

ECUACIONES DIFERENCIALES ORDINARIAS

Se considera el problema de valor inicial (PVI):

$$\begin{cases} y'(x) = f(x, y(x)), & x \in [a, b] \\ y(a) = y_0 \text{ dado,} \end{cases}$$

el que supondremos tiene solución única, $y : [a, b] \rightarrow \mathbb{R}$, la cual es acotada y depende continuamente de los datos f e y_0 .

Teorema. Sean D un subconjunto convexo de \mathbb{R}^2 , f una función continua en el dominio D y (x^*, y^*) un punto interior de D .

Si f satisface la condición de Lipschitz

$$|f(x, y_1) - f(x, y_2)| \leq k|y_1 - y_2|, \quad \forall (x, y_1), (x, y_2) \in D, \quad (*)$$

con $k \geq 0$, entonces, para algún $I = [x^* - \alpha, x^* + \alpha]$ existe una única solución y del PVI definida en I .

Observación. Si $\frac{\partial f(x, y)}{\partial y}$ existe y es acotada en D , entonces $(*)$ se satisface con

$$k = \max_{(x, y) \in D} \left| \frac{\partial f}{\partial y}(x, y) \right| < \infty.$$

Ejemplo. Considere la ecuación

$$y' = 1 + \operatorname{sen} xy \quad y \quad D = \{(x, y) \in \mathbb{R}^2, 0 \leq x \leq 1, -\infty < y < \infty\}.$$

Solución. Se tiene que:

$$f(x, y) = 1 + \operatorname{sen}(xy), \quad y \quad \frac{\partial f}{\partial y}(x, y) = x \cos(xy) \Rightarrow k = 1.$$

Luego, dado cualquier (x_0, y_0) con $0 < x_0 < 1$, por el teorema existe una única solución en $I = [0, 1]$.

Una **solución numérica** para el PVI es un conjunto de N números y_1, y_2, \dots, y_N que aproximan a los valores $y(x_1), \dots, y(x_N)$ de la solución exacta en los *odos* x_1, x_2, \dots, x_N de $[a, b]$. Supondremos los nodos definidos por $x_i = a + ih$, $i = 0, \dots, N$ siendo $h = (b - a)/N$.

METODO DE LA TANGENTE

Considere el P.V.I.

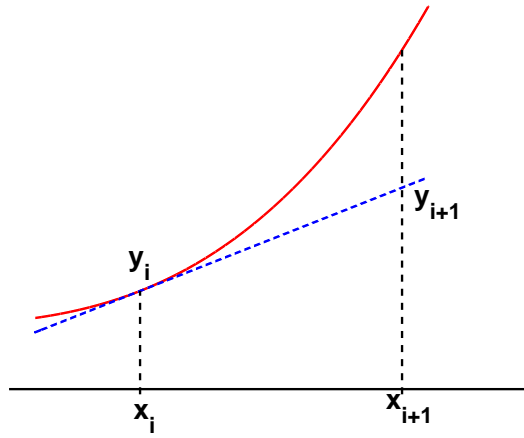
$$\begin{cases} y'(x) = f(x, y(x)), & x \in [a, b] \\ y(a) = y_0 \text{ dado.} \end{cases}$$

Una manera geométrica de aproximar la solución de este problema es reemplazando la derivada y' por alguna aproximación; por ejemplo

$$y'(x) \approx \frac{y(x+h) - y(x)}{h}$$

válida para h pequeño.

Algoritmo de la Tangente



Haciendo este reemplazo en la ecuación se encuentra

$$\frac{y(x+h) - y(x)}{h} \approx f(x, y(x))$$

de donde,

$$y(x+h) \approx y(x) + hf(x, y(x)). \quad (1)$$

Partiendo de la condición inicial $y(a) = y_0$ y considerando h pequeño, el valor

$$y_1 := y(a) + hf(a, y(a)) \quad (2)$$

define una aproximación para $y(a+h)$. Conocida esta aproximación se puede utilizar en (1) para obtener la aproximación y_2 de $y(a+2h)$, a saber

$$y_2 := y_1 + hf(a+h, y_1). \quad (3)$$

Repitiendo este proceso se pueden obtener aproximaciones para $y(a+3h)$, $y(a+4h)$, \dots , $y(a+Nh)$.

