = 1 \\ y = 2 \\ z = 4 \\ t = -3 \end{cases}$$

3.6.1. Eliminación gaussiana con pivoteo trivial

- **Observación:** en el algoritmo visto es necesario que los pivotes $a_{qq} \neq 0 \forall q$.
- Si en uno de los pasos encontramos un $a_{qq} = 0$, debemos intercambiar la q -ésima fila por una cualquiera de las siguientes, por ejemplo la fila k , en la que $a_{kq} \neq 0, k > q$.
- Esta estrategia para hallar un pivote no nulo se llama **pivoteo trivial**.
- Ejemplo propuesto:

$$\begin{cases} x - 2y + z = -4 \\ -2x + 4y - 3z = 3 \\ x - 3y - 4z = -1 \end{cases}$$

3.7. Estrategias de pivoteo para reducir los errores

- Como ya sabemos, dado que las computadoras usan una aritmética cuya precisión está fijada de antemano, es posible que cada vez que se realice una operación aritmética se introduzca un pequeño error.
- En la resolución de ecuaciones por eliminación gaussiana, un adecuado reordenamiento de las filas en el momento indicado puede reducir notablemente los errores cometidos.
- Por ejemplo, se puede mostrar cómo buscar un multiplicador de menor magnitud (es decir, un pivote de mayor magnitud) mejora la precisión de los resultados.
- Por eso, existen estrategias de pivoteo que no solamente hacen intercambio de filas cuando se tiene un pivote nulo, si no también cuando alguna de las filas posteriores tiene un pivote de mayor valor absoluto.

3.7.1. Pivoteo parcial

- Para reducir la propagación de los errores de redondeo, antes de comenzar una nueva ronda de reemplazos con el pivote a_{qq} se evalúa si debajo en la misma columna hay algún elemento con mayor valor absoluto y en ese caso se intercambian las respectivas filas.
- Es decir, se busca si existe r tal que $|a_{rq}| > |a_{qq}|$, $r > q$ para luego intercambiar las filas q y r .
- Este proceso suele conservar las magnitudes relativas de los elementos de la matriz triangular superior en el mismo orden que las de los coeficientes de la matriz original.

3.7.2. Pivoteo parcial escalado

- Reduce aún más los efectos de la propagación de los errores.
- Se elige el elemento de la columna q -ésima, en o por debajo de la diagonal principal, que tiene mayor tamaño relativo con respecto al resto de los elementos de su fila.
- **Paso 1:** buscar el máximo valor absoluto en cada fila:

$$s_r = \max\{|a_{rq}|, |a_{r,q+1}|, \dots, |a_{rn}|\} \quad r = q, q+1, \dots, n$$

- **Paso 2:** elegir como fila pivote a la que tenga el mayor valor de $\frac{|a_{rq}|}{s_r}$, $r = q, q+1, \dots, n$.
- **Paso 3:** intercambiar la fila q con la fila hallada en el paso 2.

3.8. Método de eliminación de Gauss-Jordan

- Ya vimos que el método de Gauss transforma la matriz de coeficientes en una matriz triangular superior.
- El método de Gauss-Jordan continúa el proceso de transformación hasta obtener la matriz identidad.
- **Retomar Ejemplo Diapo 15:** cuando aplicamos eliminación de Gauss llegamos a la siguiente matriz triangular.

$$\left[\begin{array}{cccc|c} 1 & 2 & -1 & 3 & -8 \\ 0 & -4 & 4 & -7 & 29 \\ 0 & 0 & -12 & 13 & -87 \\ 0 & 0 & 0 & -136 & 408 \end{array} \right]$$

- Con el método de sustitución regresiva, llegamos a encontrar $x = 1$, $y = 2$, $z = 4$ y $t = -3$.

- En lugar de aplicar sustitución regresiva, seguimos operando por fila hasta lograr una matriz diagonal:

$$\begin{array}{lcl} \text{Fila 2} & = & \text{Fila 2}/(-4) \\ \text{Fila 3} & = & \text{Fila 3}/(-12) \\ \text{Fila 4} & = & \text{Fila 4}/(-136) \end{array} \Rightarrow \left[\begin{array}{cccc|c} 1 & 2 & -1 & 3 & -8 \\ 0 & 1 & -1 & 7/4 & -29/4 \\ 0 & 0 & 1 & -13/12 & 87/12 \\ 0 & 0 & 0 & 1 & -3 \end{array} \right]$$

$$\begin{array}{lcl} \text{Fila 1} & = & \text{Fila 1} - 3 \text{ Fila 4} \\ \text{Fila 2} & = & \text{Fila 2} - 7/4 \text{ Fila 4} \\ \text{Fila 3} & = & \text{Fila 3} + 13/12 \text{ Fila 4} \end{array} \Rightarrow \left[\begin{array}{cccc|c} 1 & 2 & -1 & 0 & 1 \\ 0 & 1 & -1 & 0 & -2 \\ 0 & 0 & 1 & 0 & 4 \\ 0 & 0 & 0 & 1 & -3 \end{array} \right]$$

$$\begin{array}{lcl} \text{Fila 1} & = & \text{Fila 1} + \text{Fila 3} \\ \text{Fila 2} & = & \text{Fila 2} + \text{Fila 3} \end{array} \Rightarrow \left[\begin{array}{cccc|c} 1 & 2 & 0 & 0 & 5 \\ 0 & 1 & 0 & 0 & 2 \\ 0 & 0 & 1 & 0 & 4 \\ 0 & 0 & 0 & 1 & -3 \end{array} \right]$$

$$\text{Fila 1} = \text{Fila 1} - 2 \text{ Fila 2} \Rightarrow \left[\begin{array}{cccc|c} 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 2 \\ 0 & 0 & 1 & 0 & 4 \\ 0 & 0 & 0 & 1 & -3 \end{array} \right]$$

$$\Rightarrow \begin{cases} x = 1 \\ y = 2 \\ z = 4 \\ t = -3 \end{cases}$$

- La ventaja de este método es que permite también hallar fácilmente la matriz inversa de \mathbf{A} .
- Para esto hay que concatenar a la derecha de la matriz aumentada una matriz identidad de orden n . Cuando en la submatriz izquierda se llega a la matriz identidad, en el centro habrá quedado el vector solución y a su derecha la matriz inversa.
- Ejemplo. Resolver el siguiente sistema y hallar la inversa de la matriz de coeficientes:

$$\begin{cases} x - y + z = -4 \\ 5x - 4y + 3z = -12 \\ 2x + y + z = 11 \end{cases}$$

$$\left[\begin{array}{ccc|ccc} 1 & -1 & 1 & -4 & 1 & 0 & 0 \\ 5 & -4 & 3 & -12 & 0 & 1 & 0 \\ 2 & 1 & 1 & 11 & 0 & 0 & 1 \end{array} \right]$$

$$F2 = F2 - 5F1$$

$$F3 = F3 - 2F1$$

$$F1 = F1 + F2$$

$$F3 = F3 - 3F2$$

$$F1 = F1 + 1/5 F3$$

$$F2 = F2 + 2/5 F3$$

$$F3 = F3/5$$

$$\left[\begin{array}{ccc|ccc} 1 & 0 & 0 & 3 & -7/5 & 2/5 & 1/5 \\ 0 & 1 & 0 & 6 & 1/5 & -1/5 & 2/5 \\ 0 & 0 & 1 & -1 & 13/5 & -3/2 & 1/5 \end{array} \right]$$

- ¿Por qué este procedimiento nos devuelva la inversa de la matriz de coeficientes?
- Recordar que la inversa de \mathbf{A} es aquella matriz \mathbf{A}^{-1} que verifica $\mathbf{A}\mathbf{A}^{-1} = \mathbf{A}^{-1}\mathbf{A} = \mathbf{I}$
- Entonces queremos hallar la matriz \mathbf{A}^{-1} :

$$\mathbf{A}^{-1} = \begin{bmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{31} & x_{32} & x_{33} \end{bmatrix}$$

tal que:

$$\begin{bmatrix} 1 & -1 & 1 \\ 5 & -4 & 3 \\ 2 & 1 & 1 \end{bmatrix} \times \begin{bmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{31} & x_{32} & x_{33} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

- Esto da lugar a tres sistemas de ecuaciones con la misma matriz de coeficientes:

$$\begin{aligned} \Rightarrow \begin{bmatrix} x_{11} - x_{21} + x_{31} & x_{12} - x_{22} + x_{32} & x_{13} - x_{23} + x_{33} \\ 5x_{11} - 4x_{21} + 3x_{31} & 5x_{12} - 4x_{22} + 3x_{32} & 5x_{13} - 4x_{23} + 3x_{33} \\ 2x_{11} + x_{21} + x_{31} & 2x_{12} - x_{22} + x_{32} & 2x_{13} - x_{23} + x_{33} \end{bmatrix} = \\ = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \end{aligned}$$

$$\begin{cases} x_{11} - x_{21} + x_{31} = 1 \\ 5x_{11} - 4x_{21} + 3x_{31} = 0 \\ 2x_{11} + x_{21} + x_{31} = 0 \end{cases}$$

$$\begin{cases} x_{12} - x_{22} + x_{32} = 0 \\ 5x_{12} - 4x_{22} + 3x_{32} = 1 \\ 2x_{12} + x_{22} + x_{32} = 0 \end{cases}$$

$$\begin{cases} x_{13} - x_{23} + x_{33} = 0 \\ 5x_{13} - 4x_{23} + 3x_{33} = 0 \\ 2x_{13} + x_{23} + x_{33} = 1 \end{cases}$$

- Los resolvemos simultáneamente al agregar la matriz identidad de orden n a la derecha de la matriz aumentada del sistema original.

