Métodos numéricos

Raíces de ecuaciones

Regla de Horner o de división sintética	Evaluar un polinomio en un punto x_o es equivalente a calcular el resto de la división por $x - x_o$. (Es decir, hacemos "Ruffini" y nos quedamos con el resto)
Raíces múltiples de ecuaciones algebraicas	La condición necesaria y suficiente para que un número x_o sea una raíz múltiple de orden k de un polinomio P (x) es que anule a dicho polinomio y a sus k - 1 primeras derivadas, pero no a la siguiente.
Raíces enteras de ecuaciones	- Toda raíz entera de una ecuación algebraica con coeficientes enteros es un divisor del término independiente. - Toda raíz entera de una ecuación P (x) = 0, donde los coeficiente de P (x) son enteros, verifica simultáneamente: P (1) es múltiplo de $(x_o - 1)$ P (-1) es múltiplo de $(x_o + 1)$
Raíces fraccionarias de ecuaciones algebraicas	Para encontras las raíces fraccionarias de la ecuación $P(x) = a_n x^n + a_{n-1} x^{n-1} + \ldots + a_1 x + a_o$, se efectúa la transformación $x = y / a_n$, calculándose las raíces enteras de esta nueva ecuación. Después se deshace el cambio.
Cota de Cardano - Vieta de las raíces reales de una ecuación	Si todas las raíces de la ecuación $P(x) = a_n x^n + a_{n-1} x^{n-1} + \ldots + a_1 x + a_o = 0$ son REALES, se verifica que $\lvert \square M \rvert \leq [(a_{n-1} / a_n)^2 - 2 (a_{n-2} / a_n)]^{1/2}$ siendo $x_i \in [- M , M]$ para toda raíz x_i de la ecuación P (x) = 0.
Teorema de Bolzano	Sea y = f (x) continua en el intervalo [a, b], teniendo f (a) y f (b) signos opuestos, entonces existe un punto intermedio $c \in (a, b)$ que anula la función: f (c) = 0
Método de la bisección o del semi-intervalo	Básándose en el terorema de Bolzano, partimos de un intervalo en el que la función cambia de signo y evaluamos el signo de la función en el punto medio de dicho intervalo. Reduciremos el nuevo intervalo a aquél en el que se produzca de nuevo un cambio de signo. Repitiendo el proceso hasta obtener la precisión requerida.
Método de iteración del punto fijo	Consideremos una ecuación de la forma f (x) = 0, en la cual podremos despejar x "en función de x" (curioso, ¿no?), es decir la escribimos de la forma x = g (x). Partimos de un punto x_o, de forma que la primera iteración será $x_1 = g(x_o)$; la segunda: $x_2 = g(x_1)$, etc. En general: $\mathbf{x_n = g(x_{n-1})}$
Convergencia del método:	$-1 < g'(x_o) < 1$
Método de Newton - Raphson	Partimos de un punto x_o y calculamos el punto x_1 donde corla la recta tangente a la curva y = f (x) desde ese punto con el eje de abcisas. $x_1 = x_o - [f(x_o) / f'(x_o)]$ En general: $\mathbf{x_{n+1} = x_n - [f(x_n) / f'(x_n)]}$ Inconveniente: debemos evaluar la función y la derivada en cada punto.
Convergencia del método:	$\lvert g'(x_o) \rvert < 1$
Método de Newton modificado	$\mathbf{x_{n+1} = x_n - [f(x_n) / f'(x_o)]}$ Evaluamos la derivada sólo en el primer punto.

Derivación numérica

Desarrollo en serie de Taylor en torno al punto x_o	$f(x) \cong f(x_o) + f'(x_o)(x - x_o) / 1! + f''(x_o)(x - x_o)^2 / 2! + f'''(x_o)(x - x_o)^3 / 3! + \ldots$
Nos quedamos con los dos primeros términos:	$f(x) \cong f(x_o) + f'(x_o)(x - x_o) + \ldots$
Evaluamos en el punto $x = x_o + h$	$f(x_o + h) \cong f(x_o) + f'(x_o) h + O(h^2)$

