

Teoría

Teoría:

▼ Teoría Unidad 2-3:

número de condición: $\text{cond}(A)$

norma de la matriz: $\text{norm}(A)$

Métodos Directos:

- Gradiente Conjugado:

Si la matriz A tiene rango r , converge a lo sumo en r iteraciones.

Es eficiente si el número de condición es bajo (~ 1). Sino converge lento.

Métodos Iterativos:

→ convergen si $\|T\| < 1$ (norma de la matriz de iteración).

→ converge si $\rho(T) < 1$ (radio espectral de la matriz de iteración).

Converge más lento a medida que tiende a 1. Si tiende a 0, converge rápido.

Jacobi:

Si la matriz es diagonalmente dominante, Jacobi converge para cualquier x_0 .

Gauss-Seidel:

Si la matriz es diagonalmente dominante, GS converge.

Si la matriz es simétrica definida positiva, GS converge.

SOR:

Si $w = 1$, SOR → Gauss-Seidel

▼ Teoría Unidad 4:

Bisección:

- La función $f(x)$ debe ser **continua** en el intervalo $[a,b]$.
- Debe cumplirse que $f(a) \times f(b) < 0$ (cambio de signo \Rightarrow hay al menos una raíz por el Teorema de Bolzano).
- Siempre converge, pero lentamente (**convergencia lineal**).

Punto Fijo:

- Debes reescribir $f(x)=0$ como $x=g(x)$.
- $g(x)$ debe ser **continua** en un intervalo cerrado que contenga la solución.
- $|g'(x)| < 1$ en ese intervalo para garantizar la **convergencia** (criterio de contracción de Banach).
- Si $|g'(x)| \geq 1$ puede divergir.

Newton-Raphson:

- $f(x)$ debe ser **derivable** (y $f'(x)$ no debe ser cero cerca de la raíz).
- La raíz debe ser **simple** (no múltiple) para garantizar **convergencia cuadrática**.
- Se requiere una buena **aproximación inicial** cerca de la raíz.
- Si $f''(x)$ es pequeña o razonablemente controlada cerca de la raíz, mejora la estabilidad.

Secante:

- Similar a Newton-Raphson, pero no requiere derivada explícita.
- $f(x)$ debe ser **continua** y "suavemente comportada" (sin saltos o discontinuidades fuertes).
- Necesita dos aproximaciones iniciales buenas cerca de la raíz.
- Convergencia **superlineal** (~ 1.618), pero más lenta que Newton-Raphson.

▼ Teoría Unidad 5:

La interpolación polinomial es el proceso de encontrar un polinomio de grado n que pase exactamente por $n+1$ puntos dados $(x_0, y_0), (x_1, y_1), \dots, (x_n, y_n)$ tal que:

$$P(x_i) = y_i$$

→ **Polinomios de Lagrange:** Los coeficientes en este caso son los $y_k = f(x_k)$, datos del problema, y la construcción de los polinomios $\phi_k(x)$ es muy sencilla:

$$P(x) = \sum_{i=0}^n y_i \cdot L_i(x)$$

Tal que L es:

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

No requiere resolver sistemas, pero es poco eficiente si se agregan más puntos.

→ **Dif. divididas de Newton:**

$$P(x) = f[x_0] + f[x_0, x_1](x - x_0) + f[x_0, x_1, x_2](x - x_0)(x - x_1) + \dots$$

donde las diferencias divididas se calculan sucesivamente de forma tal:

$$f[x_i, x_{i+1}] = \frac{f(x_{i+1}) - f(x_i)}{x_{i+1} - x_i}$$

$$f[x_i, x_{i+1}, x_{i+2}] = \frac{f[x_{i+1}, x_{i+2}] - f[x_i, x_{i+1}]}{x_{i+2} - x_i}$$

→ **Polinomios Osculantes:** Un polinomio osculante coincide con la función en los $n + 1$ puntos, y sus derivadas (hasta un orden $\leq m_i$) coinciden con las derivadas respectivas de la función.