3.9. Métodos Aproximados o Iterativos

- **Objetivo:** extender a espacios de dimensión mayor que uno las ideas de los métodos iterativos vistos en la unidad 2 para resolver sistemas de ecuaciones lineales.
- Veremos:
 - Método de Jacobi
 - Método de Gauss-Seidel

3.9.1. Método de Jacobi

- Consideremos el sistema:

$$\begin{cases} 4x - y + z = 7 \\ 4x - 8y + z = -21 \\ -2x + y + 5z = 15 \end{cases} \Rightarrow \begin{cases} x = (7 + y - z)/4 \\ y = (21 + 4x + z)/8 \\ z = (15 + 2x - y)/5 \end{cases}$$

- Despejar una incógnita en cada ecuación provee expresiones que sugieren la idea de un proceso iterativo.
- Dado un vector de valores iniciales $\mathbf{x}_0 = (x_0, y_0, z_0)$, operar con la siguiente fórmula de recurrencia hasta la convergencia:

$$\begin{cases} x_{k+1} = (7 + y_k - z_k)/4 \\ y_{k+1} = (21 + 4x_k + z_k)/8 \\ z_{k+1} = (15 + 2x_k - y_k)/5 \end{cases}$$

- Por ejemplo, tomando el valor inicial $(1, 2, 2)$, el proceso converge hacia la solución exacta del sistema $(2, 4, 3)$.
- Usando 4 posiciones decimales con redondeo luego de la coma:

k	x_k	y_k	z_k
0	1	2	2
1	1.7500	3.3750	3.0000
2	1.8438	3.8750	3.0250
3	1.9625	3.9250	2.9625
4	1.9906	3.9766	3.0000
5	1.9942	3.9953	3.0009
6	1.9986	3.9972	2.9986
7	1.9997	3.9991	3.0000
8	1.9998	3.9999	3.0001
9	2.0000	3.9999	2.9999
10	2.0000	4.0000	3.0000

- *Observación:* no siempre este método converge. Es sensible al ordenamiento de las ecuaciones dentro del sistema.
- Ejemplo: tomamos el mismo sistema de antes pero intercambiamos las filas 1 y 3:

$$\begin{aligned} \begin{cases} 4x - y + z = 7 \\ 4x - 8y + z = -21 \\ -2x + y + 5z = 15 \end{cases} &\Rightarrow \begin{cases} -2x + y + 5z = 15 \\ 4x - 8y + z = -21 \\ 4x - y + z = 7 \end{cases} \\ \Rightarrow \begin{cases} x = (-15 + y + 5z)/2 \\ y = (21 + 4x + z)/8 \\ z = 7 - 4x + y \end{cases} &\Rightarrow \begin{cases} x_{k+1} = (-15 + y_k + 5z_k)/2 \\ y_{k+1} = (21 + 4x_k + z_k)/8 \\ z_{k+1} = 7 - 4x_k + y_k \end{cases} \end{aligned}$$

- Tomando el mismo valor inicial $(1, 2, 2)$, esta vez el proceso diverge:

k	x_k	y_k	z_k
0	1	2	2
1	-1.5000	3.3750	5.0000
2	6.6875	2.5000	16.3750
3	34.6875	8.0156	-17.2500
4	-46.6172	17.8125	-123.7344
5	-307.9298	-36.1504	211.2813
6	502.6281	-124.9297	1202.5688
7	2936.4572	404.2602	-2128.4421
8	-5126.4752	1204.7983	-11334.5686
9	-27741.5224	-3977.4337	21717.6991
10	52298.0309	-11153.4238	106995.6559

3.9.2. Método de Gauss-Seidel

- Toma la misma idea que Jacobi, pero con una pequeña modificación para acelerar la convergencia. Retomando el ejemplo inicial:

$$\begin{cases} 4x - y + z = 7 \\ 4x - 8y + z = -21 \\ -2x + y + 5z = 15 \end{cases} \implies \begin{cases} x = (7 + y - z)/4 \\ y = (21 + 4x + z)/8 \\ z = (15 + 2x - y)/5 \end{cases}$$

Jacobi:

$$\begin{cases} x_{k+1} = (7 + y_k - z_k)/4 \\ y_{k+1} = (21 + 4x_k + z_k)/8 \\ z_{k+1} = (15 + 2x_k - y_k)/5 \end{cases}$$

Gauss-Seidel:

$$\begin{cases} x_{k+1} = (7 + y_k - z_k)/4 \\ y_{k+1} = (21 + 4x_{k+1} + z_k)/8 \\ z_{k+1} = (15 + 2x_{k+1} - y_{k+1})/5 \end{cases}$$

- La diferencia está en que apenas calcula un nuevo valor de las incógnitas, Gauss-Seidel lo usa inmediatamente en el cálculo de las restantes, en lugar de esperar a la próxima ronda.
- Tomando otra vez el valor inicial $(1, 2, 2)$ y operando con 4 posiciones decimales con redondeo luego de la coma:

Jacobi				Gauss-Seidel			
k	x_k	y_k	z_k	k	x_k	y_k	z_k
0	1	2	2	0	1	2	2
1	1.7500	3.3750	3.0000	1	1.7500	3.7500	2.9500
2	1.8438	3.8750	3.0250	2	1.9500	3.9688	2.9862
3	1.9625	3.9250	2.9625	3	1.9957	3.9961	2.9991
4	1.9906	3.9766	3.0000	4	1.9993	3.9995	2.9998
5	1.9942	3.9953	3.0009	5	1.9999	3.9999	3.0000
6	1.9986	3.9972	2.9986	6	2.0000	4.0000	3.0000
7	1.9997	3.9991	3.0000				
8	1.9998	3.9999	3.0001				
9	2.0000	3.9999	2.9999				
10	2.0000	4.0000	3.0000				

3.9.3. Generalización

- Ahora que sabemos qué hace cada método y en qué se diferencian vamos a:
 - Generalizar las fórmulas recursivas
 - Escribirlas matricialmente
 - Evaluar la convergencia
 - Ver los programas
- Tenemos un sistema de n ecuaciones lineales con n incógnitas y despejamos una variable en cada ecuación:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n = b_2 \\ \vdots \\ a_{j1}x_1 + a_{j2}x_2 + \cdots + a_{jn}x_n = b_j \\ \vdots \\ a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nn}x_n = b_n \end{cases}$$

$$\Rightarrow \begin{cases} x_1 = \frac{b_1 - a_{12}x_2 - \dots - a_{1n}x_n}{a_{11}} \\ x_2 = \frac{b_2 - a_{21}x_1 - a_{23}x_3 - \dots - a_{2n}x_n}{a_{22}} \\ \vdots \\ x_j = \frac{b_j - a_{j1}x_1 - \dots - a_{j(j-1)}x_{j-1} - a_{j(j+1)}x_{j+1} - \dots - a_{jn}x_n}{a_{jj}} \\ \vdots \\ x_n = \frac{b_n - a_{n1}x_1 - \dots - a_{n(n-1)}x_{n-1}}{a_{nn}} \end{cases}$$

3.9.3.1. Generalización del Método de Jacobi

- Con los supra índices indicando el número de iteración, la fórmula de recurrencia es:

$$\begin{cases} x_1^{(k+1)} = \frac{b_1 - a_{12}x_2^{(k)} - \dots - a_{1n}x_n^{(k)}}{a_{11}} \\ x_2^{(k+1)} = \frac{b_2 - a_{21}x_1^{(k)} - a_{23}x_3^{(k)} - \dots - a_{2n}x_n^{(k)}}{a_{22}} \\ \vdots \\ x_j^{(k+1)} = \frac{b_j - a_{j1}x_1^{(k)} - \dots - a_{j(j-1)}x_{j-1}^{(k)} - a_{j(j+1)}x_{j+1}^{(k)} - \dots - a_{jn}x_n^{(k)}}{a_{jj}} \\ \vdots \\ x_n^{(k+1)} = \frac{b_n - a_{n1}x_1^{(k)} - \dots - a_{n(n-1)}x_{n-1}^{(k)}}{a_{nn}} \end{cases}$$

$$\Rightarrow x_j^{(k+1)} = \frac{b_j - a_{j1}x_1^{(k)} - \dots - a_{j(j-1)}x_{j-1}^{(k)} - a_{j(j+1)}x_{j+1}^{(k)} - \dots - a_{jn}x_n^{(k)}}{a_{jj}}$$

$$k = 1, 2, \dots, n$$

3.9.3.2. Generalización del Método de Gauss-Seidel

- Con los supra índices indicando el número de iteración, la fórmula de recurrencia es:

$$\begin{cases} x_1^{(k+1)} = \frac{b_1 - a_{12}x_2^{(k)} - \dots - a_{1n}x_n^{(k)}}{a_{11}} \\ x_2^{(k+1)} = \frac{b_2 - a_{21}x_1^{(k+1)} - a_{23}x_3^{(k)} - \dots - a_{2n}x_n^{(k)}}{a_{22}} \\ \vdots \\ x_j^{(k+1)} = \frac{b_j - a_{j1}x_1^{(k+1)} - \dots - a_{j(j-1)}x_{j-1}^{(k+1)} - a_{j(j+1)}x_{j+1}^{(k)} - \dots - a_{jn}x_n^{(k)}}{a_{jj}} \\ \vdots \\ x_n^{(k+1)} = \frac{b_n - a_{n1}x_1^{(k+1)} - \dots - a_{n(n-1)}x_{n-1}^{(k+1)}}{a_{nn}} \end{cases}$$

$$\Rightarrow x_j^{(k+1)} = \frac{b_j - a_{j1}x_1^{(k+1)} - \dots - a_{j(j-1)}x_{j-1}^{(k+1)} - a_{j(j+1)}x_{j+1}^{(k)} - \dots - a_{jn}x_n^{(k)}}{a_{jj}}$$

$$k = 1, 2, \dots, n$$

3.9.3.3. Expresión matricial para el Método de Jacobi

- Si descomponemos a la matriz \mathbf{A} como $\mathbf{A} = \mathbf{D} + \mathbf{R}$, donde \mathbf{D} es la matriz diagonal formada con la diagonal de \mathbf{A} y \mathbf{R} es igual a \mathbf{A} excepto en la diagonal donde posee todos ceros, tenemos:

$$\begin{aligned} \mathbf{Ax} &= \mathbf{b} \\ (\mathbf{D} + \mathbf{R})\mathbf{x} &= \mathbf{b} \\ \mathbf{Dx} &= \mathbf{b} - \mathbf{Rx} \\ \mathbf{x} &= \mathbf{D}^{-1}(\mathbf{b} - \mathbf{Rx}) \end{aligned}$$