Usando los nodos x_i definidos anteriormente obtenemos el siguiente [algoritmo de la tangente](#).

Algoritmo

Para $i = 0, \dots, N - 1$

$$x_i = a + ih$$

$$y_{i+1} = y_i + hf(x_i, y_i)$$

i continue \square

Suponiendo que y_i es igual a $y(x_i)$ podemos acotar (estimar) el error de la aproximación y_{i+1} , cuando $y \in C^2([a, b])$, como sigue:

$$y(x_{i+1}) = y(x_i) + hy'(x_i) + \frac{h^2}{2}y''(z_i),$$

$$y(x_{i+1}) = y_i + hf(x_i, y_i) + \frac{h^2}{2}y''(z_i),$$

$$y(x_{i+1}) = y_{i+1} + \frac{h^2}{2}y''(z_i),$$

donde $x_i < z_i < x_{i+1}$. De esta última igualdad se encuentra

$$|y(x_{i+1}) - y_{i+1}| \leq Ch^2,$$

para

$$C = \max_{z \in [a, b]} \left| \frac{y''(z)}{2} \right|.$$

MÉTODOS DE RUNGE-KUTTA

Los métodos de Runge-Kutta consideran, en cada intervalo $[x_i, x_{i+1}]$, algunos *nodos auxiliares* de la forma:

$$x_{ij} := x_i + \theta_j h, \quad j = 0, 1, \dots, q \quad (4)$$

donde, $q \in \mathbb{N}$ recibe el nombre de *rango del algoritmo* y los parámetros θ_j verifican: $0 \leq \theta_j \leq 1$ con $\theta_0 = 0$ y $\theta_q = 1$; así, en cada caso $x_{i0} = x_i$ y $x_{iq} = x_{i+1}$.

En los nodos auxiliares se definen los valores: $y_{i0} := y_i$,

$$y_{ij} := y_i + h \sum_{l=0}^{j-1} A_{jl} f(x_{il}, y_{il}), \quad j = 1, \dots, q \quad (5)$$

siendo $y_{i+1} := y_{iq}$ el valor aproximado por el método para $y(x_{i+1})$.

Cuando los puntos x_{ij} de (4) y las constantes A_{jl} de (5) se determinan de manera que el desarrollo de y_{ij} coincida hasta el término en h^p con el desarrollo en serie de Taylor de la solución exacta del *problema local*:

$$\begin{aligned} y'(x) &= f(x, y(x)), \quad x \in [x_i, x_{i+1}] \\ y(x_i) &= y_i, \quad \underline{\text{con } y_i \text{ considerada exacta,}} \end{aligned} \quad (6)$$

el correspondiente algoritmo se dice de *orden* p y *rango* q , lo que se abrevia por RK_{pq} .

Definición. Se define el *error de truncamiento local* (por paso) en x_{i+1} por $E_{i+1} := y(x_{i+1}) - y_{i+1}$, donde y es la solución del problema (6) e y_{i+1} es el valor obtenido por el método.

Definición. Se define el *error de truncamiento global* como el máximo de los valores absolutos de todos los errores de truncamiento local, es decir

$$E := \max_{0 \leq i \leq N-1} |E_{i+1}|.$$

Para un método de Runge-Kutta de orden p el error global resulta ser de orden h^p .

Método de Euler

Corresponde al caso en que $q = p = 1$, luego RK_{11} es un método de orden uno, rango uno y error por paso $O(h^2)$. Además, $x_{i0} = x_i$, $x_{i1} = x_{i+1}$ y los valores de (5) dan:

$$\begin{aligned} y_{i0} &= y_i \\ y_{i1} &= y_i + hA_{10}f(x_{i0}, y_{i0}), \end{aligned}$$

es decir,

$$\begin{aligned} y_{i0} &= y_i \\ y_{i1} &= y_i + hA_{10}f(x_i, y_i) \quad (*) \end{aligned}$$