Un polinomio osculante es aquel que iguala no solo los valores de la función, sino también sus derivadas:

$$P(x_i) = f(x_i), \quad P'(x_i) = f'(x_i), \quad P''(x_i) = f''(x_i)$$

→ **Polinomios de Hermite:** Son polinomios osculantes con $m_i = 1, \forall i$. Coinciden el polinomio y la función en sus valores y en sus primeras derivadas, en todos los puntos x_i ($i = 0, 1, \dots, n$). Dado un conjunto de datos que incluye valores y derivadas:

$$f(x_0), f'(x_0), f(x_1), f'(x_1), \dots$$

Se construye un polinomio $H(x)$ tal que:

$$H(x_i) = f(x_i), \quad H'(x_i) = f'(x_i)$$

→ **Forma en dif. divididas de Newton para Pol. de Hermite:**

Para cada punto x_i con valor y derivada conocida, se duplica la entrada en la tabla. Por ejemplo:

$$z_0 = x_0, \quad z_1 = x_0, \quad z_2 = x_1, \quad z_3 = x_1, \dots$$

$$f[z_0, z_1] = f'(x_0)$$

$$f[z_1, z_2] = \frac{f(x_1) - f(x_0)}{x_1 - x_0}, \quad \text{etc.}$$

El polinomio de Hermite es:

$$H(x) = f[z_0] + f[z_0, z_1](x - z_0) + f[z_0, z_1, z_2](x - z_0)(x - z_1) + \dots$$

→ **Splines:** Si hay grandes cambios de curvatura en partes de la función puede ser que los polinomios globales se desvíen mucho de la curva a representar → descomponer la curva en subintervalos y usar polinomios diferentes para cada subintervalo: aproximación segmentaria

1. En lugar de un polinomio global, dividimos el intervalo $[x_0, x_n]$ en subintervalos: $[x_0, x_1], [x_1, x_2], \dots, [x_{n-1}, x_n]$.
2. En cada subintervalo, se usa **un polinomio diferente**, pero con condiciones para que la unión de todos esos polinomios sea **suave y continua**.
 - a. **Spline de grado 0** (por tramos constantes) → Resultado: una **función escalonada**.

$$S(x) = \begin{cases} C_0 & x \in [x_0, x_1) \\ C_1 & x \in [x_1, x_2) \\ \vdots & \\ C_{n-1} & x \in [x_{n-1}, x_n) \end{cases}$$

- b. **Spline de grado 1** (por tramos lineales) → Rectas conectadas con **continuidad**, pero **no tienen suavidad** en las pendientes.

$$S(x) = \begin{cases} a_0x + b_0 & x \in [x_0, x_1) \\ a_1x + b_1 & x \in [x_1, x_2) \\ \vdots & \vdots \\ a_{n-1}x + b_{n-1} & x \in [x_{n-1}, x_n) \end{cases}$$

- c. **Spline cúbico** → Continúa en los puntos de unión (llamados "nudos") y las derivadas primera y segunda también son continuas, lo que da una **curva suave**.

$$S_i(x) = a_i + b_i(x - x_i) + c_i(x - x_i)^2 + d_i(x - x_i)^3$$

Tipos de fronteras de splines cúbicos:

Spline cúbico natural → $S''(x_0)=0$ y $S''(x_n)=0$

La curva es recta en los extremos (curvatura nula).

Spline cúbico sujeto → $S'(x_0)=f'(x_0)$ y $S'(x_n)=f'(x_n)$

Necesita conocer las **derivadas en los extremos**.

Construir spline cúbico:

Paso 1: Condiciones a cumplir

Para $n + 1$ puntos $(x_0, f_0), \dots, (x_n, f_n)$, se tienen n tramos S_0, S_1, \dots, S_{n-1} . Cada uno tiene 4 coeficientes (a_i, b_i, c_i, d_i) , así que hay $4n$ incógnitas.

Hay que generar $4n$ ecuaciones. Estas vienen de:

1. Interpolación exacta: $S_i(x_i) = f(x_i) \rightarrow n$ ecuaciones
2. Continuidad entre tramos: $S_i(x_{i+1}) = S_{i+1}(x_{i+1}) \rightarrow n - 1$ ecuaciones
3. Derivadas continuas:
 - Primera derivada continua $\rightarrow n - 1$
 - Segunda derivada continua $\rightarrow n - 1$
4. Valor en el último punto: $S_{n-1}(x_n) = f(x_n) \rightarrow 1$ ecuación

Esto suma:

$$n + (n - 1) + (n - 1) + (n - 1) + 1 = 4n - 2$$

👉 Faltan 2 ecuaciones → se agregan con condiciones de frontera.