Primera aproximación a la derivada:	**$f'(x_o) = [f(x_o + h) - f(x_o)] / h + O(h)$**
Segunda aproximación a la derivada:	**$f'(x_o) = [f(x_o + h) - f(x_o - h)] / (2h) + O(h^2)$**

Integración numérica

$_a\int^b f(x)\,dx$	La integral definida entre a y b de la función f (x) nos da el área de la región limitada por la curva y = f (x) y el eje x
Tamaño del paso h	$h = (b - a) / N$ donde N es el número de intervalos
Regla del rectángulo	$_a\int^b f(x)\,dx \cong$ **$h\,[f(a) + f(a + h) + f(a + 2h) + \ldots + f(b - h)]$**
	$_a\int^b f(x)\,dx \cong$ **$h\,[f(a + h) + f(a + 2h) + \ldots + f(b - h) + f(b)]$**
	N sumandos
	Error: $O(h)$
Regla del punto medio	$_a\int^b f(x)\,dx \cong$ **$h\,[f(a + h/2) + f(a + 3h/2) + \ldots + f(b - h/2)]$**
	N sumandos
	Error: $O(h^2)$
Regla del trapecio	$_a\int^b f(x)\,dx \cong$ **$h\,[f(a) / 2 + f(a + h) + f(a + 2h) + \ldots + f(b - h) + f(b) / 2]$**
	N + 1 sumandos
	Error: $O(h^2)$
Regla de Simpson 1/3	$_a\int^b f(x)\,dx \cong$ **$h/3\,[f(a) + 4f(a + h) + 2f(a + 2h) + 4f(a + 3h) + 2f(a + 4h) \ldots + f(b)]$**
	N debe ser un número par
	N + 1 sumandos
	Error: $O(h^4)$
Método de Romberg	En combinación con el método del trapecio
	$I_c(h) = I_v + a\,h^2 + O(h^4)$ $I_c(h/2) = I_v + a\,(h/2)^2 + O(h^4)$
	Multiplicando la segunda ecuación por 4, restando la primera y despejando I_v, se obtiene: $I_v = [4\,I_c(h/2) - I_c(h)] / 3 + O(h^4)$
	Después se repite el proceso.

Interpolación y aproximación

Interpolación lineal	Polinomio interpolador de primer grado que pasa por los puntos $P_o(x_o, f(x_o))$ y $P_1(x_1, f(x_1))$
	$P(x) = f(x_o) + (x - x_o)\,[f(x_1) - f(x_o)] / [x_1 - x_o]$

Interpolación de Lagrange	Dados (n+1) puntos, se trata de calcular el polinomio interpolador de grado n que pasa por todos ellos.
Si n = 2 (tres puntos)	$P_o\ (x_o, f\ (x_o))$, $P_1\ (x_1, f\ (x_1))$, $P_2\ (x_2, f\ (x_2))$
	$\mathbf{P\ (x) = f\ (x_o)\ L_o\ (x) + f\ (x_1)\ L_1\ (x) + f\ (x_2)\ L_2\ (x)}$
donde:	$L_o\ (x) = [\ (x - x_1)\ (x - x_2)\ ]\ /\ [\ (x_o - x_1)\ (x_o - x_2)\ ]$
	$L_1\ (x) = [\ (x - x_o)\ (x - x_2)\ ]\ /\ [\ (x_1 - x_o)\ (x_1 - x_2)\ ]$
	$L_2\ (x) = [\ (x - x_o)\ (x - x_1)\ ]\ /\ [\ (x_2 - x_o)\ (x_2 - x_1)\ ]$
Recta de mínimos cuadrados	$\mathbf{y = a + b\ x}$
	Construimos la función de dos variables: $f\ (a, b) = \Sigma (a + b\ x_i - y_i)^2$ donde el sumatorio se extiende desde i = 1 al número de datos n. La condición de mínimo (o máximo) exige que $\partial f / \partial a = 0$ y que $\partial f / \partial b = 0$. Ello conduce al sistema:
	$\mathbf{a\ n + b\ \Sigma x_i = \Sigma y_i}$
	$\mathbf{a\ \Sigma x_i + b\ \Sigma x_i^2 = \Sigma x_i\ y_i}$
Parábola de mínimos cuadrados	$\mathbf{y = a + b\ x + c\ x^2}$
	Construimos la función de tres variables: $f\ (a, b, c) = \Sigma (a + b\ x_i + c\ x_i^2 - y_i)^2$ donde el sumatorio se extiende desde i = 1 al número de datos n. La condición de mínimo (o máximo) exige que $\partial f / \partial a = \partial f / \partial b = \partial f / \partial c = 0$. Ello conduce al sistema:
	$\mathbf{a\ n + b\ \Sigma x_i + c\ \Sigma x_i^2 = \Sigma y_i}$
	$\mathbf{a\ \Sigma x_i + b\ \Sigma x_i^2 + c\ \Sigma x_i^3 = \Sigma x_i\ y_i}$
	$\mathbf{a\ \Sigma x_i^2 + b\ \Sigma x_i^3 + c\ \Sigma x_i^4 = \Sigma x_i^2\ y_i}$
Tipo potencial	$\mathbf{y = a\ x^b}$
	Tomando logaritmos (da igual en qué base, por ejemplo neperianos -en base e-) ln y = ln a + b ln x Llamando Y = ln y A = ln a X = ln X podemos reducir el problema a la recta de mínimos cuadrados: Y = A + b X (No olvidar deshacer los cambios)
Tipo exponencial	$\mathbf{y = a\ b^x}$
	Tomando logaritmos (da igual en qué base, por ejemplo neperianos -en base e-) ln y = ln a + x ln b Llamando Y = ln y A = ln a B = ln b podemos reducir el problema a la recta de mínimos cuadrados: Y = A + B x (No olvidar deshacer los cambios)