Considere ahora el desarrollo en serie de Taylor para la solución del PVI (6):

$$y(x_{i+1}) = y(x_i) + hy'(x_i) + \cdots + \frac{h^n}{n!}y^{(n)}(x_i) + R_{n+1}(\phi_i),$$

donde

$$R_{n+1}(\phi_i) = \frac{h^{n+1}}{(n+1)!}y^{(n+1)}(\phi_i), \quad x_i < \phi_i < x_{i+1}.$$

Comparando este desarrollo con el de (*) se ve que coinciden hasta términos en h^1 para $A_{10} = 1$. Luego (*) conduce al proceso:

$$y_{i+1} = y_i + hf(x_i, y_i), \quad i = 0, 1, \dots, N-1;$$

el cual corresponde al algoritmo de la tangente deducido anteriormente de una manera geométrica.

Algoritmo de Euler:

Algoritmo

Para $i = 0, \dots, N-1$

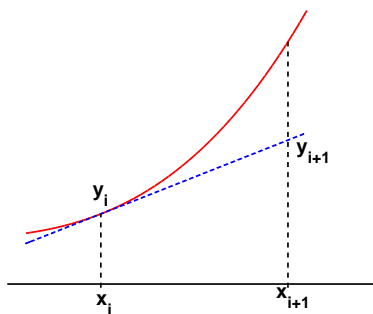
$$x_i = a + ih$$

$$y_{i+1} = y_i + hf(x_i, y_i)$$

i continue ▢

Geométricamente:

Algoritmo de Euler



Algoritmos RK_{22}

Como $p = q = 2$ estos métodos resultan ser de orden dos ($p = 2$), rango dos ($q = 2$) y error local de orden h^3 . Los nodos de (4) se obtienen con $j = 0, 1, 2$ y son $x_{i0} = x_i$, $x_{i1} = x_i + \theta_1 h$ y $x_{i2} = x_{i+1}$. Los desarrollos en potencias de h para los correspondientes valores y_{i1} e y_{i2} , se obtienen de (5) con $q = 2$. Tales desarrollos se igualan a los desarrollos en serie de Taylor de la solución de (6) hasta términos en h^2 , ($p = 2$), para obtener la siguiente familia de algoritmos RK_{22} :

$$\begin{aligned} y_{i1} &:= y_i + h\theta_1 f(x_i, y_i) \\ x_{i1} &:= x_i + \theta_1 h \\ y_{i+1} &:= y_i + h\left(1 - \frac{1}{2\theta_1}\right)f(x_i, y_i) + h\left(\frac{1}{2\theta_1}\right)f(x_{i1}, y_{i1}). \end{aligned} \quad (7)$$

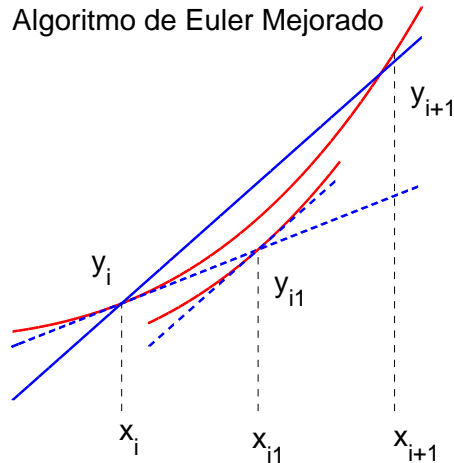
donde $y_i = y_{i0}$, $y_{i+1} = y_{i2}$ y el parámetro θ_1 es tal que $0 < \theta_1 \leq 1$.

Casos particulares

1.-) Algoritmo de Euler mejorado (Tangente mejorada)

Corresponde a la elección de $\theta_1 = \frac{1}{2}$, con lo cual x_{i1} resulta ser el punto medio del intervalo $[x_i, x_{i+1}]$. De (7) se obtiene $y_{i+1} = y_i + hf(x_{i1}, y_{i1})$ y el algoritmo queda:

Algoritmo
 Para $i = 0, \dots, N - 1$
 $x_i = a + ih$
 $x_{i1} = x_i + \frac{h}{2}$
 $y_{i1} = y_i + \frac{h}{2}f(x_i, y_i)$
 $y_{i+1} = y_i + hf(x_{i1}, y_{i1})$
i continue □