Paso 2: Expresiones útiles

Define:

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

Se forma un sistema para hallar los c_i :

$$h_{i-1}c_{i-1} + 2(h_{i-1} + h_i)c_i + h_ic_{i+1} = 3 \left(\frac{a_{i+1} - a_i}{h_i} - \frac{a_i - a_{i-1}}{h_{i-1}} \right)$$

- Esto forma una **matriz tridiagonal** de tamaño $n - 1 \times n - 1$

Luego se calculan los otros coeficientes:

$$d_i = \frac{c_{i+1} - c_i}{3h_i}, \quad b_i = \frac{a_{i+1} - a_i}{h_i} - \frac{2c_i + c_{i+1}}{3}h_i$$

Paso 3: Resolver sistema lineal

$$A \cdot x = b$$

- A : matriz tridiagonal
- x : los c_i
- b : vector con las diferencias de los $a_i = f(x_i)$

Una vez obtenidos los c_i , puedes calcular todos los b_i, d_i , y ya tienes tu spline.

Ajuste de curvas por mínimos cuadrados: Se puede proponer funciones no polinómicas para ajustar los datos. Ej: $y = c \cdot e^{ax}$ → utilizo logaritmo.

▼ Teoría Unidad 6:

▼ Derivación Numérica:

La derivada de una función $f(x)$ se define como:

$$f'(x) = \lim_{h \rightarrow 0} \frac{f(x+h) - f(x)}{h}$$

Como el límite no puede calcularse directamente en un computador, se utilizan **aproximaciones numéricas**. Estas se basan en el desarrollo en series de Taylor de la función alrededor del punto x .

▼ Fórmula de 2 puntos (diferencias finitas hacia adelante).

Partiendo del desarrollo de Taylor y truncando después del primer término:

$$f(x+h) = f(x) + hf'(x) + \frac{h^2}{2}f''(x) + \frac{h^3}{6}f'''(x) + \dots$$

Truncando después del primer término:

$$\tilde{f}'(x) = \frac{f(x+h) - f(x)}{h}$$

Donde el error de truncamiento, disminuye linealmente con h y es:

$$E = \frac{h}{2}f''(\xi), \quad \xi \in (x, x+h)$$

Se tomaron los puntos en (x) y $(x+h)$ por este motivo se denominatambién fórmula en diferencias finitas hacia adelante o fórmula en diferencias finitas progresivas. En cualquiera de estos casos el error de de primer orden, ya que depende de h .

*Si uso $(x-h)$ es diferencia finitas hacia atrás.

▼ Fórmula de 3 puntos.

Escribiendo el desarrollo en serie de Taylor:

$$\begin{aligned} f(x+h) &= f(x) + hf'(x) + \frac{h^2}{2}f''(x) + \frac{h^3}{3!}f^{(3)}(\xi_1) \\ f(x-h) &= f(x) - hf'(x) + \frac{h^2}{2}f''(x) - \frac{h^3}{3!}f^{(3)}(\xi_2) \end{aligned}$$

Restando ambas expresiones:

$$f'(x) = \frac{f(x+h) - f(x-h)}{2h} - \frac{h^2}{6} \left(f^{(3)}(\xi_1) + f^{(3)}(\xi_2) \right)$$

El estimador de la derivada queda:

$$\tilde{f}'(x) = \frac{f(x+h) - f(x-h)}{2h}$$

$$E = \frac{h^2}{6}f'''(\xi), \quad \xi \in (x-h, x+h)$$

El error es de segundo orden (depende de h^2).

▼ Fórmula de 5 puntos.

$$\tilde{f}'(x) = \frac{-f(x+2h) + 8f(x+h) - 8f(x-h) + f(x-2h)}{12h}$$

▼ Extrapolación de Richardson.

La **extrapolación de Richardson** es una técnica **muy inteligente** que se usa para **mejorar la precisión** de una estimación numérica y reducir el **error de truncamiento** al calcular derivadas numéricamente.