- Esto da lugar a la siguiente fórmula de recurrencia, que es equivalente a las vistas anteriormente, y facilita la programación del método:

$$\mathbf{x}^{(k+1)} = \mathbf{D}^{-1}(\mathbf{b} - \mathbf{R}\mathbf{x}^{(k)})$$

- Requisito: ningún elemento en la diagonal de \mathbf{A} es cero.

3.9.3.4. Expresión matricial para el Método de Gauss-Seidel

- Si descomponemos a la matriz \mathbf{A} como $\mathbf{A} = \mathbf{L} + \mathbf{U}$, donde \mathbf{L} es una matriz triangular inferior (incluyendo la diagonal de \mathbf{A}) y \mathbf{U} es una matriz triangular superior (con ceros en la diagonal), tenemos:

$$\begin{aligned}\mathbf{A}\mathbf{x} &= \mathbf{b} \\ (\mathbf{L} + \mathbf{U})\mathbf{x} &= \mathbf{b} \\ \mathbf{L}\mathbf{x} &= \mathbf{b} - \mathbf{U}\mathbf{x} \\ \mathbf{x} &= \mathbf{L}^{-1}(\mathbf{b} - \mathbf{U}\mathbf{x})\end{aligned}$$

- Esto da lugar a la siguiente fórmula de recurrencia, que es equivalente a las vistas anteriormente, y facilita la programación del método:

$$\mathbf{x}^{(k+1)} = \mathbf{L}^{-1}(\mathbf{b} - \mathbf{U}\mathbf{x}^{(k)})$$

- Requisito: ningún elemento en la diagonal de \mathbf{A} es cero.

3.9.4. Convergencia

Definición:

Se dice que una matriz \mathbf{A} de orden $n \times n$ es **diagonal estrictamente dominante** cuando el elemento diagonal es mayor a la suma del resto de los elementos de su fila en valor absoluto:

$$|a_{kk}| > \sum_{\substack{j=0 \\ j \neq k}}^n |a_{kj}| \quad k = 1, 2, \dots, n$$

Teorema

Si la matriz \mathbf{A} es diagonal estrictamente dominante, entonces el sistema lineal $\mathbf{A}\mathbf{x} = \mathbf{b}$ tiene solución única y los procesos iterativos de Jacobi y de Gauss-Seidel convergen hacia la misma cualquiera sea el vector de partida \mathbf{x}_0 .

- Es una condición suficiente pero no necesaria.

Criterios para la convergencia

- **Norma L1 (distancia Manhattan):**

$$\|\mathbf{x}_{k+1} - \mathbf{x}_k\|_1 = \sum_{j=1}^n |x_j^{(k+1)} - x_j^{(k)}| < \epsilon$$

- **Norma L2 (norma euclídea):**

$$\|\mathbf{x}_{k+1} - \mathbf{x}_k\|_2 = \sqrt{(\mathbf{x}_{k+1} - \mathbf{x}_k)'(\mathbf{x}_{k+1} - \mathbf{x}_k)} = \sqrt{\sum_{j=1}^n (x_j^{(k+1)} - x_j^{(k)})^2} < \epsilon$$

- **Norma L_∞ (máxima diferencia):**

$$\|\mathbf{x}_{k+1} - \mathbf{x}_k\|_\infty = \max_j |x_j^{(k+1)} - x_j^{(k)}| < \epsilon$$

- Establecer un número máximo de iteraciones

3.9.4.1. Ventajas de Gauss-Seidel

- Converge más rápidamente.
- Puede converger cuando Jacobi no lo hace (aunque \mathbf{A} no sea diagonal estrictamente dominante, por ejemplo si \mathbf{A} es semidefinida positiva).
- Pero puede haber casos en los que Jacobi converge y Gauss-Seidel no.

Unidad 4

Aproximación Polinomial: interpolación y extrapolación

4.1. Generalidades

- En matemática estudiamos funciones de la forma $y = f(x)$, donde se conoce la expresión matemática que define a $f(x)$, para determinar sus derivadas y calcular integrales definidas.
- Sin embargo, muchas veces en aplicaciones prácticas se requiere trabajar con funciones cuya derivación e integración presenta dificultades porque:
 - no son funciones elementales (polinomios, expresiones racionales, funciones trigonométricas, exponenciales o combinaciones sencillas de estas);
 - directamente se desconoce su expresión analítica (ejemplo: datos obtenidos experimentalmente).
- En estos casos, se suele contar con una tabla de valores compuestas por puntos (x_i, y_i) a partir de la cual se desea:
 - aproximar $f(x)$ en abscisas que no están tabuladas (valores de x no dados), es decir, interpolar
 - integrar $f(x)$ en un intervalo determinado
 - derivar $f(x)$
- En esta unidad estudiaremos métodos numéricos que cumplen con estos objetivos que consisten en sustituir la función complicada o que está determinada tabularmente, por una función polinomial que se aproxime a los puntos disponibles.

4.2. Diferencias finitas

- Antes de ver el primer método veremos cómo crear una tabla de diferencia que utilizaremos más adelante.
- Se tiene una función $y = f(x)$ definida en forma tabular con saltos equiespaciados en x :

x_k	$y_k = f(x_k)$
x_0	y_0
$x_1 = x_0 + h$	y_1
$x_2 = x_0 + 2h$	y_2
\vdots	\vdots
$x_{n-1} = x_0 + (n-1)h$	y_{n-1}
$x_n = x_0 + nh$	y_n

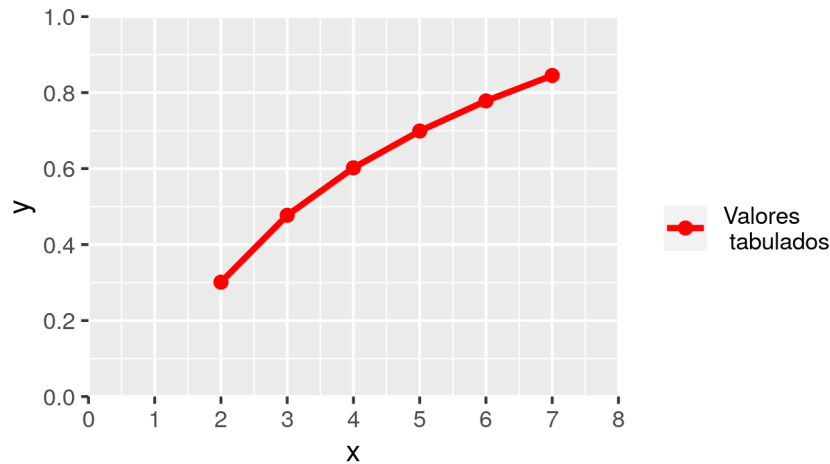
$\Delta y_k = y_{k+1} - y_k$	$\Delta^2 y_k = \Delta y_{k+1} - \Delta y_k$	$\Delta^3 y_k = \Delta^2 y_{k+1} - \Delta^2 y_k$...
$\Delta y_0 = y_1 - y_0$	$\Delta^2 y_0 = \Delta y_1 - \Delta y_0$	$\Delta^3 y_0 = \Delta^2 y_1 - \Delta^2 y_0$...
$\Delta y_1 = y_2 - y_1$	$\Delta^2 y_1 = \Delta y_2 - \Delta y_1$	$\Delta^3 y_1 = \Delta^2 y_2 - \Delta^2 y_1$...
$\Delta y_2 = y_3 - y_2$	$\Delta^2 y_2 = \Delta y_3 - \Delta y_2$	$\Delta^3 y_2 = \Delta^2 y_3 - \Delta^2 y_2$...

$\Delta y_k = y_{k+1} - y_k$	$\Delta^2 y_k = \Delta y_{k+1} - \Delta y_k$	$\Delta^3 y_k = \Delta^2 y_{k+1} - \Delta^2 y_k$...
\vdots	\vdots	\vdots	\ddots
$\Delta y_{n-1} = y_n - y_{n-1}$	-	-	...
-	-	-	...