2.-) Algoritmo de Euler-Cauchy

Corresponde al caso en que $\theta_1 = 1$, con lo cual $x_{i1} = x_{i+1}$ y el algoritmo (7) resulta ser:

Algoritmo

Para $i = 0, \dots, N - 1$

$$x_i = a + ih$$

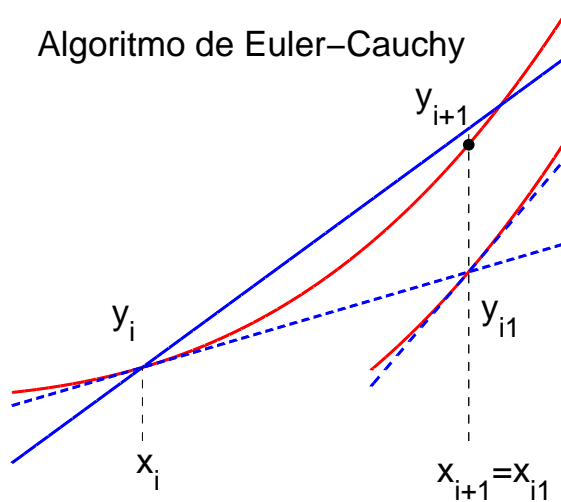
$$x_{i1} = x_i + h$$

$$y_{i1} = y_i + hf(x_i, y_i)$$

$$y_{i+1} = y_i + \frac{h}{2}[f(x_i, y_i) + f(x_{i1}, y_{i1})]$$

i continue □

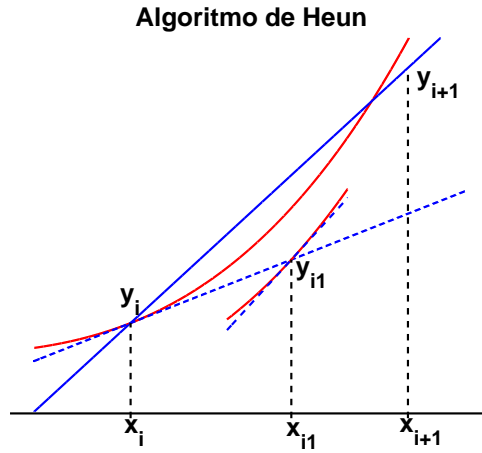
Algoritmo de Euler-Cauchy



3.-) Algoritmo de Heun

Ahora $\theta_1 = \frac{2}{3}$, luego $x_{i1} = x_i + \frac{2}{3}h$ y el algoritmo (7) conduce a:

Algoritmo
 Para $i = 0, \dots, N - 1$
 $x_i = a + ih$
 $x_{i1} = x_i + \frac{2}{3}h$
 $y_{i1} = y_i + \frac{2}{3}hf(x_i, y_i)$
 $y_{i+1} = y_i + \frac{h}{4}[f(x_i, y_i) + 3f(x_{i1}, y_{i1})]$
i continue ◻



Algoritmo RK_{44} Clásico

Como en estos métodos $q = 4$, entonces de (5) se obtienen desarrollos en potencias de h para y_{i1} , y_{i2} , y_{i3} , e y_{i4} . Tales desarrollos se igualan a los desarrollos en serie de Taylor de la solución de (6), hasta términos en h^p , con $p = 4$, para obtener una infinidad de algoritmos RK_{44} , todos con error por paso de orden h^5 . (Similar al caso RK_{22} donde θ_1 apareció como el parámetro de una familia). El siguiente algoritmo particular, comúnmente utilizado, es conocido como el **algoritmo RK_{44} clásico**:

Algoritmo
 Para $i = 0, \dots, N - 1$
 $x_i = a + ih$
 $k_1 = hf(x_i, y_i)$
 $k_2 = hf(x_i + \frac{h}{2}, y_i + \frac{1}{2}k_1)$
 $k_3 = hf(x_i + \frac{h}{2}, y_i + \frac{1}{2}k_2)$
 $k_4 = hf(x_i + h, y_i + k_3)$
 $y_{i+1} = y_i + \frac{1}{6}[k_1 + 2(k_2 + k_3) + k_4]$
i continue ◻

CONTROL DEL ERROR POR TRUNCAMIENTO LOCAL

Resulta más fácil estimar el error local por truncamiento a posteriori que a priori.