1. Calculamos la derivada con paso h :

$$\phi(h) = \text{aproximación con paso } h$$

2. Calculamos la derivada con paso $h/2$:

$$\phi(h/2) = \text{más precisa (porque usa un paso más chico)}$$

3. Combinamos ambas así:

$$\psi(h) = \frac{4}{3}\phi(h/2) - \frac{1}{3}\phi(h)$$

👉 ¡Y listo! Esta nueva fórmula tiene un **error mucho menor** (de orden h^4 en vez de h^2).

La idea es **usar dos aproximaciones distintas**: una con paso h y otra con paso más pequeño, $h/2$, para **cancelar el error principal**.

Se puede repetir el proceso. Si usás $\psi(h)$ y $\psi(h/2)$, los combinás de nuevo y eliminás el siguiente error. Así obtenés una derivada con error de orden $6h^6$, luego $8h^8$ y así...

▼ Derivadas a partir de polinomios interpolantes.

La idea es **interpol**ar la función $f(x)$ con un **polinomio de Lagrange**, y luego derivar ese polinomio para obtener una aproximación a la derivada de f .

$$f(x) = \sum_{i=0}^n f(x_i) L_{n,i}(x) + \frac{1}{(n+1)!} f^{(n+1)}(\xi_x) \prod_{j=0}^n (x - x_j)$$

$$L_{n,i}(x) = \prod_{\substack{j=0 \\ j \neq i}}^n \frac{x - x_j}{x_i - x_j}$$

$$f'(x) = \sum_{i=0}^n f(x_i) L'_{n,i}(x) + \frac{1}{(n+1)!} \frac{d}{dx} \left(f^{(n+1)}(\xi_x) \prod_{j=0}^n (x - x_j) \right)$$

$$x_k f'(x_k) = \sum_{i=0}^n f(x_i) L'_{n,i}(x_k)$$

$$\frac{d}{dx} \left[\prod_{j=0}^n (x - x_j) \right] = \sum_{i=0}^n \prod_{\substack{j=0 \\ j \neq i}}^n (x - x_j)$$

$$\frac{d}{dx} \left[\prod_{j=0}^n (x_k - x_j) \right] = \prod_{\substack{j=0 \\ j \neq k}}^n (x_k - x_j)$$

Al evaluar numericamente la derivada se introducen 2 tipos de errores:

- Error de truncamiento → Es debido al método numérico. Por ejemplo el error introducido al retener algunos términos de la Serie de Taylor, y descartar el resto.
- Error de redondeo → Debido a la aritmética finita de la computadora (aquí entrar los errores por poda -hacia abajo- o por redondeo -hacia arriba-)

▼ Integración Numérica:

El problema de hallar la integral definida de una función $f(x)$ no siempre puede lograrse analíticamente. Puede ser que:

- No se conozca la primitiva de la función $f(x)$
- No se conozca la función $f(x)$, sino que la misma esté descripta como una tabla de valores (x_i, y_i) $i = 0, n$

Una manera de abordar el problema es aproximar la función $f(x)$ con otra conocida (por ej. polinomio interpolador) y luego evaluar la integral de ese polinomio.

▼ Fórmulas de Newton-Cotes.

Definimos Newton-Cotes como la integral sobre una función, donde los puntos están igualmente espaciados. Fórmula:

$$I = \sum_{i=0}^n a_i f(x_i)$$

- Si se usan $(n+1)$ puntos → $a = x_0$ | $b = x_n$ | $h = (b-a)/n$ la fórmula se llama fórmula cerrada de Newton-Cotes.
 - Regla del Trapecio (El caso más simple de Fórmula de Newton-Cotes cerrada es con $n = 1$).
 - Regla de Simpson (La fórmula de Newton-Cotes con $n = 2$)
- Si se excluyen los puntos a y b → $a = x_0 - h$ | $b = x_n + h$ | $x_i = x_0 + ih$ | $h = (b-a)/(n+2)$ la fórmula se llama fórmula abierta de Newton-Cotes.

Precisión:

- Si n par → grado de exactitud $n+1$.
- Si n impar → grado de exactitud n .