...	$\Delta^{n-1} y_k = \Delta^{n-2} y_{k+1} - \Delta^{n-2} y_k$	$\Delta^n y_k = \Delta^{n-1} y_{k+1} - \Delta^{n-1} y_k$
...	$\Delta^{n-1} y_0 = \Delta^{n-2} y_1 - \Delta^{n-2} y_0$	$\Delta^n y_0 = \Delta^{n-1} y_1 - \Delta^{n-1} y_0$
...	$\Delta^{n-1} y_1 = \Delta^{n-2} y_2 - \Delta^{n-2} y_1$	-
...	-	-
\ddots	\vdots	\vdots
...	-	-
...	-	-

Ejemplo 1. Siendo $y = f(x)$ una función desconocida de la cual se tienen los valores tabulados (x_k, y_k) que se presentan a continuación, junto con una representación gráfica de los mismos:

k	x_k	y_k
0	2	0,3010
1	3	0,4771
2	4	0,6021
3	5	0,6990
4	6	0,7781
5	7	0,8451



- La correspondiente tabla de diferencias finitas es:

k	x_k	y_k	Δy_k	$\Delta^2 y_k$	$\Delta^3 y_k$	$\Delta^4 y_k$	$\Delta^5 y_k$
0	2	0,3010	0,1761	-0,0511	0,0230	-0,0127	0,0081
1	3	0,4771	0,1250	-0,0281	0,0103	-0,0046	-
2	4	0,6021	0,0969	-0,0178	0,0057	-	-
3	5	0,6990	0,0791	-0,0121	-	-	-
4	6	0,7781	0,0670	-	-	-	-
5	7	0,8451	-	-	-	-	-

Observaciones:

- Si tenemos $n + 1$ puntos podemos calcular n columnas de diferencias hacia adelante.

- Si la función $f(x)$ que dio lugar a la tabla es un polinomio de orden q , entonces la columna para la diferencia de orden q es constante y las siguientes columnas son todas nulas.
- Por lo tanto, si en el proceso de obtención de las diferencias sucesivas de una función, las diferencias de orden q se vuelven constantes (o aproximadamente constantes), sabemos que los datos provienen exactamente (o muy aproximadamente) de un polinomio de orden q .
- Errores de redondeo podrían hacer que a pesar de que los datos provengan de un polinomio, no encontremos diferencias constantes.

4.3. Interpolación de Newton para incrementos constantes

- Ahora intentaremos resolver el problema de aproximar la función $f(x)$ para valores de x que no forman parte de la tabla.
- Operando con los valores de la tabla de diferencia, se comprueba que los valores tabulados y_k se pueden escribir como:

$$\begin{aligned} y_k &= \sum_{j=0}^k \binom{k}{j} \Delta^j y_0 \\ &= y_0 + k\Delta y_0 + \frac{k(k-1)}{2!} \Delta^2 y_0 + \frac{k(k-1)(k-2)}{3!} \Delta^3 y_0 + \dots \end{aligned}$$

donde $\Delta^0 y_0 = y_0$.

- Por ejemplo:

$$y_2 = 0.3010 + 2 \cdot 0.1761 + \frac{2 \cdot 1}{2!} (-0.0511) = 0.6021$$

- Esta fórmula se generaliza para proporcionar valores aproximados de $y = f(x)$ para cualquier x :

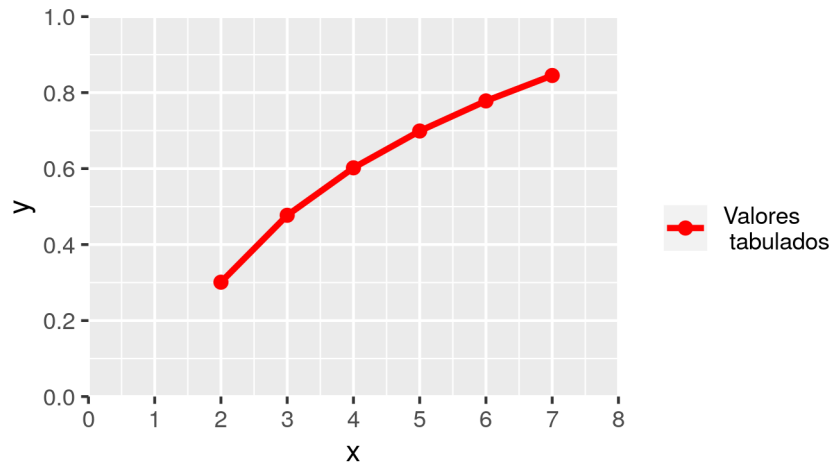
$$\begin{aligned} y = f(x) &\cong \sum_{j=0}^g \binom{k}{j} \Delta^j y_0 = \\ &y_0 + k\Delta y_0 + \frac{k(k-1)}{2!} \Delta^2 y_0 + \frac{k(k-1)(k-2)}{3!} \Delta^3 y_0 + \dots \end{aligned}$$

donde g es el orden de la aproximación, $\Delta^0 y_0 = y_0$ y $k = \frac{x-x_0}{h}$.

- Si $x_0 < x < x_n$, este proceso se llama **interpolación** y la fórmula anterior, **Fórmula de interpolación de Newton para incrementos constantes**.
- Si para hallar una aproximación sólo empleamos la diferencia de 1º orden, tenemos una **interpolación lineal**: $f(x) \approx y_0 + k\Delta y_0$. El polinomio interpolador es de grado 1: una recta que pasa por los puntos (x_0, y_0) y (x_1, y_1) .
- Si empleamos hasta la diferencia de 2º orden, tenemos una **interpolación cuadrática**: $f(x) \approx y_0 + k\Delta y_0 + \frac{k(k-1)}{2!} \Delta^2 y_0$. El polinomio interpolador es de grado 2: una parábola que pasa por los puntos (x_0, y_0) , (x_1, y_1) y (x_2, y_2) .
- Cuantas más diferencias empleemos, el polinomio interpolador es de mayor orden y puede brindar mejores aproximaciones.
- Si empleamos las n diferencias, el polinomio interpolador es el único polinomio de grado n que pasa exactamente por los $n+1$ puntos tabulados.

Retomando el Ejemplo 1: vamos a aproximar el valor de $f(2,3)$. Recordamos la tabla de diferencias y el gráfico:

x_k	y_k	Δy_k	$\Delta^2 y_k$	$\Delta^3 y_k$	$\Delta^4 y_k$	$\Delta^5 y_k$
2	0,3010	0,1761	-0,0511	0,0230	-0,0127	0,0081
3	0,4771	0,1250	-0,0281	0,0103	-0,0046	-
4	0,6021	0,0969	-0,0178	0,0057	-	-
5	0,6990	0,0791	-0,0121	-	-	-
6	0,7781	0,0670	-	-	-	-
7	0,8451	-	-	-	-	-



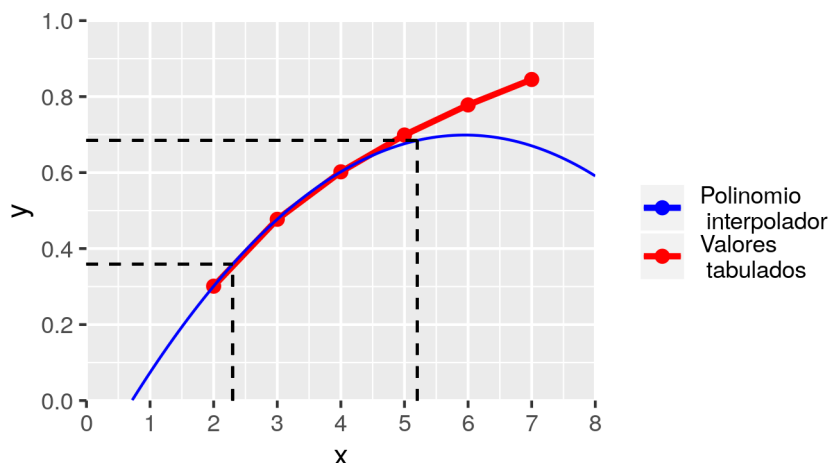
- ¿Hasta qué orden de diferencias deberíamos incluir? Es decir, ¿qué grado elegimos para el polinomio interpolador?
- Viendo el gráfico podemos pensar que una función cuadrática puede proveer un buen ajuste.
- Por eso, proponemos un polinomio interpolador de grado 2, usando hasta la diferencia de segundo orden:
 - $x = 2, 3$
 - $x_0 = 2$
 - $h = 1$
 - $k = \frac{x-x_0}{h} = \frac{2,3-2}{1} = 0,3$
 - $y_0 = 0,3010$; $\Delta y_0 = 0,1761$; $\Delta^2 y_0 = -0,0511$

$$\begin{aligned}
 y = f(2,3) &\approx y_0 + k\Delta y_0 + \frac{k(k-1)}{2!}\Delta^2 y_0 \\
 &= 0,3010 + k \cdot 0,1761 + \frac{k(k-1)}{2!} \cdot (-0,0511) \\
 &= 0,3010 + 0,3 \cdot 0,1761 + \frac{0,3(0,3-1)}{2}(-0,0511) \\
 &= 0,3592
 \end{aligned}$$

- ¿Podemos encontrar la expresión del polinomio interpolador que acabamos de usar?
- Sí, sólo debemos reemplazar k en la fórmula anterior por $k = \frac{x-x_0}{h} = \frac{x-2}{1} = x-2$:

$$\begin{aligned}
 y = f(x) &\approx 0,3010 + k \cdot 0,1761 + \frac{k(k-1)}{2!} \cdot (-0,0511) \\
 &= 0,3010 + (x-2) \cdot 0,1761 + \frac{(x-2)(x-2-1)}{2!} \cdot (-0,0511) \\
 &= -0,02555x^2 + 0,30385x - 0,2045
 \end{aligned}$$