Ecuaciones diferenciales

Queremos resolver la ecuación diferencial $y' = f(x, y)$ con la condición $y(x_o) = y_o$

Método de Taylor de tres términos	
Desarrollo en serie de Taylor:	$y(x) \cong y(x_o) + y'(x_o)(x - x_o) / 1! + y''(x_o)(x - x_o)^2 / 2! + \dots$
Evaluamos en el punto $x = x_n + h$ (con $x_o = x_n$):	$y(x_n + h) \cong y(x_n) + y'(x_n) h + y''(x_n) h^2 / 2! + \dots$
En una notación más compacta:	$\mathbf{y_{n+1} = y_n + y_n' h + y_n'' h^2 / 2}$
Método de Euler o de las tangentes	$y_{n+1} = y_n + y_n' h =$ $\mathbf{y_{n+1} = y_n + h f(x_n, y_n)}$
Método de Euler modificado	$\mathbf{y_{n+1} = y_n + (h/2) [f(x_n, y_n) + f(x_{n+1}, y^*_{n+1})]}$
donde	$\mathbf{y^*_{n+1} = y_n + h f(x_n, y_n)}$
Método de Runge - Kutta	$\mathbf{y_{n+1} = y_n + [k_1 + 2 k_2 + 2 k_3 + k_4] / 6}$
donde	$\mathbf{k_1 = h f(x_n, y_n)}$
	$\mathbf{k_2 = h f(x_n + h/2, y_n + k_1/2)}$
	$\mathbf{k_3 = h f(x_n + h/2, y_n + k_2/2)}$
	$\mathbf{k_4 = h f(x_n + h, y_n + k_3)}$

Sistemas de ecuaciones diferenciales de primer orden

$x'(t) = f(t, x, y)$ $y'(t) = g(t, x, y)$	$x(t_o) = x_o$ $y(t_o) = y_o$
Taylor	$\mathbf{x_{n+1} = x_n + x_n' h + x_n'' h^2 / 2}$ $\mathbf{y_{n+1} = y_n + y_n' h + y_n'' h^2 / 2}$

Euler	$\mathbf{x_{n+1} = x_n + h\, x_n'}$ $\mathbf{y_{n+1} = y_n + h\, y_n'}$
donde	$x_n' = f(t_n, x_n, y_n)$ $y_n' = g(t_n, x_n, y_n)$ $x_n'' = f'(t_n, x_n, y_n)$ $y_n'' = g'(t_n, x_n, y_n)$