Ejemplos:		
Método	Fórmula	Error
Trapecio (n=1)	$\frac{b-a}{2} [f(a) + f(b)]$	$-\frac{(b-a)^3}{12} f''(\xi)$
Simpson (n=2)	$\frac{b-a}{6} [f(a) + 4f(\frac{a+b}{2}) + f(b)]$	$-\frac{(b-a)^5}{2880} f^{(4)}(\xi)$
Simpson 3/8 (n=3)	$\frac{3b-a}{8} [f(x_0) + 3f(x_1) + 3f(x_2) + f(x_3)]$	$-\frac{3(b-a)^5}{80} f^{(4)}(\xi)$
n=4 fórmula	$\frac{2b-a}{45} [7f(x_0) + 32f(x_1) + 12f(x_2) + 32f(x_3) + 7f(x_4)]$	$-\frac{8(b-a)^7}{945} f^{(6)}(\xi)$

Las fórmulas de integración pueden usarse para integrar funciones definidas en una cantidad de puntos mayor. **Estas formulas se llaman compuestas.** Por ejemplo, la Regla del Trapecio Compuesta integra con su fórmula cada intervalo de definición de la función.

▼ Cuadratura.

Cuadratura es sinónimo de **integración definida numérica**, es decir, aproximar el valor de una integral definida cuando:

- No tenemos una primitiva fácil de calcular, o
- La función se conoce solo en ciertos puntos (por ejemplo, una tabla de datos), o
- Queremos automatizar el proceso computacionalmente.

La forma general de una fórmula de cuadratura es:

$$\int_a^b f(x) dx \approx \sum_{i=0}^n a_i f(x_i) + e_T$$

Donde:

- x_i : puntos o nodos en los que se evalúa la función.
- a_i : coeficientes (pesos).
- e_T : error de truncamiento (depende de la fórmula y del número de nodos).
- Si la fórmula es exacta para polinomios de grado $\leq n$, se dice que tiene orden n .

Los puntos x_i **no son equidistantes**, sino que son las **raíces de ciertos polinomios ortogonales** (como Legendre, Chebyshev, Laguerre, Hermite, dependiendo del intervalo y del peso).

Teorema clave de la Cuadratura de Gauss:

Si elegimos $n+1$ nodos x_i , que son las raíces de un **polinomio ortogonal $q(x)$ de grado $n+1$** respecto a una función peso $w(x)$, entonces:

$$\int_a^b f(x)w(x) dx = \sum_{i=0}^n a_i f(x_i)$$

es **exacta** para cualquier polinomio de grado hasta $2n+1$.

→ Función ortogonal:

Un conjunto de funciones ϕ_i se dice ortogonal respecto a una función peso $w(x)$ si:

$$\int_a^b \phi_j(x) \phi_k(x) w(x) dx = \begin{cases} 0, & j \neq k \\ > 0, & j = k \end{cases}$$

Ejemplos de funciones ortogonales:

- Polinomios de **Legendre**: $w(x)=1$, intervalo $[-1,1]$.
- Polinomios de **Chebyshev**: $w(x)=1/\sqrt{1-x^2}$ en $[-1,1]$.

***VER APLICACIÓN EN INTEGRALES MÚLTIPLES PORQUE LO PUEDE TOMAR**

Cuadratura de Gauss: en las fórmulas de Newton-Cotes los nodos se eligen a priori e igualmente espaciados. Luego, se calculan los coeficientes de la cuadratura utilizando los polinomios de Lagrange o el método de coeficientes indeterminados. En cambio, para las fórmulas de cuadratura Gaussiana se calculan no sólo los coeficientes de la cuadratura sino también las coordenadas de los nodos de forma tal de reducir el error al aproximar

$$\int_a^b f(x)dx \approx \sum_{i=1}^n c_i f(x_i)$$

Con esta generalización el grado de la clase de polinomios para la cuál la aproximación será exacta es mayor.

▼ Teoría Unidad 7:

Problemas con Valores Iniciales (PVI).

$$\begin{cases} y'' = f(x, y, y') & \text{para } a \leq x \leq b \\ y(a) = \bar{y}_a \\ y'(a) = \bar{y}'_a \end{cases}$$

O sea:

👉 Te dan una ecuación diferencial de segundo orden (con derivadas y todo)

👉 Y además te dicen **qué valor tiene la función y su derivada** en el punto de inicio $x=a$.