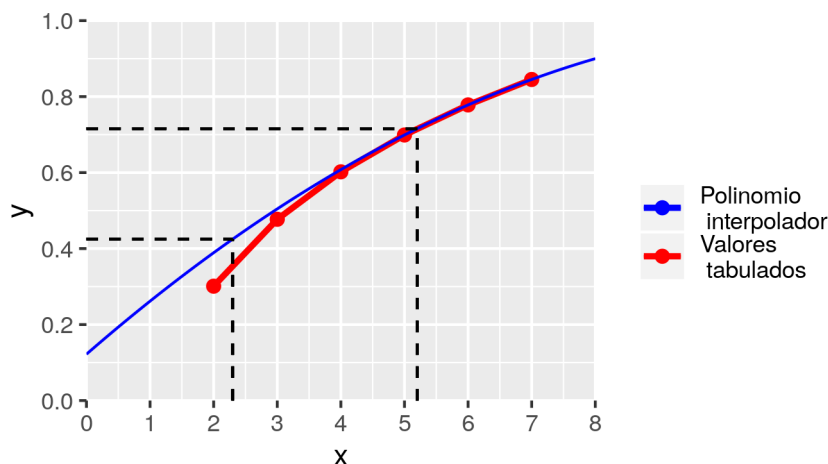
- Al representar gráficamente el polinomio interpolador, observamos que ajusta exactamente a los 3 primeros puntos tabulados y parece dar una aproximación razonable para $x = 2, 3$.



- Sin embargo, si queremos interpolar para $x = 5,2$, este polinomio no parece ser muy útil, porque 5,2 se aleja demasiado de los puntos ajustados.
- Por eso, siempre “corremos” el valor inicial x_0 hasta el inmediato inferior del que queremos aproximar.
- Entonces, para $x = 5,2$ tomamos:
 - $x_0 = 5$
 - $h = 1$
 - $k = \frac{x-x_0}{h} = \frac{5,2-5}{1} = 0,2$
 - $y_0 = 0,6990$; $\Delta y_0 = 0,0791$; $\Delta^2 y_0 = -0,0121$

$$\begin{aligned}
 y = f(5,2) &\approx y_0 + k\Delta y_0 + \frac{k(k-1)}{2!}\Delta^2 y_0 \\
 &= 0,6990 + 0,2 \cdot 0,0791 + \frac{0,2(0,2-1)}{2}(-0,0121) \\
 &= 0,7158
 \end{aligned}$$

- Así, con $x_0 = 5$, la expresión del polinomio interpolador queda $f(x) \approx -0,00605x^2 + 0,14565x - 0,122$ y ajusta exactamente a los tres últimos puntos tabulados:



- Retomemos la interpolación cuadrática para $x = 2,3$ que nos dio $f(x) \approx 0,3592$.
- ¿Cómo cambiará la aproximación si usamos polinomios de mayor orden, incluyendo diferencias superiores?
- Incrementar un grado en el orden del polinomio interpolador es muy sencillo, porque solamente tenemos que sumarle un término al polinomio de grado inferior.
- En el ejemplo con $x = 2,3$; $x_0 = 2$; $h = 1$; $k = 0,3$:

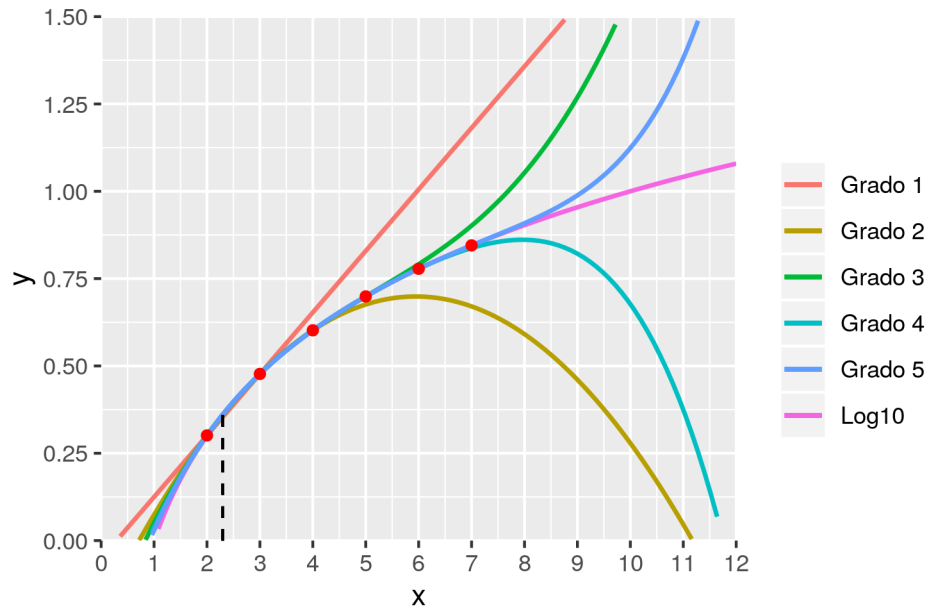
Grado	$f(2,3) \approx$
1	$y_0 + k\Delta y_0 = 0,3538$

Grado	$f(2.3) \approx$
2	$0.3538 + \frac{k(k-1)}{2!} \Delta^2 y_0 = 0.3592$
3	$0.3592 + \frac{k(k-1)(k-2)}{3!} \Delta^3 y_0 = 0.3606$
4	$0.3606 + \frac{k(k-1)(k-2)(k-3)}{4!} \Delta^4 y_0 = 0.3611$
5	$0.3611 + \frac{k(k-1)(k-2)(k-3)(k-4)}{5!} \Delta^5 y_0 = 0.3613$

- La función que generó la tabla de este ejemplo es el logaritmo en base 10, es decir, que el valor exacto es $f(2, 3) = \log(2, 3) = 0,3617$.
- Con esta información, podemos calcular el Error Relativo que resulta de hacer las aproximaciones anteriores:

Grado	$f(2.3) \approx$	ER
1	0.3538	0.0218
2	0.3592	0.0070
3	0.3606	0.0032
4	0.3611	0.0018
5	0.3613	0.0012

Polinomios interpoladores con $x_0 = 2$:



- Realizar una interpolación cúbica para $x = 6.4$.
- Si tomamos $x_0 = 6$, nos faltan las diferencias de 2° y 3° orden y no podemos obtener el polinomio deseado.
- Modificación:** tomar como x_0 al inmediato superior y emplear las diferencias que se encuentran en la diagonal ascendente que comienza en dicho punto.
- Se debe redefinir k y alternar signos positivos y negativos en la fórmula:

x_k	y_k	Δy_k	$\Delta^2 y_k$	$\Delta^3 y_k$	$\Delta^4 y_k$	$\Delta^5 y_k$
2	0,3010	0,1761	-0,0511	0,0230	-0,0127	0,0081
3	0,4771	0,1250	-0,0281	0,0103	-0,0046	-
4	0,6021	0,0969	-0,0178	0,0057	-	-
5	0,6990	0,0791	-0,0121	-	-	-
6	0,7781	0,0670	-	-	-	-
7	0,8451	-	-	-	-	-

$$x_0 = 7$$

$$k = \frac{x_0 - x}{h} = \frac{7 - 6.4}{1} = 0.6$$

$$y_0 = 0,8451; \Delta y_0 = 0,0670;$$

$$\Delta^2 y_0 = -0.0121; \Delta^3 y_0 = 0.0057$$

$$\begin{aligned} y = f(6.4) &\approx y_0 - k\Delta y_0 + \frac{k(k-1)}{2!}\Delta^2 y_0 - \frac{k(k-1)(k-2)}{3!}\Delta^3 y_0 \\ &= 0.8451 - 0,6 \cdot 0.0670 + \frac{0.6(0.6-1)}{2}(-0.0121) - \frac{0.6(0.6-1)(0.6-2)}{6}0.0057 \\ &= 0.8063 \end{aligned}$$

Ejemplo 2. Dada la siguiente función tabulada, interpolar para $x = 3.2$.

x	$y = f(x)$
0	2
2	8
4	62
6	212
8	506
10	992

Ejemplo 2. La tabla de diferencias es:

x_k	$y_k = f(x_k)$	Δy_k	$\Delta^2 y_k$	$\Delta^3 y_k$	$\Delta^4 y_k$
0	2	6	48	48	0
2	8	54	96	48	0
4	62	150	144	48	-
6	212	294	192	-	-
8	506	486	-	-	-
10	992	-	-	-	-

- Las diferencias de orden 3 son constantes. Esto significa que la función $f(x)$ es exactamente un polinomio de grado 3.
- Es decir, si usamos hasta la diferencia de orden 3, con la fórmula de Newton podemos hallar exactamente el valor de $f(x)$ para cualquier x (y no importa cuál valor tomamos como x_0).