Una forma de hacer esto se puede deducir de la siguiente manera. Suponga que usamos el mismo método de RK, pero con pasos diferentes ($2h$ y h).

Si un método de RK de orden p es usado, partiendo del dato *exacto* (que no está disponible para $n \geq 1$), se tiene:

$$y(x_{n+1}) - y_{n+1} = \Phi(y_n)h^{p+1} + O(h^{p+2}) \quad (8)$$

donde Φ es una función incógnita evaluada en y_n .

Haciendo este mismo cálculo para $2h$, se tiene:

$$y(x_{n+1}) - u_{n+1} = \Phi(y_n)(2h)^{p+1} + O(h^{p+2}) \quad (9)$$

donde u_{n+1} es la solución calculada para t_{n+1} .

Restando (9) a (8), se encuentra:

$$(2^{p+1} - 1) h^{p+1} \Phi(y_n) = y_{n+1} - u_{n+1}, \quad (10)$$

reemplazando en (8), se llega a:

$$y(x_{n+1}) - y_{n+1} \simeq \frac{y_{n+1} - u_{n+1}}{(2^{p+1} - 1)} = \epsilon. \quad (11)$$

Este método requiere de un gran esfuerzo computacional (debido a las evaluaciones necesarias de la función f). Sin embargo

$$|u_{n+1} - y_{n+1}| \quad (12)$$

nos da una estimación a posteriori del error local.

Una pregunta que resulta natural es:

¿ Cómo poder variar el paso h para que el error por truncamiento local sea menor que una tolerancia prefijada ?.

Para responder esta pregunta se debe cosiderar el uso simultáneo de dos métodos de RK con orden diferente.

Uno de los esquemas más usado, que usa tal estrategia, es el de **Runge-Kutta-Fehlberg** de cuarto orden (RK45) el cual consiste en un RK de orden 4 acoplado con un RK de orden 5, este esquema minimiza el número de evaluaciones de f a 6 en vez de 10 por paso. Además este método permite una selección automática del paso de integración h (métodos adaptivos o de paso variable) como se puede apreciar en el algoritmo que sigue.

Algoritmo

Dados h y ϵ
 Para $i = 0, \dots, N - 1$
 (1) $x_{i+1} = x_i + h$
 $k_1 = hf(x_i, y_i)$
 $k_2 = hf(x_i + \frac{1}{4}h, y_i + \frac{1}{4}k_1)$
 $k_3 = hf(x_i + \frac{3}{8}h, y_i + \frac{3}{32}k_1 + \frac{9}{32}k_2)$
 $k_4 = hf(x_i + \frac{12}{13}h, y_i + \frac{1932}{2197}k_1 - \frac{7200}{2197}k_2 + \frac{7296}{2197}k_3)$
 $k_5 = hf(x_i + h, y_i + \frac{439}{216}k_1 - 8k_2 + \frac{3680}{513}k_3 - \frac{845}{4104}k_4)$
 $k_6 = hf(x_i + \frac{1}{2}h, y_i - \frac{8}{27}k_1 + 2k_2 - \frac{3544}{2565}k_3 + \frac{1859}{4104}k_4 - \frac{11}{40}k_5)$
 $y_{i+1} = y_i + \frac{25}{216}k_1 + \frac{1408}{2565}k_3 + \frac{2197}{4104}k_4 - \frac{1}{5}k_5$
 $E_{i+1} := \left| \frac{1}{360}k_1 - \frac{128}{4275}k_3 - \frac{2197}{75240}k_4 + \frac{1}{50}k_5 + \frac{2}{55}k_6 \right|$
 $s := 0.84 \left(\frac{\epsilon}{E_{i+1}} \right)^{1/4}$
 Si $|E_{i+1}| < \epsilon$ hacer $h = 2sh \rightarrow$ (2)
 $h := sh \rightarrow$ (1)
 (2) i continue \blacksquare