Con eso, se toma desde el punto a y se sigue la historia de la función hacia adelante, como si fuera una ruta ya marcada.

▼ Teoría Unidad 8:

$$\begin{cases} y'' = f(x, y, y') & \text{para } a \leq x \leq b \\ y(a) = \bar{y}_a \\ y(b) = \bar{y}_b \end{cases}$$

Esto **ya no es un problema con valores iniciales**, sino un Problema de Valor de Borde/Frontera/Contorno.

diferencia está en que ahora, te dicen el valor de la función en **dos puntos distintos** del intervalo, uno al principio $y(a)$, y otro al final $y(b)$, pero **no te dicen la derivada en ninguno**.

Existencia y unicidad de la solución.

Si la función $f(x, y, y')$ que aparece en la ecuación cumple con lo siguiente:

Es continua, sus **derivadas parciales** también son suaves y $\partial f / \partial y > 0$ en todo el dominio. Existe una constante M tal que: $|\partial f / \partial y'| \leq M$ (los cambios respecto a la pendiente y' **están acotados**, no se van al infinito).

PVC lineales.

Si la función $f(x, y, y')$ puede expresarse: $f(x, y, y') = p(x) y' + q(x) y + r(x)$ la ecuación diferencial $y'' = f$ se dice lineal. *Si es continua y > 0 en $[a, b]$ entonces tiene solución única.

El método del disparo.

Es una estrategia para resolver un problema de valor de contorno **transformándolo** en uno de valor inicial.

Dado un PVC, se sabe cuánto vale la función en los bordes a y b , pero **no sabés cómo arranca la pendiente** $y'(a)$.

Entonces convertimos el PVC en un Problema de Valor Inicial (PVI) así:

$$\begin{cases} y'' = f(x, y, y') \\ y(a) = \alpha \\ y'(a) = z \end{cases}$$

A esta solución la llamamos $y_z(x)$, porque depende del valor inicial z que elegimos.

Queremos que esa solución cumpla **también** la condición de que $y_z(b) = \beta$ entonces armamos esta función:

$\varphi(z) = y_z(b) - \beta$ entonces buscamos que se anule:

$\varphi(z) = 0 \Rightarrow y_z(b) = \beta$

Como $\varphi(z)$ es una función **no lineal**, usamos métodos numéricos como:

- Método de la **bisección**
- Método de la **secante**
- Método de **Newton** (si podemos)

Paso a paso:

Entradas: Intervalo $[a,b]$ - Valores de contorno α, β - Cantidad de pasos N - Tolerancia - Máximo de iteraciones

1. Estimar una primera pendiente $z = (b-a)/(\beta-\alpha)$
2. Resolver PVI con $y'(a)=z$
3. Calcular $\varphi(z)=y_z(b)-\beta$

*Si el PVI es lineal, **combinación lineal** de dos soluciones, y te ahorrás el loop del método de la secante.

Problema	Lo que sabés	Lo que hacés
PVC	$y(a), y(b)$	Buscás $y'(a)$ que encaje
PVI	$y(a), y'(a)$	Integrás para encontrar $y(b)$
Método del disparo	Convertís el PVC en PVI	Ajustás $y'(a)$ para que la solución encaje con $y(b)$

El método de las diferencias finitas.

Dado un PVC lineal, es decir de la forma:

$$\begin{cases} y'' = p(x)y' + q(x)y + r(x), & x \in [a, b] \\ y(a) = \alpha \\ y(b) = \beta \end{cases}$$

La idea es **aproximar** la solución de esta EDO **no resolviéndola analíticamente**, sino convirtiéndola en un **sistema de ecuaciones algebraicas**, o sea, una matriz.

Paso a paso:

1. Dividís el intervalo.

Tomás el intervalo $[a, b]$ y lo dividís en $N + 1$ pedacitos (subintervalos), lo que te da N nodos interiores.

$$h = \frac{b - a}{N + 1}$$

Entonces tenés los puntos:

$$x_0 = a, \quad x_1, \quad x_2, \quad \dots, \quad x_N, \quad x_{N+1} = b$$

Y llamás a $y(x_i) = y_i$, o sea, los valores de la solución en esos nodos.