Ejemplo 2. Continuando con el objetivo de hallar $f(3.2)$:

- $x = 3.2$
- $x_0 = 2$
- $h = 2$
- $k = \frac{x - x_0}{h} = \frac{3.2 - 2}{2} = 0.6$
- $y_0 = 8; \Delta y_0 = 54; \Delta^2 y_0 = 96; \Delta^3 y_0 = 48$

$$\begin{aligned} y = f(3.2) &= y_0 + k\Delta y_0 + \frac{k(k-1)}{2!}\Delta^2 y_0 + \frac{k(k-1)(k-2)}{3!}\Delta^3 y_0 \\ &= 8 + 0.6 \cdot 54 + \frac{0.6(0.6-1)}{2}96 + \frac{0.6(0.6-1)(0.6-2)}{6}48 \\ &= 31.568 \end{aligned}$$

Ejemplo 2. Ahora interpolar para $x = 9$. Recordemos la tabla:

x_k	$y_k = f(x_k)$	Δy_k	$\Delta^2 y_k$	$\Delta^3 y_k$	$\Delta^4 y_k$
0	2	6	48	48	0
2	8	54	96	48	0

x_k	$y_k = f(x_k)$	Δy_k	$\Delta^2 y_k$	$\Delta^3 y_k$	$\Delta^4 y_k$
4	62	150	144	48	-
6	212	294	192	-	-
8	506	486	-	-	-
10	992	-	-	-	-

- Si tomamos como x_0 al inmediato inferior, es decir, $x_0 = 8$, nos faltan dos diferencias para completar el cálculo.
- Pero como tenemos diferencias constantes y la verdadera función es un polinomio, cualquier punto de partida nos dará el valor exacto de $f(9)$.
- Así que podemos tomar, por ejemplo, el primer renglón de la tabla: $x_0 = 0$.

Ejemplo 2. Continuando con el objetivo de hallar $f(9)$:

- $x = 9$
- $x_0 = 0$
- $h = 2$
- $k = \frac{x-x_0}{h} = \frac{9-0}{2} = 4.5$
- $y_0 = 2$; $\Delta y_0 = 6$; $\Delta^2 y_0 = 48$; $\Delta^3 y_0 = 48$

$$\begin{aligned}
 y = f(9) &= y_0 + k\Delta y_0 + \frac{k(k-1)}{2!}\Delta^2 y_0 + \frac{k(k-1)(k-2)}{3!}\Delta^3 y_0 \\
 &= 2 + 4.5 \cdot 6 + \frac{4.5(4.5-1)}{2}48 + \frac{4.5(4.5-1)(4.5-2)}{6}48 \\
 &= 722
 \end{aligned}$$

- En los ejemplos anteriores obtuvimos el valor de y para x comprendida entre dos valores x_k de la tabla (*interpolación*).
- Ahora vamos a calcular el valor de y para una x fuera del rango de los valores de x_k en la tabla.
- Esto se llama **extrapolación**.

Ejemplo 3. Extrapolar para $x = -1$.

- $x = -1$
- $x_0 = 0$
- $h = 2$
- $k = \frac{x-x_0}{h} = \frac{-1-0}{2} = -0.5$
- $y_0 = 2$; $\Delta y_0 = 6$; $\Delta^2 y_0 = 48$; $\Delta^3 y_0 = 48$

$$\begin{aligned}
 y = f(-1) &= y_0 + k\Delta y_0 + \frac{k(k-1)}{2!}\Delta^2 y_0 + \frac{k(k-1)(k-2)}{3!}\Delta^3 y_0 \\
 &= 2 - 0.5 \cdot 6 + \frac{-0.5(-0.5-1)}{2}48 + \frac{-0.5(-0.5-1)(-0.5-2)}{6}48 \\
 &= 2
 \end{aligned}$$

Ejemplo 4. Extrapolar para $x = 11$.

- $x = 11$
- $x_0 = 0$
- $h = 2$
- $k = \frac{x-x_0}{h} = \frac{11-0}{2} = 5.5$
- $y_0 = 2$; $\Delta y_0 = 6$; $\Delta^2 y_0 = 48$; $\Delta^3 y_0 = 48$

$$y = f(11) = \dots = 1322$$

4.4. Interpolación de Lagrange

- Como ya mencionamos, en la interpolación lineal se utiliza un segmento rectilíneo que pasa por dos puntos que se conocen.

- De nuestros conocimientos de Geometría sabemos que la ecuación de la recta que pasa por dos puntos (x_0, y_0) y (x_1, y_1) es:

$$y = f(x) = y_0 + \frac{y_1 - y_0}{x_1 - x_0}(x - x_0)$$

- Es fácil ver que la recta anterior pasa por los puntos dados, ya que $f(x_0) = y_0$ y $f(x_1) = y_1$.
- Lagrange propuso reescribir la recta anterior como:

$$f(x) = \frac{x - x_1}{x_0 - x_1}y_0 + \frac{x - x_0}{x_1 - x_0}y_1$$

- Esta es expresión que se puede extender para obtener polinomios de grados superiores que pasen por más puntos.
- Por ejemplo, para hallar el polinomio que pasa por los puntos (x_0, y_0) , (x_1, y_1) y (x_2, y_2) :

$$f(x) = \frac{(x - x_1)(x - x_2)}{(x_0 - x_1)(x_0 - x_2)}y_0 + \frac{(x - x_0)(x - x_2)}{(x_1 - x_0)(x_1 - x_2)}y_1 + \frac{(x - x_0)(x - x_1)}{(x_2 - x_0)(x_2 - x_1)}y_2$$

- Se puede ver fácilmente que este polinomio pasa exactamente por los tres puntos dados.
- De la misma forma, el polinomio de grado 3 que pasa por cuatro puntos (x_0, y_0) , (x_1, y_1) , (x_2, y_2) y (x_3, y_3) es:

$$f(x) = \frac{(x - x_1)(x - x_2)(x - x_3)}{(x_0 - x_1)(x_0 - x_2)(x_0 - x_3)}y_0 + \frac{(x - x_0)(x - x_2)(x - x_3)}{(x_1 - x_0)(x_1 - x_2)(x_1 - x_3)}y_1 + \frac{(x - x_0)(x - x_1)(x - x_3)}{(x_2 - x_0)(x_2 - x_1)(x_2 - x_3)}y_2 + \frac{(x - x_0)(x - x_1)(x - x_2)}{(x_3 - x_0)(x_3 - x_1)(x_3 - x_2)}y_3$$

- Se puede ver fácilmente que este polinomio pasa exactamente por los cuatro puntos dados.
- Generalizando, la fórmula de interpolación de Lagrange para construir un polinomio de grado n que pase por $n + 1$ puntos $(x_0, y_0), (x_1, y_1), \dots, (x_n, y_n)$ es:

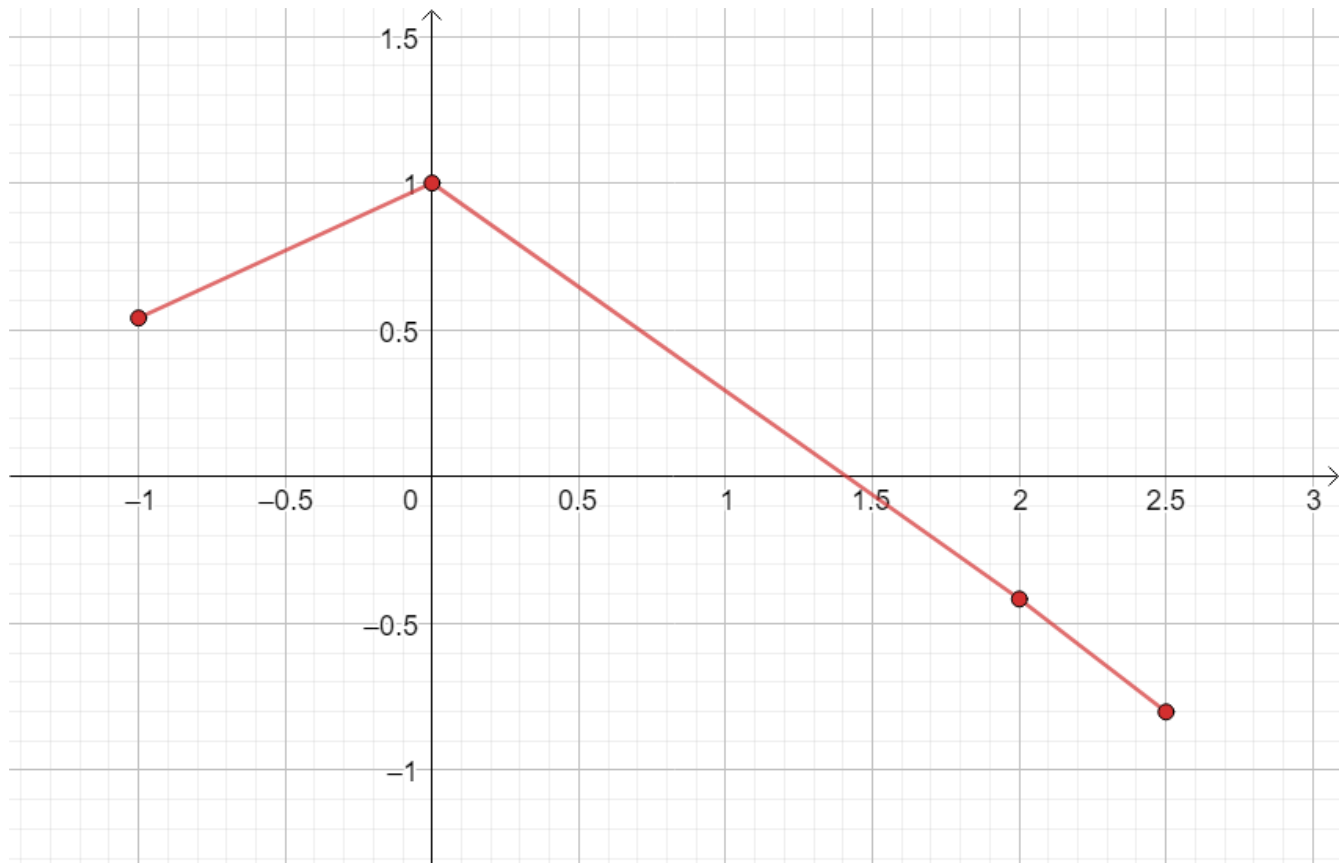
$$y = f(x) = \sum_{i=0}^n \frac{\prod_{\substack{j=0 \\ j \neq i}}^n (x - x_j)}{\prod_{\substack{j=0 \\ j \neq i}}^n (x_i - x_j)} y_i$$

Comparación con el método de Newton

- **Ventaja:** no requiere que los valores tabulados de x estén equiespaciados. Si lo están, ambos métodos coinciden.
- **Desventaja:**
 - En ocasiones es útil considerar varios polinomios interpoladores de distinto grado para luego elegir el más adecuado.
 - Con el método de Newton esto es muy sencillo, ya que como vimos se pueden construir recursivamente (por ejemplo, el de grado 3 es igual al de grado 2 más un término extra).
 - En cambio, con el método de Lagrange no hay relación entre la construcción de $P_{n-1}(x)$ y la de $P_n(x)$; cada polinomio debe construirse individualmente realizando todos los cálculos otra vez.
 - Esto implica que este método sea menos eficiente al tener que recalcular todo el polinomio si varía el número de puntos.