2. Reemplazás derivadas por diferencias finitas.

Para derivadas, usás fórmulas aproximadas que salen del desarrollo en serie de Taylor:

- Primera derivada centrada:
$$y'(x_i) \approx \frac{y_{i+1} - y_{i-1}}{2h}$$
- Segunda derivada centrada:
$$y''(x_i) \approx \frac{y_{i+1} - 2y_i + y_{i-1}}{h^2}$$

Estas fórmulas tienen un error chiquito del orden $O(h^2)$, o sea, son bastante buenas.

3. Reemplazás en la ecuación original.

La ecuación era:

$$y'' = p(x)y' + q(x)y + r(x)$$

Reemplazás todo con las diferencias finitas en el nodo x_i :

$$\frac{y_{i+1} - 2y_i + y_{i-1}}{h^2} = p_i \cdot \frac{y_{i+1} - y_{i-1}}{2h} + q_i y_i + r_i$$

Y ahora tenés una **ecuación algebraica** que conecta tres nodos: y_{i-1}, y_i, y_{i+1}

La reorganizás y queda así:

$$\left(-1 - \frac{h}{2}p_i\right)y_{i-1} + (2 + h^2q_i)y_i + \left(-1 + \frac{h}{2}p_i\right)y_{i+1} = -h^2r_i$$

✳ Esa ecuación es una fila de tu sistema lineal.

4. Agregás las condiciones de contorno

Como sabés que:

- $y_0 = \alpha$
- $y_{N+1} = \beta$

Podés meter esos valores como datos conocidos y modificar el primer y último renglón del sistema para tener todo en función de las incógnitas y_1, y_2, \dots, y_N .

El sistema completo tiene la forma:

$$A \cdot y = f$$

Donde:

- A es una matriz **tridiagonal** (solo tiene elementos en la diagonal principal y las dos diagonales vecinas).
- y es el vector de incógnitas $[y_1, y_2, \dots, y_N]^T$
- f es un vector con los términos independientes, **ajustados por las condiciones de contorno**.

Teorema de existencia y unicidad.

Si p, q, r son funciones continuas en $[a, b]$, y si además $q(x) \geq 0$ para todo x , entonces el sistema tiene una **única solución** si:

$$h < \frac{2}{L}, \quad L = \max_{x \in [a, b]} |p(x)|$$

Esto te dice que mientras no elijas una **discretización muy grosera**, vas a estar bien.

Ecuaciones Diferenciales Parciales (EDP).

Son ecuaciones donde la incógnita $\phi(x, y)$ depende de **más de una variable** y aparecen derivadas **parciales**. Un ejemplo genérico de segundo orden es:

$$a_{11} \frac{\partial^2 \phi}{\partial x^2} + 2a_{12} \frac{\partial^2 \phi}{\partial x \partial y} + a_{22} \frac{\partial^2 \phi}{\partial y^2} = F(x, y, \phi, \phi_x, \phi_y)$$

Estas EDPs aparecen en todo: calor, vibraciones, fluidos, electromagnetismo, etc.

Según el discriminante $D = a_{12}^2 - a_{11}a_{22}$:

Tipo	Condición	Ejemplo
Elíptica	$D < 0$	Laplace, Poisson (calor estacionario)
Parabólica	$D = 0$	Ecuación del calor (transitorio)
Hiperbólica	$D > 0$	Onda (vibración de cuerda)



Condiciones de Contorno (CC)

Necesarias para que el problema tenga una **solución única**.

Tipo	Ejemplo
Dirichlet	$\phi = \tilde{\phi}$ en el borde
Neumann	$\frac{\partial \phi}{\partial n} = \tilde{q}$ (flujo normal)
Robin	$a\phi + b\frac{\partial \phi}{\partial n} = \tilde{g}$ mezcla de ambas

Método de Diferencias Finitas (MDF):

Convierte una EDP en un **sistema de ecuaciones lineales** al discretizar el dominio en una grilla.
El error del método es del orden $O(h^2)$