Ejemplo 5. Emplear el método de Lagrange para interpolar el valor de la función f en $x = 2.25$, con el polinomio interpolador de grado 3 que pasa por los cuatro puntos tabulados:

x_k	y_k
-1	0,5403
0	1,0000
2	-0,4162
2.5	-0,8011

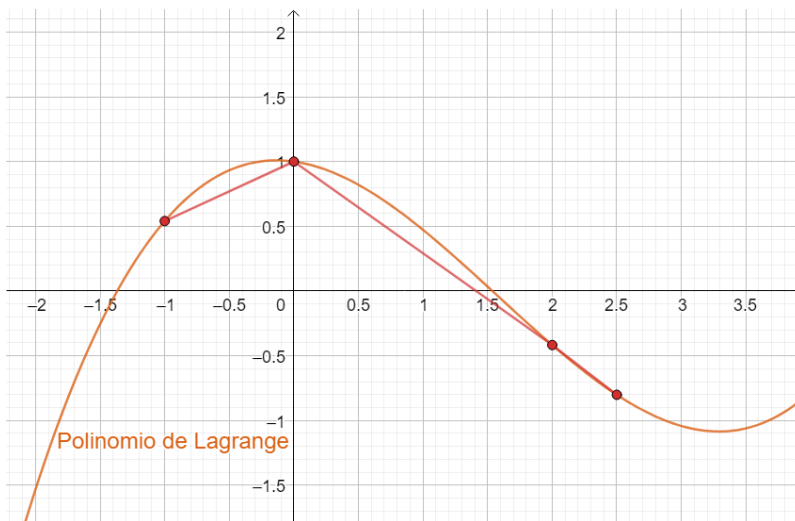


Ejemplo 5

$$\begin{aligned}
 f(x) &= \sum_{i=0}^3 \frac{\prod_{\substack{j=0 \\ j \neq i}}^3 (x - x_j)}{\prod_{\substack{j=0 \\ j \neq i}}^3 (x_i - x_j)} y_i \\
 &= \frac{(x - x_1)(x - x_2)(x - x_3)}{(x_0 - x_1)(x_0 - x_2)(x_0 - x_3)} y_0 + \frac{(x - x_0)(x - x_2)(x - x_3)}{(x_1 - x_0)(x_1 - x_2)(x_1 - x_3)} y_1 \\
 &\quad + \frac{(x - x_0)(x - x_1)(x - x_3)}{(x_2 - x_0)(x_2 - x_1)(x_2 - x_3)} y_2 + \frac{(x - x_0)(x - x_1)(x - x_2)}{(x_3 - x_0)(x_3 - x_1)(x_3 - x_2)} y_3 \\
 \Rightarrow y = f(2.25) &\approx \frac{(2.25 - 0)(2.25 - 2)(2.25 - 2.5)}{(-1 - 0)(-1 - 2)(-1 - 2.5)} 0.5403 + \frac{(2.25 + 1)(2.25 - 2)(2.25 - 2.5)}{(0 + 1)(0 - 2)(0 - 2.5)} 1.0000 \\
 &\quad + \frac{(2.25 + 1)(2.25 - 0)(2.25 - 2.5)}{(2 + 1)(2 - 0)(2 - 2.5)} (-0.4162) + \frac{(2.25 + 1)(2.25 - 0)(2.25 - 2)}{(2.5 + 1)(2.5 - 0)(2.5 - 2)} (-0.8011) \\
 &= -0.6218 \\
 \therefore f(2.25) &\approx -0.6218
 \end{aligned}$$

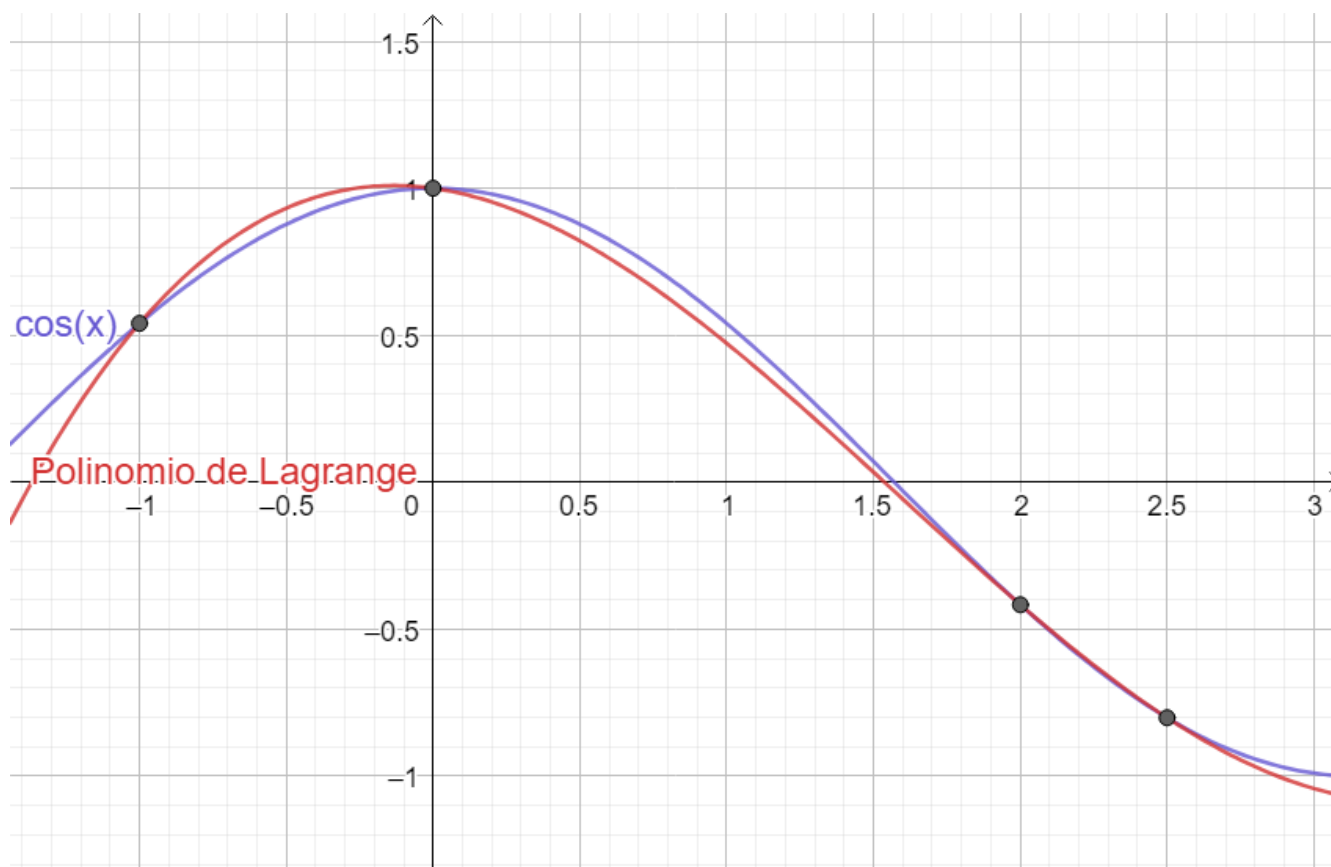
Ejemplo 5. Si en la fórmula anterior en lugar de reemplazar x por un valor particular (en este caso, 2.5) operamos y reacomodamos los términos, podemos hallar la expresión del polinomio interpolador:

$$f(x) = 0.1042x^3 - 0.4934x^2 - 0.1379x + 1$$

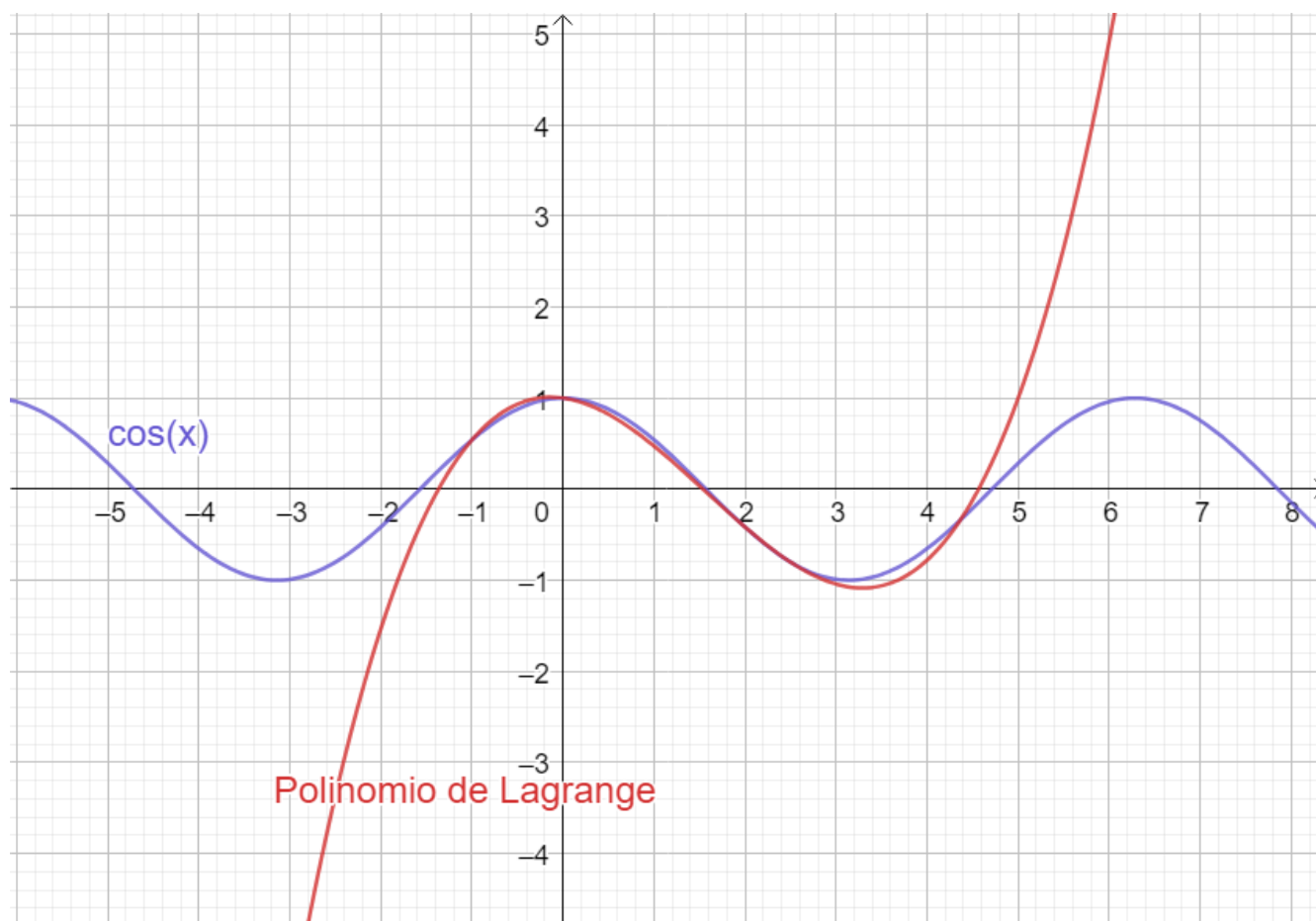


Ejemplo 5. La función que generó los valores tabulados es $\cos(x)$:

En el rango tabulado:



Un poco más lejos:



- Siendo $\cos(2.25) = -0.6282$, el error relativo de la aproximación fue $|-0.6282 - 0.6218| / |-0.6285| = 1.02\%$

Interpolación inversa

- El problema de interpolación consiste en determinar el valor de la función desconocida $f(x)$ a partir de un valor conocido de x .
- En la *interpolación inversa* se resuelve el problema contrario: determinar el valor de x conociendo el de $f(x)$.
- Podemos implementar interpolación inversa con el método de Lagrange intercambiando el rol de las columnas tabuladas x e y , **pero sólo para rangos tabulados donde la función f es monótona**.

Ejemplo 6. Con los datos del Ejemplo 1, ¿cuál es el valor de x tal que $f(x) = -0.75$?

x_k	y_k
-1	0,5403
0	1,0000
2	-0,4162
2.5	-0,8011

↓

x_k	y_k
1,0000	0
-0,4162	2
-0,8011	2.5

$$f(x) = \sum_{i=0}^2 \frac{\prod_{\substack{j=0 \\ j \neq i}}^2 (x - x_j)}{\prod_{\substack{j=0 \\ j \neq i}}^2 (x_i - x_j)} y_i$$

$$= \frac{(x - x_1)(x - x_2)}{(x_0 - x_1)(x_0 - x_2)} y_0 + \frac{(x - x_0)(x - x_2)}{(x_1 - x_0)(x_1 - x_2)} y_1 + \frac{(x - x_0)(x - x_1)}{(x_2 - x_0)(x_2 - x_1)} y_2$$

$$\Rightarrow f(-0.75) \approx \frac{(-0.75 + 0.4162)(-0.75 + 0.8011)}{(1 + 0.4162)(1 + 0.8011)} 0$$

$$+ \frac{(-0.75 - 1)(-0.75 + 0.8011)}{(-0.4162 - 1)(-0.4162 + 0.8011)} 2$$

$$+ \frac{(-0.75 - 1)(-0.75 + 0.4162)}{(-0.8011 - 1)(-0.8011 + 0.4162)} 2.5$$

$$= 2.4347$$

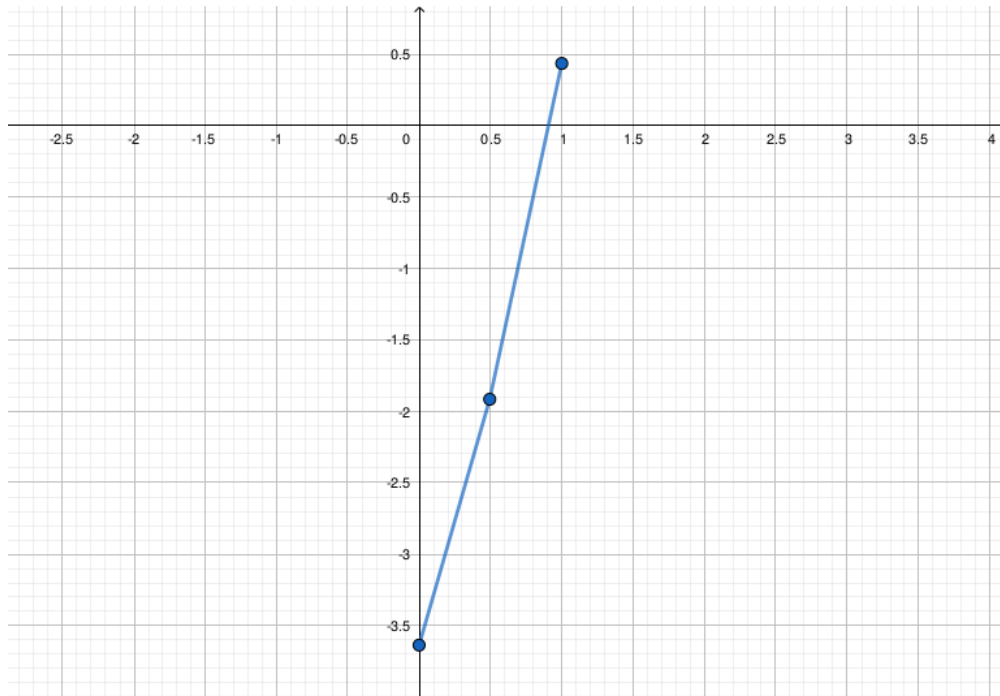
\therefore el valor de x tal que $f(x) = -0.75$ es $x \approx 2.4347$.

Interpolación inversa. Podemos usar la interpolación inversa para resolver ecuaciones.

Ejemplo 7: hallar una raíz positiva para la ecuación $f(x) = x - 4 \sin(x + 2) = 0$.

- Tabulamos algunos valores y graficamos:

x_k	y_k
0	-3.6372
0.5	-1.8939
1	0.4355



Ejemplo 7

- Podemos usar los dos últimos puntos donde vemos que la función cortó a las abscisas y hacer una interpolación lineal, los tres puntos y tener una interpolación cuadrática, o incluso agregar más puntos.

x_k	y_k
0	-3.6372
0.5	-1.8939
1	0.4355

↓

x_k	y_k
-3.6372	0
-1.8939	0.5
0.4355	1

$$y = f(x) = \frac{x - x_1}{x_0 - x_1} y_0 + \frac{x - x_0}{x_1 - x_0} y_1$$

$$\Rightarrow f(0) \approx \frac{0 - 0.4355}{-1.8939 - 0.4355} 0.5 + \frac{0 + 1.8939}{0.4355 + 1.8939} 1$$

$$= 0.9065$$

∴ La solución positiva de la ecuación es $x \approx 0.9065$ ya que $f(0.9065) \approx 0$.

4.4.1. Observaciones finales

- Un polinomio de grado n ajustado a $n + 1$ puntos es único.
- El polinomio de interpolación se puede expresar en varias formas distintas, pero todas son equivalentes por el punto anterior.
- Si una función se aproxima mediante un polinomio de interpolación, no hay garantía de que dicho polinomio converja a la función exacta al aumentar el número de datos. En general, la interpolación mediante un polinomio de orden grande debe evitarse o utilizarse con precauciones extremas.
- Aunque no existe un criterio para determinar el orden óptimo del polinomio de interpolación, generalmente se recomienda utilizar uno con orden relativamente bajo en un pequeño rango de x .